DB2 Server for VM

Messages and Codes

Version 7 Release 1

GC09-2984-00

DB2 Server for VM

Messages and Codes

Version 7 Release 1

GC09-2984-00

Note!
Before using this information and the product it supports, be sure to read the general information under "Notices" on page 431.

First Edition (September 2000)

This edition applies to Version 7, Release 1, Modification 0 of the IBM® DATABASE 2™ Server for VSE & VM Program, (product number 5697-F42) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces GC09-2664-00.

© Copyright International Business Machines Corporation 1987, 2000. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About This Manual v
Conventions Used in This Manual vi
Online HELP Text vi
Messages Issued from EXECs vi
Components of the Relational Database Management System vi
SQL Reserved Words vii
How to Send Your Comments viii

Summary of Changes xi
Summary of Changes for DB2 Version 7 Release 1 xi
Enhancements, New Functions, and New Capabilities xi
Reliability, Availability, and Serviceability Improvements xiii
Library Enhancements xiii

Chapter 1. Introduction 1
Using This Manual 1
Message Format and Associated Text 1

Chapter 2. ARI0001–ARI0198 DB2 Server for VM Control Messages 3

Chapter 3. ARI0200–ARI0299, ARI2000–ARI2048 DBSS Messages . . 43

Chapter 4. ARI2900–ARI2916 Common Messages 77

Chapter 5. ARI0300–ARI0399 CMS EXEC Messages. 81

Chapter 6. ARI0428–ARI0499 Resource Adapter Messages. 93

Chapter 7. ARI4021–ARI4175 DB2 Server for VM Control Messages . . . 97

Chapter 8. ARI0500–ARI0513 SQL Messages 105

Chapter 9. ARI0520–ARI0535, ARI5806 Catalog Generation/Migration/Index Reorganization Messages 109

Chapter 10. ARI0536–ARI0599, ARI5361–ARI5406 Preprocessor Messages 113

Chapter 11. ARI5500–ARI5593 FLAGGER Messages 131

Chapter 12. ARI0600–ARI0699, ARI6000–ARI6702, ARI0700–ARI0799 CMS EXEC Messages 141

Chapter 13. ARI7001–ARI7970 ISQL Messages 183

Chapter 14. ARI0800–ARI0899, ARI8000–ARI8999 DBS Utility Messages 223
Common Error Action 223
Rerunning DBS Utility 223

Chapter 15. ARI0900–ARI0949 Database Maintenance and Database I/O Messages 251

Chapter 16. ARI0950–ARI0987 Trace Formatter Messages 263

Chapter 17. VM Abend Codes 269

Chapter 18. DBSS Return Codes . . . 271
Warning Return Codes 271
Bad Return Codes 271

Chapter 19. SQLCODEs 277
SQLCODEs 0, and +12 through +863 277
SQLCODEs -084 through -30090 288

Chapter 20. SQLSTATEs 377

Chapter 21. DB2 Server for VM Reason Codes 399
VM System Services Return Codes 399
VM Abnormal Termination Codes 399
VM APPC/VM Communication Codes 400
VM IUCV Communication Codes 400
Codes Common to the DB2 Server for VM and Resource Adapter Virtual Machines 401
Codes of the DB2 Server for VM Virtual Machine 401
Codes of the Resource Adapter Virtual Machine 401
Codes Arising during Block I/O Operations . . 402

Chapter 22. Debugging Communication Errors 403
Possible Reasons for TCP/IP Errors 404

© Copyright IBM Corp. 1987, 2000

Possible Reasons for Communication Errors . . . 405

Chapter 23. Sequential File Errors . . 407
Problem Determination Notes:. 407

Chapter 24. Include File Access Errors 411
Problem Determination Notes:. 411

Chapter 25. Limit Errors 415

Chapter 26. System Errors 421

Chapter 27. Hardware Errors 423

Chapter 28. DPROPR MESSAGES . . 425
ASN0000E–ASN0203I DPROPR Capture Messages 425
. 425

Notices 431
Trademarks 432

Bibliography 433

Index 437

About This Manual

This manual lists the messages and codes issued by the DB2 Server for VM system. This reference manual is for users, operators, programmers, and maintainers of this product. This manual explains the messages you see on your display terminal.

The manual contains messages with the prefix ARI and ASN. The Data Propagator Relational Capture for VSE & VM messages are included in the manual and contain the ASN prefix.

The messages and codes, where appropriate, consist of:

- An expanded explanation of the message text
- The action taken by the system
- The response required of the user, operator, programmer, database administrator, and system programmer.

The codes section contains information for:

- ISQL VM abend codes
- DBSS return codes
- SQLCODEs
- SQLSTATEs
- Reason codes, including:
  - System service return codes
  - System abend codes
  - System communication codes
- Application Requester Communication Errors
- Sequential file errors
- Include file access errors
- Limit errors
- System errors
- Hardware errors.

The "Summary of Changes" section summarizes the technical and library changes made for the DB2 Server for VM Version 7 Release 1 product.

Chapters 17 through 21 explain the various codes and SQLSTATEs that can be returned to a DB2 Server for VM user or application program. Most of the codes and SQLSTATEs report error conditions. The codes are presented in groups, according to the source of the codes. The codes can come from the DB2 Server for VM system itself, or from various other programs with which the DB2 Server for VM system works.

If you are the system programmer or the database administrator, see the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for information on reporting a problem whenever you are asked in **System Programmer Response** or **Database Administrator Response** to contact your designated support group for your installation.

Throughout this manual, the DB2 Server for VM Version 7 Release 1 product is referred to as DB2 Server for VM, the database system, database manager or application server, unless otherwise noted. In some cases, the online help may use different terminology to that found in the manual.

© Copyright IBM Corp. 1987, 2000

For a list of related publications, see “Bibliography” on page 433.

Conventions Used in This Manual

In this manual, the following terms are used:

Term **Meaning**

DBS utility
Database Services Utility

ISQL Interactive Structured Query Language

LUW Logical unit of work

LUW ID
Logical unit of work identifier

saved segment
Saved segment, discontiguous saved segment, shared segment

saved segment ID
Parameter used in DB2 Server for VM EXECs to identify saved segments

DB2 Server for VM
DB2 Server for VM Version 7 Release 1

user ID
User identification

VM VM/ESA Release 2.2 and later

VTAM®
Virtual Telecommunications Access Method

Online HELP Text

Most of the information in this manual is also available as online HELP text. Installation of the online HELP text is optional. To use the online HELP text, when installed, enter the ISQL command HELP followed by a space and the message number or SQLCODE, or enter HELP SQLSTATE followed by the 5-digit SQLSTATE number.

Note: The online HELP text for messages, codes, and SQLSTATE information is not serviced by the IBM Support Center. The online HELP text is similar to this manual. You can use the Readers' Comment Form in the back of this manual to comment on the online HELP text. If you want to submit an Authorized Program Analysis Report (APAR) against documentation, it must be written against this manual, not the online HELP text.

Messages Issued from EXECs

DB2 Server for VM EXECs use the CMS XMITMSG command to issue messages. Because of an XMITMSG restriction, blanks cannot be imbedded in literal substitution strings. Therefore, a DB2 Server for VM EXEC may not be able to show the exact user input that caused an error. When this condition exists, an underscore (_) replaces each blank character in the substitution.

Components of the Relational Database Management System

Figure 1 on page vii depicts a typical configuration with one database and two users.

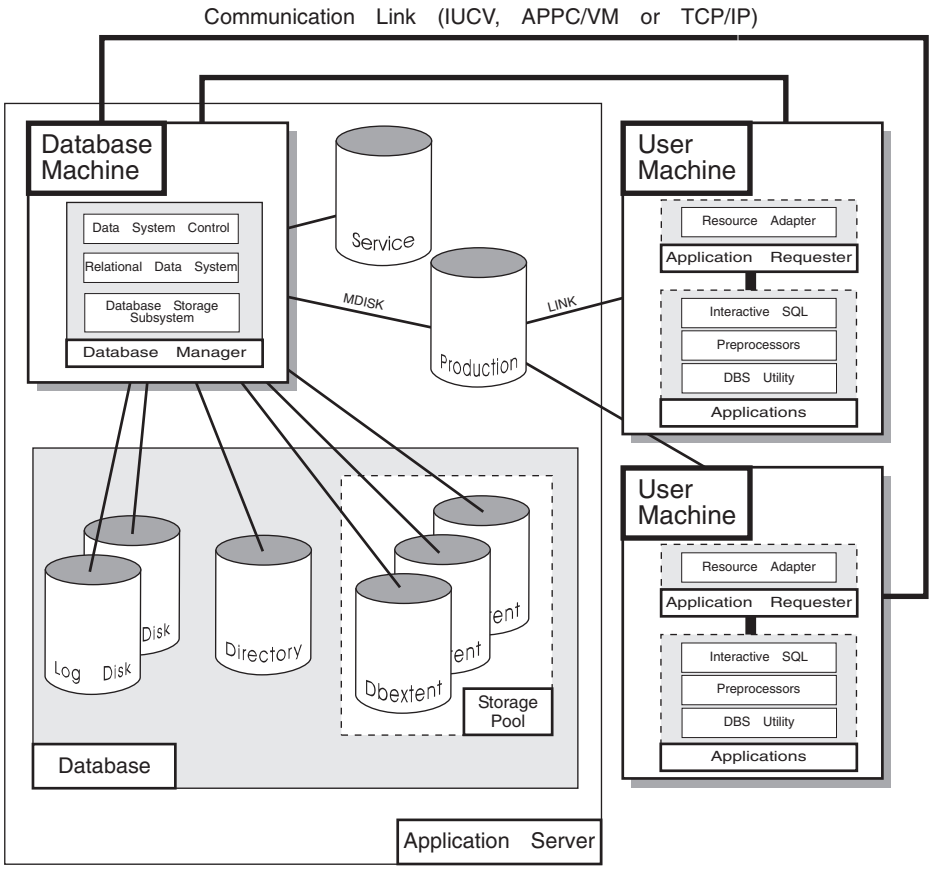


Figure 1. Basic Components of the RDBMS

The **database** is composed of:
- A collection of data contained in one or more *storage pools*, each of which in turn is composed of one or more *database extents (dbextents).* A dbextent is a VM minidisk.
- A *directory* that identifies data locations in the storage pools. There is only one directory per database.
- A *log* that contains a record of operations performed on the database. A database can have either one or two logs.

The **database manager** is the program that provides access to the data in the database. It is loaded into the database virtual machine from the production disk.

The **application server** is the facility that responds to requests for information from and updates to the database. It is composed of the database and the database manager.

The **application requester** is the facility that transforms a request from an application into a form suitable for communication with an application server.

SQL Reserved Words

The following words are reserved in the SQL language. They cannot be used in SQL statements except for their defined meaning in the SQL syntax or as host variables, preceded by a colon.

In particular, they cannot be used as names for tables, indexes, columns, views, or dbspaces unless they are enclosed in double quotation marks (").

ACQUIRE
ADD
ALL
ALTER
AND
ANY
AS
ASC
AVG

BETWEEN
BY

CALL
CHAR
CHARACTER
COLUMN
COMMENT
COMMIT
CONCAT
CONNECT
COUNT
CREATE
CURRENT

DBA
DBSPACE
DELETE
DESC
DISTINCT
DOUBLE
DROP

EXCLUSIVE
EXECUTE
EXISTS
EXPLAIN

FIELDPROC
FOR
FROM

GRANT
GRAPHIC
GROUP

HAVING

IDENTIFIED
IN
INDEX
INSERT
INTO
IS

LIKE
LOCK
LONG

MAX
MIN
MODE

NAMED
NHEADER
NOT
NULL

OF
ON
OPTION
OR
ORDER

PACKAGE
PAGE
PAGES
PCTFREE
PCTINDEX
PRIVATE
PRIVILEGES
PROGRAM
PUBLIC

RESOURCE
REVOKE
ROLLBACK
ROW
RUN

SCHEDULE
SELECT
SET
SHARE
SOME
STATISTICS
STORPOOL
SUM
SYNONYM

TABLE
TO

UNION
UNIQUE
UPDATE
USER

VALUES
VIEW

WHERE
WITH
WORK

How to Send Your Comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this book or any other DB2 Server for VSE & VM documentation:

- Visit our home page at:

  `http://www-4.ibm.com/software/data/db2/vse-vm/`

- A form for readers' comments is provided at the back of this publication. If the form has been removed, address your comments to:

  IBM CANADA LTD.
  DB2 Server for VSE & VM
  2S/240/1150/TOR
  1150 Eglinton Ave. East
  North York, Ontario
  Canada M3C 1H7

- Send your comments by electronic mail to one of the following addresses:

Format	Address
Internet	torrcf@ca.ibm.com
Facsimile	(416) 448-6161 (Attention RCF Coordinator)

  Be sure to include the name of the book, the form number (including the suffix), and the page, section title, or topic you are commenting on.

  If you choose to respond through the Internet, please include either your entire Internet network address, or a postal address.
- Fill out the form at the back of this book and return it by mail, by fax, or by giving it to an IBM representative.

Summary of Changes

This is a summary of the technical changes to the DB2 Server for VSE & VM database management system for this edition of the book. All manuals are affected by some or all of the changes discussed here. For your convenience, the changes made in this edition are identified in the text by a vertical bar (|) in the left margin. This edition may also include minor corrections and editorial changes that are not identified.

This summary does not list incompatibilities between releases of the DB2 Server for VSE & VM product; see either the *DB2 Server for VSE & VM SQL Reference*, *DB2 Server for VM System Administration*, or the *DB2 Server for VSE System Administration* manuals for a discussion of incompatibilities.

Summary of Changes for DB2 Version 7 Release 1

Version 7 Release 1 of the DB2 Server for VSE & VM database management system is intended to run on the Virtual Machine/Enterprise Systems Architecture (VM/ESA®) Version 2 Release 3 or later environment and on the Virtual Storage Extended/Enterprise Systems Architecture (VSE/ESA™) Version 2 Release 3 Modification 1 or later environment.

Enhancements, New Functions, and New Capabilities

TCP/IP Support for DB2 Server for VSE

TCP/IP support allows:

- VSE online and batch application programs to access remote application servers which support IBM's implementation of the DRDA architecture over TCP/IP.
- Remote application requesters which support IBM's implementation of the DRDA architecture to access the DB2 for VSE application server over TCP/IP.

For more information, see the following DB2 Server for VSE & VM documentation:

- *DB2 Server for VSE & VM Database Administration*
- *DB2 Server for VSE System Administration*
- *DB2 Server for VSE Program Directory*.

DRDA RUOW Application Requester for VSE (Batch)

DRDA Remote Unit of Work Application Requester provides read and update capability in one location in a single unit of work.

This support provides VSE batch application programs with the ability to execute SQL statements to access and manipulate data managed by any remote application server that supports IBM's implementation of the DRDA architecture.

VSE batch application programs can access only one remote application server per unit of work, and must use TCP/IP communications.

For more information, see the following DB2 Server for VSE & VM documentation:

- *DB2 Server for VSE System Administration*
- *DB2 Server for VSE & VM Database Administration*
- *DB2 Server for VSE & VM Application Programming*
- *DB2 Server for VSE Program Directory*.

© Copyright IBM Corp. 1987, 2000

Stored Procedures Application Requester

A stored procedure is a user-written application program compiled and stored at the server. Stored procedures allow logic to be encapsulated in a procedure that is local to the database manager. The ability to use stored procedures provides distributed solutions that let more people access data faster. SQL statements and replies flowing across the network are reduced and performance is improved.

This support provides VM and VSE (online and batch) application programs with the ability to invoke stored procedures from any remote application servers that support IBM's implementation of the DRDA architecture. It also allows processing of result sets if supported by the remote DRDA application server.

For more information, see the following DB2 Server for VSE & VM documentation:

- *DB2 Server for VSE & VM Application Programming*
- *DB2 Server for VSE & VM SQL Reference*.

Simplified DB2 Server for VSE Installation/Migration

A REXX procedure Job Manager is supplied to assist in the DB2 Server for VSE Installation/Migration process. It controls the overall job flow based on the contents of the job list control tables and the parameter table (supplied as Z-type members). The job manager selects the job control member from the job list file (a Z-type member), extracts the member from the Installation Library, modifies the JCL, submits the job, evaluates the execution, posts the results, and then repeats the process as required. The users are required to modify the parameter table, according to their environment.

This support simplifies the process of installation and migration by reducing user intervention - the Job Manager submits the prepared jobs.

See the *DB2 Server for VSE Program Directory* for further details.

New Code Page and Euro Symbol Code Page Support

The following CCSIDs are now supported:

- 1137: Hindi
- 1142: E-Danish/Norweigan
- 1143: E-Finnish/Swedish
- 1145: E-Spanish.

Additional support has been added for conversions from Unicode (UTF-8) to host CCSIDs.

For a complete list of CCSIDs supported, refer to the *DB2 Server for VM System Administration* and *DB2 Server for VSE System Administration* manuals.

Control Center for VM Enhancements

The following is a list of enhancements that have been made to the Control Center for VM:

- QMF™ Tools: allow the user to list QMF objects, view and unload QMF queries and PROCS, schedule QMF PROCS to execute, and run explain on QMF queries.
- Table Create Tool: allows the user to create new tables.
- Search List improvements.
- Referential Integrity Report tool: A referential integrity map report can now be generated directly from the CMS command interface.
- PL/I prerequisite removal.
- New and improved tape hopper support.
- High density tape drive support: support for high density (non-CMS density) tape drives.

Control Center for VSE Enhancements

The following enhancements have been provided for Control Center for VSE:

- Additional Operator Command Support

- Installation of IBM-provided Stored Procedures.

QMF for VSE & VM Optional Feature

The following enhancements have been provided for QMF for VSE & VM:

- Application Requester support for VSE QMF users
- Command enhancements to default to object type
- Fast path to the QMF home screen
- Cross-platform install capability
- DB2 for AS/400 database access.

QMF for Windows® Optional Feature

The following enhancements have been provided for QMF for Windows :

- Java-based Query
- Aggregating, grouping and formatting directly within query results and automatic Form creation
- Personal portal user interface that launches centrally shared queries and reports, and sends results to spreadsheets, desktop databases, and browsers
- Procedures with REXX.

Reliability, Availability, and Serviceability Improvements

DBNAME Directory Restructuring

ARISDIRD has been restructured to improve readability and flexibility. Each DBNAME entry is now defined explicitly by its type (Local, Remote or Host VM (Guest Sharing)). CICS AXE Transaction TPNs (Transaction Program Names) are still included in the directory as a type of 'LOCALAXE'. The DBNAME Directory Builder program, ARICBDID has been rewritten as a REXX/VSE procedure with extensive error and dependency checking. Support for TCP/IP information is added and 'alias' DBNAMEs are supported. **ALL** DBNAMEs **must** be specified in the new DBNAME Directory, including the Product Default DBNAME "SQLDS". A migration REXX/VSE procedure, ARICCDID, is provided to assist in migrating to the new format. See the *DB2 Server for VSE System Administration* and *DB2 Server for VSE Program Directory* for additional information.

Migration Considerations

Migration is supported from SQL/DS™ Version 3 and DB2 Server for VSE & VM Versions 5 and 6. Migration from SQL/DS Version 2 Release 2 or earlier releases is not supported. Refer to the *DB2 Server for VM System Administration* or *DB2 Server for VSE System Administration* manual for migration considerations.

Library Enhancements

Some general library enhancements include:

- The following books have been removed from the library:
  - *DB2 Server for VM Application Programming*
  - *DB2 Server for VSE Application Programming*
  - *DB2 Server for VM Database Administration*
  - *DB2 Server for VSE Database Administration*
  - *DB2 Server for VSE Installation*
  - *DB2 REXX SQL Interface Installation*
  - *DB2 REXX SQL Reference*
  - *DB2 Server for VM Diagnosis Guide and Reference*
  - *DB2 Server for VSE Diagnosis Guide and Reference*
  - *DB2 VM Data Spaces Support*

    Note: Information from this book can now be found in the *DB2 Server for VSE & VM Performance Tuning Handbook*
  - *DB2 Server for VM Master Index and Glossary*
  - *DB2 Server for VSE Master Index and Glossary*.

- The following books have been added to the library:
  - *DB2 Server for VSE & VM Database Administration*
  - *DB2 Server for VSE & VM Application Programming*
  - *DB2 REXX SQL for VM/ESA Installation and Reference*
  - *DB2 Server for VSE & VM Diagnosis Guide and Reference*
  - *DB2 Server for VSE & VM Master Index and Glossary*.

Refer to the new *DB2 Server for VSE & VM Overivew* for a better understanding of the benefits DB2 Server for VSE & VM can provide.

Chapter 1. Introduction

Using This Manual

If you use this manual regularly, you should read this section along with the following section "Message Format and Associated Text". They provide information that is helpful when you use this manual for reference.

All messages have four digits following the ARI or ASN prefix. Old 3-digit messages are now preceded by a zero. For example, ARI201E is now ARI0201E.

The table of contents can help you in looking up a particular message, code, or error because headings are provided to show the numeric sections and page locations. You can also use the index at the back of the book to find the page or pages of any particular message or code.

Message Format and Associated Text

In this book, the conventions for showing messages are:

- All the letters, numbers, and punctuation marks (periods, commas, underscores, dashes, and so on) following the message identifier in bold show the actual message text. The exceptions to this convention are brackets [ ], braces { }, logical OR symbols |, blank lines, and format numbers.
- Letters, words, and numbers in the text of the message that are not bold represent variables. Hyphenated words that are not bold also represent variables. The database manager substitutes specific values for the variables when it displays the message.

  In the explanation following the message text, the variables are in bold or italics.
- Brackets [ ] enclose optional message text. Depending on the circumstances, an item or group of items in brackets might appear when the message is displayed.
- Braces { } enclose alternative items, which are separated by a logical OR | symbol. Depending on the circumstances, one of the alternative items appears when the message is displayed.

All DB2 Server for VM messages have a message identifier in the form of **XXXnnnnt**, where:

XXX Identifies the major component generating the message:

- **ARI** shows the message is from the database manager component of the DB2 Server
- **ASN** shows the message is from the DPROPR component
- **DSQ** see *QMF Messages and Codes* manual
- **XTS** shows the message is from the Data Restore Feature.

nnnn Identifies the particular message number.

t Is the action indicator:

- **A** Action required
- **D** Decision required
- **E** Error
- **I** Information only
- **W** System wait.

For messages beginning with:

XTS See the *DB2 Server for VSE & VM Data Restore* manual

SQM See the *DB2 for VM Control Center Operations Guide*.

ELO See the *DB2 REXX SQL for VM/ESA Installation and Reference* manual

Explanation contains information about the message. This includes the following:

- Definitions for variable fields in the message text
- Possible reasons why the message occurred
- Additional descriptive information about the condition that caused the message to occur.

System Action describes the resulting action that was or will be taken by the database manager.

User Response, **Operator Response**, **Programmer Response**, **Database Administrator Response**, and **System Programmer Response** describe the responses that might be required of those persons.

© Copyright IBM Corp. 1987, 2000

Introduction

Chapter 2. ARI0001–ARI0198 DB2 Server for VM Control Messages

ARI0001E **An error occurred while attempting to obtain storage.**

Explanation: Either:

- All available free storage has been exhausted; the reason code, normally displayed during termination, will be 01 (X'01')
- An internal error occurred while attempting to get storage; the reason code will be the VM return code.

Note: The DB2 Server for VM reason code is the CMS free storage macro return code. It is displayed in message ARI0042I.

System Action: If SQLCODE -930 accompanies this message, the database manager continues and additional requests to obtain storage may result in the message being issued again. If the message is issued without SQLCODE -930, the database manager is ended.

Operator Response: If free storage has been exhausted (reason code = 01), restart the database manager after allocating a larger virtual machine.

System Programmer Response: You may have to assist the operator in determining what virtual machine size will be required if storage is exhausted. Contact the designated support group for errors.

ARI0002E **Unable to open the** *xxxxxxxx* **parameter data set. Reason Code =** *nn***. Return Code=***nn***.**

Explanation: *xxxxxxxx* is the name of the parameter data set as specified in the PARMID parameter.

This parameter is in the SQLSTART EXEC command to identify a card image CMS file having file type SQLPARM and file mode *.

The Reason Code value is the Primary Error Code. The Return Code value is the Secondary Error Code. For further explanation, look up the displayed error codes under either "Chapter 24. Include File Access Errors" on page 411, or "INCLUDE FILE ERRORS" in the HELP text.

System Action: The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Either:

- Create the required CMS file, or
- Correct the PARMID parameter in the SQLSTART EXEC command.

If the problem persists, for corrective action, look up the displayed error codes under either "Chapter 24. Include File Access Errors" on page 411 or "INCLUDE FILE ERRORS" in the HELP text.

ARI0003E **An error occurred reading the** *xxxxxxxx* **parameter data set. Reason Code =** *nn* **Return Code =** *nn***.**

Explanation: *xxxxxxxx* is the name of the parameter data set as specified in the PARMID parameter. This parameter is in the SQLSTART EXEC command to identify a card image CMS file having file type SQLPARM and file mode *.

The REASON value is the Primary Error Code. The CODE value is the Secondary Error Code. For further explanation, look up the displayed error codes under either "Chapter 24. Include File Access Errors" on page 411, or "INCLUDE FILE ERRORS" in the HELP text.

System Action: The database manager ends.

Operator Response: For further explanation, look up the displayed error codes under either "Chapter 24. Include File Access Errors" on page 411, or "INCLUDE FILE ERRORS" in the HELP text.

ARI0004I *xxxxxxxx* **- The database manager ignores PARMID within a parameter data set.**

Explanation: *xxxxxxxx* is the name of the parameter data set which contains the PARMID initialization parameter. The PARMID parameter is ignored in a parameter data set.

System Action: Processing continues.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the parameter data set specification to eliminate this message.

ARI0005E **{CMD | *xxxxxxxx*} -** *yyyyyyyy* **parameter is specified more than once.**

Explanation: **CMD** indicates that the parameter in error was specified in the SQLSTART EXEC.

xxxxxxxx is the name of the parameter data set which contains the parameter in error.

yyyyyyyy is the parameter keyword that was specified more than once in the initialization parameter input.

© Copyright IBM Corp. 1987, 2000

Note: The same parameter keyword can be specified once in the SQLSTART EXEC and once in the parameter data set.

System Action: If this error occurs in the SQLSTART EXEC parameter processing, the database manager ends.

If this error occurs during parameter data set processing, the database manager ends unless the *yyyyyyyy* parameter is correctly specified in the SQLSTART EXEC. See System Programmer Response.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Remove the extra occurrences of the initialization parameter from either the SQLSTART EXEC or the parameter data set. If this parameter was specified more than once in the parameter data set, it may be overridden by a correct specification in the SQLSTART EXEC. However, the message will still be issued but may be ignored and the database manager will continue processing.

ARI0006E **{CMD | *xxxxxxxx*} - *yyyyyyyy* is an invalid parameter specification.**

Explanation: **CMD** indicates that the parameter in error was specified in the SQLSTART EXEC.

xxxxxxxx is the name of the parameter data set which contains the parameter in error.

yyyyyyyy is the initialization parameter keyword which is invalid (or misspelled). If the parameter keyword in error was greater than eight characters, only the first nine characters are printed.

System Action: The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the invalid initialization parameter specification in either the SQLSTART EXEC or the parameter data set.

ARI0007E **{CMD | *xxxxxxxx*} error. A keyword is missing.**

Explanation: **CMD** indicates that the parameter in error was specified in the SQLSTART EXEC.

xxxxxxxx is the name of the parameter data set which contains the parameter in error.

System Action: The database manager is ended.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the invalid initialization parameter specification in either the SQLSTART EXEC or the parameter data set. This error would be caused by coding ...,=value,... .

ARI0008E **{CMD | OPER | *xxxxxxxx*} - *yyyyyyyy* = *c* is invalid. It must be *d***

Explanation: **CMD** indicates that the parameter in error was specified in the SQLSTART EXEC.

OPER indicates that the parameter in error was specified during run time using the **SET** operator command.

xxxxxxxx is the name of the parameter data set which contains the parameter in error.

yyyyyyyy is the initialization parameter keyword which has the invalid parameter value. It must be one of the following keywords: ACCOUNT, DBMODE, DSPSTATS, DSPLYDEV, DUALLOG, DUMPTYPE, EXTEND, LOGMODE, MAPPING, PROTOCOL, SECALVER, SECTYPE, STARTUP, SYSMODE, SEPINTDB, or SYNCPNT.

c is the first (or only) character of the invalid parameter value. If no value was specified, a comma is substituted for the missing value. If more than one character is given for the parameter value, it will not be accepted even if the first character is valid.

d is a character string indicating the correct parameter value as follows:

yyyyyyyy	d	(default)
ACCOUNT	D or N	N
DBMODE	G, L, or N	Value in CP Directory
DSPLYDEV	C	C
DSPSTATS	0, 1, or 2	0
DUALLOG	N or Y	N
DUMPTYPE	P, F, or N	F
EXTEND	Y or N	N
LOGMODE	Y, A, L, or N	Y
MAPPING	L or P	L
PROTOCOL	SQLDS or AUTO	SQLDS
SECALVER	N or Y	N
SECTYPE	DB2 or ESM	DB2
SEPINTDB	N or Y	see note below
STARTUP	C, E, I, L, M, R, S, U, or W	W
SYNCPNT	Y or N	Y
SYSMODE	M or S	M

Note: When the DB2 DSS feature is installed and the database machine is in XC mode, then the default value for SEPINTDB is Y, otherwise the default value is N.

System Action: If this error occurs in the SQLSTART EXEC parameter processing, the database manager ends.

If this error occurs during parameter data set processing, the database manager ends unless the *yyyyyyyy* parameter is correctly specified in the SQLSTART EXEC. See System Programmer Response.

If this error occurs while processing the **SET** command, the command is ignored and processing continues.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the invalid parameter value in the SQLSTART EXEC, the parameter data set, or the operator console. If the invalid parameter was specified in the parameter data set, it may be overridden by a correct specification in the SQLSTART EXEC. However, the message will still be issued but may be ignored and the database manager will continue processing.

ARI0009E **{CMD | OPER | *xxxxxxxx*} - *yyyyyyyy* = *c* is invalid. It must be in the range from *n1* to *n2*.**

Explanation: **CMD** indicates that the parameter in error was specified in the SQLSTART EXEC.

OPER indicates that the parameter in error was specified during run time using the **SET** operator command.

xxxxxxxx is the name of the parameter data set which contains the parameter in error.

yyyyyyyy is the parameter keyword which has the invalid parameter value. It must be one of the following keywords: ARCHPCT, CHKINTVL, DISPBIAS, LTIMEOUT, NCSCANS, NCUSERS, NDIRBUF, NLRBS, NLRBU, NPACKAGE, NPACKPCT, NPAGBUF, PROCMXAB, PTIMEOUT, SAVEINTV, SLOGCUSH, SOSLEVEL, TARGETWS, TCPPORT, or TRACEBUF.

c is the invalid parameter value. If no value was specified, a comma is substituted for the missing value. If the parameter is longer than eight characters, only the first eight characters are displayed. The parameter value can be up to eight characters. Leading zeros are suppressed.

n1 is the lower limit (or minimum) for the correct parameter value.

n2 is the upper limit (or maximum) for the correct parameter value.

yyyyyyyy	n1	n2	(default)
ARCHPCT	10	99	80
CHKINTVL	1	99999999	10
DISPBIAS	1	10	7
LTIMEOUT	0	99999	0
NCSCANS	1	655	30
NCUSERS	1	252	5
NDIRBUF	10	400000	NPAGBUF
NLRBS	Larger of 50 or (2 x NCUSERS)	583333	(2 x NCUSERS) + (NLRBU x NCUSERS)/2 +10
NLRBU	10	583233	1000
NPACKAGE	1	32766/NCUSERS	10
NPACKPCT	0	100	30
NPAGBUF	10	400000	(4 x NCUSERS) + 10
PROCMXAB	0	255	0
PTIMEOUT	0	99999	180
SLOGCUSH	11	100	90
SOSLEVEL	1	100	10
TARGETWS	1	99999	32
TCPPORT	0	65535	ETC SERVICES
TRACEBUF	0	99999	0

System Action: If this error occurs in the SQLSTART EXEC parameter processing or during parameter data set processing, the database manager ends.

If this error occurs while processing the **SET** operator command, the command is ignored and processing continues.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the invalid parameter value in the SQLSTART EXEC, the parameter data set, or the operator console. If the invalid parameter was specified in the parameter data set, it may be overridden by a correct specification in the SQLSTART EXEC. However, the message will still be issued but may be ignored and the database manager will continue processing.

ARI0010E **{CMD | *xxxxxxxx*} - {TRACDBSS | TRACCONV | TRACRDS | TRACDSC | TRACWUM | TRACDRRM | TRACSTG} is invalid. The parameter value must be either {0, 1, or 2 | 0 or 1}.**

Explanation: **CMD** indicates that the parameter in error was specified in the SQLSTART EXEC.

xxxxxxxx is the name of the parameter data set which contains the parameter in error.

Specifying a parameter value other than 0, 1, or 2 for the TRACDBSS, TRACCONV, TRACRDS, TRACDSC, TRACWUM, or TRACDRRM initialization parameter is invalid. Specifying a parameter value other than 0 or 1 for the TRACSTG initialization parameter is invalid.

System Action: If this error occurs in the SQLSTART EXEC parameter processing, the database manager ends. If this error occurs during parameter data set processing, the database manager ends unless the TRACDBSS, TRACCONV, TRACRDS, TRACDSC, TRACWUM, TRACDRRM, or TRACSTG parameter is correctly specified in the SQLSTART EXEC. See System Programmer Response.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the invalid initialization parameter value in either the SQLSTART EXEC or the parameter data set. If the invalid parameter was specified in the parameter data set, it may be overridden by a correct specification in the SQLSTART EXEC. However, the message will still be issued but may be ignored and the database manager will continue processing.

ARI0011E **{CMD | *xxxxxxxx*} - *yyyyyyyy* is invalid. The parameter string must be from *1* to *n* characters.**

Explanation: **CMD** indicates that the parameter in error was specified in the SQLSTART EXEC.

xxxxxxxx is the name of the parameter data set which contains the parameter in error.

yyyyyyyy is the initialization parameter keyword which has the invalid parameter string length. It must be one of the following keywords: CHARNAME, DBNAME, DSPSTATS, PARMID, PROGNAME, PROTOCOL, TRACCONV, TRACDBSS, TRACRDS, TRACDSC, TRACWUM, TRACDRRM, or TRACSTG.

n is the maximum character string length of the value.

yyyyyyyy	n
CHARNAME	18
DBNAME	18
DSPSTATS	2
PARMID	8
PROGNAME	8
PROTOCOL	5
TRACCONV	1
TRACDBSS	11
TRACRDS	6
TRACDSC	2
TRACWUM	1
TRACDRRM	4
TRACSTG	1

Note: PROTOCOL also has a minimum length of 4.

System Action: If this error occurs in the SQLSTART EXEC parameter processing, the database manager ends.

If this error occurs during parameter data set processing, the database manager ends unless the *yyyyyyyy* parameter is correctly specified in the SQLSTART EXEC. See System Programmer Response.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the invalid initialization parameter value in either the SQLSTART EXEC or the parameter data set. If the invalid parameter was specified in the parameter data set, it may be overridden by a correct specification in the SQLSTART EXEC. However, the message will still be issued but may be ignored and the database manager will continue processing. PARMID is valid only on the SQLSTART EXEC and, therefore, must always be corrected.

ARI0012I **The database manager ignores *yyyyyyyy* = *c* when *zzzzzzzz* = *d*.**

Explanation: *yyyyyyyy* is the initialization parameter keyword of the value that is ignored.

c is the parameter value that was specified.

zzzzzzzz is the initialization parameter keyword that is used and causes the *yyyyyyyy* parameter to be ignored.

d is the parameter value specified for *zzzzzzzz*.

yyyyyyyy	c	zzzzzzzz	d
ACCOUNT	D	STARTUP	C, E, I, L, M, or S
ACCOUNT	D	PROGNAME	ARISEGB
ACCOUNT	D	PROGNAME	ARISRMG
ARCHPCT	value	LOGMODE	Y or N
DBMODE	G, L, or N	SYSMODE	S
DISPBIAS	value	SYSMODE	S
DUALLOG	N or Y	SYSMODE	M
DUALLOG	N or Y	STARTUP	E, I, M, R, S, U, or W
NCUSERS	value > 1	SYSMODE	S
PROCMXAB	non-zero value	SYSMODE	S
PROGNAME	program-name	SYSMODE	M
PROGNAME	program-name	STARTUP	C, E, I, L, M, or S
PROTOCOL	SQLDS or AUTO	SYSMODE	S
PTIMEOUT	non-zero value	SYSMODE	S
TCPPORT	non-zero value	SYSMODE	S
TRACDRRM	non-zero value	PROTOCOL	SQLDS
TRACWUM	1 or 2	PROTOCOL	SQLDS
SYNCPNT	Y	PROTOCOL	SQLDS
SECALVER	N or Y	SYSMODE	S
SECTYPE	DB2 or ESM	SYSMODE	S
LTIMEOUT	non-zero value	SYSMODE	S
SYNCPNT	Y	AMODE	24

Notes:

1. The database manager ignores the PROTOCOL parameter when SYSMODE=S because it has no significance in single user mode. It is supported only in multiple user mode.
2. The database manager ignores the PROCMXAB and PTIMEOUT parameters when SYSMODE=S because they have no significance in single user mode. They are supported only in multiple-user mode.
3. The database manager ignores the TRACDRRM and TRACWUM parameters when PROTOCOL=SQLDS because the DRRM and WUM components are available only when PROTOCOL=AUTO.
4. The database manager ignores the TCPPORT parameter when SYSMODE=S because it has no significance in single user mode. It is supported only in multiple-user mode.

System Action: Processing continues.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Remove the initialization parameter specification causing the message.

ARI0013E *yyyyyyyy* **must be** *c* **when** *zzzzzzzz* **=** *d* **.**

Explanation: *yyyyyyyy* is the initialization parameter keyword of the value that is in conflict.

c is the parameter value that was specified.

zzzzzzzz is the initialization parameter keyword that is used and causes the *yyyyyyyy* parameter specification to be invalid.

d is the parameter value specified for *zzzzzzzz*.

yyyyyyyy	c	zzzzzzzz	d
LOGMODE	Y	STARTUP	L
LOGMODE	Y, A, or L	SYSMODE	M
NETID	specified	PROTOCOL	AUTO
PROTOCOL	AUTO	RESID	Registered DRDA TPN
STARTUP	W, R, or U	EXTEND	Y
STARTUP	W, R, or U	SYSMODE	M

Notes:

1. DBMODE must be G or L when PROTOCOL is AUTO because DBMODE = N implies the use of IUCV communications, whereas PROTOCOL=AUTO implies the use of APPC/VM communications.
2. PROTOCOL must be AUTO when RESID is a registered DRDA TPN because the server must provide all the capabilities associated with DRDA support.
3. NETID must be specified when PROTOCOL is AUTO because the name of the host network is required for network problem management.

System Action: If this error occurs in the SQLSTART EXEC parameter processing, the database manager ends.

If this error occurs during parameter data set processing, the database manager ends unless the *yyyyyyyy* parameter is correctly specified in the SQLSTART EXEC. See System Programmer Response.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the invalid initialization parameter value in either the SQLSTART EXEC or the parameter data set. If the invalid parameter was specified in the parameter data set, it may be overridden by a correct specification in the SQLSTART EXEC. However, the message will still be issued but may be ignored and the database manager will continue processing.

ARI0014E **The database manager requires PROGNAME when SYSMODE = S AND STARTUP = {W|R}.**

Explanation: When SYSMODE=S and STARTUP=W (or STARTUP=R) is specified, the PROGNAME initialization parameter must be specified.

System Action: If this error occurs in the SQLSTART EXEC parameter processing, the database manager ends. If this error occurs during parameter data set processing, the database manager ends unless the PROGNAME or SYSMODE or STARTUP parameter is correctly specified in the SQLSTART EXEC. See System Programmer Response.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Either specify the PROGNAME initialization parameter or correct the invalid parameter value (SYSMODE or STARTUP) in either the SQLSTART EXEC or the parameter data set that is causing this message. If this parameter was specified more than once in the parameter data set, it may be overridden by a correct specification in the SQLSTART EXEC. However, the message will still be issued but may be ignored and the database manager will continue processing.

ARI0015I *yyyyyyyy* **parameter value is** *c***.**

Explanation: This message displays the final values used by the database manager as a result of the merging of SQLSTART EXEC, parameter data set, and default parameter values.

yyyyyyyy is the initialization parameter keyword.

c is the parameter value to be used by the database manager.

yyyyyyyy	c
ACCOUNT	D or N
CHARNAME	character-set-name
DBNAME	database name, up to 18 characters
DBMODE	G, L, or N
DSPLYDEV	C
DSPSTATS	0, 1, or 2
DUALLOG	N or Y
DUMPTYPE	P, F, or N
EXTEND	Y or N
LOGMODE	Y, A, L, or N
NETID	network identifier
PARMID	parameter-data-set-name
PROGNAME	program-name
PROTOCOL	SQLDS or AUTO
RESID	hexadecimal/character TPN
SECALVER	Y or N
SECTYPE	DB2 or ESM
STARTUP	C, E, I, L, M, R, S, U, or W
SYSMODE	M or S
TRACCONV	0, 1, or 2
TRACDBSS	0, 1, or 2

```
TRACRDS    0, 1, or 2
TRACDSC    0, 1, or 2
TRACWUM    0, 1, or 2
TRACDRRM   0, 1, or 2
TRACSTG    0 or 1
SEPINTDB   Y or N
MAPPING    L or P
```

Notes:

1. The NETID parameter value must be from 1 to 8 characters long. NETID is a character string comprised of alphabetic (uppercase), numeric, and special characters #, @, $. The first character must not be numeric. Its length can be from 1 to 8 bytes.
2. The RESID parameter value must be from 1 to 8 characters long. It must belong to one of the following categories:
   - A string starting with an alphabetic character and followed by a sequence of alphabetic, numeric, and underscore characters.
   - A registered DDM TPN (refer to SNA Formats)
   - A registered DRDA TPN (refer to the *Distributed Relational Database Architecture Reference* manual).

System Action: Processing continues.

Notes:

1. DUALLOG will be displayed only if STARTUP=C or STARTUP=L.
2. PARMID will be displayed only when specified.
3. PROGNAME will be displayed only when SYSMODE=S and STARTUP=W, R or U.
4. DBMODE will be displayed only when specified.
5. SEPINTDB and MAPPING will be displayed only when the database manager is using data spaces.

System Programmer Response: You may want to verify that the initialization parameter values are correct.

ARI0016I *yyyyyyyy* **parameter value is** *n*.

Explanation: This is an information message that displays the final values used by the database manager as a result of the merging of SQLSTART EXEC, parameter data set, and default parameter values.

yyyyyyyy is the initialization parameter keyword.

n is the parameter value to be used by the database manager.

```
yyyyyyyy   n
---------  --------------
ARCHPCT    10 to 99
CHKINTVL   1 to 99999999
DISPBIAS   1 to 10
NCSCANS    1 to 655
NCUSERS    1 to 252
NDIRBUF    10 to 400000
NLRBS      50 to 583333
NLRBU      10 to 583233
NPACKAGE   1 to 32766/NCUSERS
NPACKPCT   0 to 100
NPAGBUF    10 to 400000
PROCMXAB   0 to 255
PTIMEOUT   0 to 99999
SLOGCUSH   11 to 100
SOSLEVEL   1 to 100
SAVEINTV   1 to 999999
TARGETWS   1 to 99999
TCPPORT    0 to 65535
TRACEBUF   0 to 99999
```

System Action: Processing continues.

Notes:

1. The ARCHPCT parameter is displayed only if LOGMODE=A.
2. The SAVEINTV and TARGETWS parameters are displayed only when the database manager is using data spaces for its storage pools.

System Programmer Response: You may want to verify that the initialization parameter values are correct.

ARI0017E **An error occurred in the parameter specification.**

Explanation: One or more errors were found while processing the initialization parameters. Previous messages indicate the parameter in error and the correction.

System Action: The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the invalid initialization parameter value in either the SQLSTART EXEC or the parameter data set. If the invalid parameter was specified in the parameter data set, it may be overridden by a correct specification in the SQLSTART EXEC.

ARI0018I **{CMD|***xxxxxxxx***} - {DBPSWD|DSPLYDEV} parameter is ignored for VM.**

Explanation: **CMD** indicates that the parameter in error was specified in the SQLSTART EXEC.

xxxxxxxx is the name of the parameter data set which contains the parameter in error.

System Action: Processing continues. The specified keyword is ignored. DSPLYDEV is always set to C (console).

Operator Response: Refer this message to your system programmer.

System Programmer Response: You may want to remove the specification of DBPSWD or DSPLYDEV to prevent the issuance of this message in the future.

ARI0019E DBNAME parameter must be specified for VM.

Explanation: The DBNAME parameter must be specified when the application server is started with SYSMODE=M. (This is the default.)

System Action: The database manager is ended.

System Programmer Response: Re-execute the SQLSTART EXEC with the DBNAME parameter specified. Ensure that any EXEC that may invoke the SQLSTART EXEC specifies the DBNAME parameter. If the application server is being started by a means other than the SQLSTART EXEC, ensure that the DBNAME parameter is specified in the input parameters. For example:

```
PARM ( DBNAME=server-name,... )
```

ARI0020I Virtual machine addressing mode = *amode*. Virtual machine storage size = *machine-size*KB

Explanation: The application server is starting up and will run in addressing mode *amode*, where *amode* is either 24 or 31. The application server can run AMODE 24 or 31 in a virtual machine running in XA or XC mode and can run AMODE 24 in a virtual machine which is in 370 mode. The virtual machine size of the database machine is *machine-size* kilobytes.

System Action: Processing continues.

System Programmer Response: If the addressing mode or virtual machine size is not as desired, issue SQLEND to stop the application server.

If you wish to switch to AMODE 24, you can take one of the following actions.

- Use the CP SET MACHINE 370 command to change the operating mode of the virtual machine.

  or
- Specify the AMODE(24) option when running SQLSTART. If you do this, the application server will continue to start up in AMODE 24 until AMODE(31) is next specified when running SQLSTART.

If you wish to switch to AMODE 31, ensure that your virtual machine is operating in XA or XC mode, and use the AMODE(31) startup parameter when running the SQLSTART EXEC. If you do this, the application server will continue to start up in AMODE 31 unless AMODE(24) is used during a future startup of the application server, or the operating mode is reset to 370.

To change the storage size of the virtual machine, use the CP DEFINE STORAGE command.

ARI0021E The *program-name* program was loaded above 16MB. The program must be RMODE 24 when running with the AMODE(24) option.

Explanation: The *program-name* program must be loaded below 16MB when the application server is running AMODE 24.

System Action: The application server is ended.

System Programmer Response: Refer to the *DB2 Server for VM System Administration* manual for information on how your single-user-mode applications are loaded and ensure that your application will be loaded below the 16MB line.

If there is no longer any need for the application server to run AMODE 24, (i.e. all your user exits and single-user-mode applications support 31-bit addressing), then you can use the AMODE(31) parameter when running SQLSTART. AMODE 31 will then become the default addressing mode for the application server and your single-user-mode application can be loaded above the 16MB line.

ARI0022E Field procedure *fieldproc-name* has abnormally terminated. Please remove it from your installation.

Explanation: The field procedure *fieldproc-name* has abended. You must remove the procedure from the system before restarting the system. Executing field procedures that contain errors can affect the security and integrity of the database. Thoroughly test the field procedure before reinstalling it on your system.

System Action: The application server will produce a mini-dump. When possible, processing will continue.

Operator Response: Save the console output for the system programmer.

System Programmer Response: Remove the field procedure from your system before restarting the database manager. Ensure that the field procedure is thoroughly tested before reinstalling it.

ARI0023I The *program-name* program will be invoked in AMODE *amode*.

Explanation: The *program-name* program will be started in the addressing mode *amode* specified. This could be either 24-bit or 31-bit addressing. Note that this addressing mode corresponds to the addressing mode in which the application server is started, unless the program is a DBS utility or a DB2 Server for VM preprocessor. In those instances, the addressing mode is always AMODE 24. This message is issued before invoking the single-user-mode application *program-name*.

System Action: Processing continues.

System Programmer Response: If the addressing

mode is not what you want, refer to the *DB2 Server for VM System Administration* manual for information on running applications in single-user mode.

ARI0024E **DBCS conversion procedure** *transproc-name* **has abnormally terminated. Please remove it from your installation.**

Explanation: The DBCS conversion procedure *transproc-name* has abended. Thoroughly test the procedure before re-installing it on your system. If you want to replace or remove this procedure from your system, contact your system programmer.

System Action: The application server will produce a mini-dump. When possible, processing will continue.

Operator Response: Save the console output for the system programmer.

System Programmer Response: Determine the CCSID conversions that use this DBCS conversion procedure by checking the TRANSPROC column of the catalog table SYSTEM.SYSSTRINGS. Refer to the *DB2 Server for VM System Administration* manual for information on updating these CCSID conversions to make use of another DBCS conversion routine.

ARI0025I **The program** *program-name* **is loaded at** *load-address***.**

Explanation: *program-name* is the name of the load module loaded by by the database manager.

load-address is the storage address (expressed in hexadecimal) where the load module was loaded.

System Action: Processing continues.

System Programmer Response: You may want to note the address for use in any potential problem determination.

ARI0026E **An error occurred in the attempt to load program** *xxxxxxxx***.**

Explanation: *xxxxxxxx* is the name of the load module that the application server attempted to load.

The application server attempts to load the application program by following this sequence:

1. If the program was not loaded, use the CMS LOADMOD command.
2. If the program was not loaded in the previous step, use the CMS NUCXLOAD command (this assumes the program may be in a CMS LOADLIB).
3. If the saved segment was not loaded, use the CP SEGMENT LOAD function.
4. If the application server was started with the SET DOS ON command, use the CMS/DOS FETCH function.
5. If the program was not loaded, use the CMS LOAD command.

If the program was not loaded, the reason code is:

4 Not found
8 Insufficient storage
12 Other error.

System Action: The database manager is ended.

Operator Response: If the reason code indicates insufficient storage, restart the application server after allocating a larger virtual machine. For all other reason codes, notify your system programmer.

System Programmer Response:

Ensure that the program runs as a VM load module, in a CMS LOADLIB, a DOSLIB (CMS/DOS), or a TXTLIB, or has been installed as a saved segment.

ARI0027E **A communication link error has occurred.**

Explanation: The database manager has violated the use of the IUCV or APCC/VM services. The most probable causes may be:

- Control blocks related to the use of communication services were damaged.
- Registers or register save areas used by the database manager in performing communication functions may have had their contents altered.

(See "Chapter 21. DB2 Server for VM Reason Codes" on page 399.)

System Action: The database manager is ended.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0028I **The database manager is terminating.**

Explanation: This informational message is issued as acknowledgment when an SQLEND command (other than SQLEND QUICK) is first entered.

Note: SHUTDOWN is a valid synonym for SQLEND.

System Action: The database manager will start termination procedures; all active users (DB2 Server for VM agent structures) will be allowed to complete their processing. No new users will be allowed access to database services. After all active users have completed their processing, the database manager will perform any optional functions specified on the SQLEND command and end.

ARI0029I *n* **communication links are still connected.**

Explanation: This informational message tells the operator how many users are active when the SQLEND command was entered. This number is the number of pseudo agent structures containing connected users.

This message is issued when an SQLEND command (other than SQLEND QUICK) is first issued. It is also issued with an updated number as each communication link is disconnected. This may allow the operator to determine how long it might take to complete the SQLEND process.

Note: SHUTDOWN is a valid synonym for SQLEND.

System Action: Processing continues.

ARI0030I **Database manager termination is already in progress.**

Explanation: This informational message is issued in response to a subsequent SQLEND command (other than SQLEND QUICK).

System Action: Processing continues and the command is ignored.

Operator Response: None unless an immediate shutdown is required, in which case an SQLEND QUICK command can be issued. If that command is issued, any logical units of work in progress will be resolved (either committed or backed out) the next time the database manager is brought up.

Note: SHUTDOWN is a valid synonym for SQLEND.

ARI0031E **Invalid SQLEND parameter.**

Explanation: The SQLEND command was entered with an invalid parameter. The operator should specify only a valid SQLEND command. For more information, refer to the *DB2 Server for VSE & VM Operation* manual.

Note: SHUTDOWN is a valid synonym for SQLEND.

System Action: Operator command processing ends.

Operator Response: Reenter the command correctly.

ARI0032I **The database manager has terminated.**

Explanation: This informational message is issued prior to closing any data sets and ending.

Note: The virtual machine operator should not enter an HX command after this message has been displayed. It will cause immediate termination of the database manager with the message ARI0034E being displayed.

Note: If the database manager has ended abnormally, a DB2 Server for VM mini-dump will be displayed. These dumps are described in the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual.

System Action: The database manager is ended.

ARI0034E **An error occurred during termination.**

Explanation: This message is issued whenever database manager termination is in progress and an error (for example: a program check) occurs which causes the termination process to be invoked. This message is also issued if the operator enters a HX command after message ARI0032I has been displayed.

Note: Tape files may not have been closed and may be missing the last buffer contents and have no tape mark (EOF indicator).

System Action: The database manager ends immediately.

Operator Response: Notify your system programmer. You may wish to write a tapemark on any open tape files.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0035I **Application server cancel has been requested.**

Explanation: The virtual machine operator has issued an HX command. The operator is allowed to request a dump if desired. See message ARI0044D.

System Action: The database manager ends.

ARI0036E **No SAVEAREA found for ARIXRDS.**

Explanation: A program check interrupt occurred in the package. The database manager attempted to recover, but could not find the save area for the RDS recovery routine (ARIXEBR).

System Action: The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0037E The entry point for ARIXEBR1 cannot be found.

Explanation: A program check interrupt occurred in the package. The database manager attempted to recover, but could not find the entry point to the RDS recovery routine (ARIXEBR1).

System Action: The database manager is ended.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0038E Invalid return code from ARI*xxxx*.

Explanation: An internal error has occurred.

ARI*xxxx* is the name of the module which returned an unexpected or invalid return code to the caller.

System Action: The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0039E Application server limit error occurred - ARI*xxxx nn*.

Explanation: This message is usually due to causes such as:

- Insufficient virtual machine free storage.
- Module not found.
- Insufficient external storage space for functions such as DB2 Server for VM logging.

Other messages issued prior to this message will identify the error condition and aid in the corrective action to be taken.

ARI*xxxx* is the name of the module which detected the limit error.

nn is the error detection point within that module. See "Chapter 25. Limit Errors" on page 415 for a list of critical detection points and their related error conditions.

No DB2 Server for VM dump or mini-dump will be displayed because a limit error occurred.

System Action: The database manager is ended.

Operator Response: If the error is due to insufficient free storage, restart the database manager after allocating a larger virtual machine.

System Programmer Response: You may have to reallocate external storage space; catalog missing modules into the saved segment area, LOADLIB, or DOSLIB; generate modules; or assist the operator in allocating a larger virtual machine size.

ARI0040E A database manager system error occurred - ARI*xxxx nn*.

Explanation: An internal error occurred within the database manager system. Usually a "should not occur" condition has occurred. A dump will be taken according to the DUMPTYPE initialization parameter. This is a system error.

ARI*xxxx* is the name of the module that detected the error.

nn is the error detection point within the module. See "Chapter 26. System Errors" on page 421 for a list of communication link errors.

Note: If the database manager has ended abnormally, a mini-dump will be displayed. These dumps are described in the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual. Refer also to message ARI0042I for additional diagnostic information.

System Action: The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0041E System hardware error occurred - ARI*xxxx nn*.

Explanation: A hardware error was detected during an I/O operation.

ARI*xxxx* is the name of the module that detected the error.

nn is the error detection point in the module.

A preceding message describes the cause of the hardware error. A dump or mini-dump will not be taken if a hardware error occurs.

System Action: The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Perform problem determination. If a hardware error is indicated, report this problem to the hardware service person at your installation. If a database system error occurred, make a record of what went wrong and contact the designated

support group for your installation.

ARI0042I Database manager reason code is *n1* - X'*n2*'.

Explanation: This code is a VM system return code returned to the database manager. It can be used for problem determination.

n1 is the decimal representation of the reason code.

n2 is the hexadecimal representation of the reason code.

Note: A previous message should have been displayed related to the reason code. See "Chapter 21. DB2 Server for VM Reason Codes" on page 399 and "Chapter 18. DBSS Return Codes" on page 271 for additional information about the reason codes. The reason code is a DB2 Server for VM reason code if the application program could not be loaded. (See message ARI0026E.)

System Action: The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Note the reason code for problem determination.

ARI0043I Database manager return code is *n*.

Explanation: This return code is for informational purposes. Except for return code 0, it is related to any earlier DB2 Server for VM messages that have been displayed. This return code is also returned to VM upon return by the database manager.

n is one of the following:

0 The database manager ended normally.

24 An error exists in the logic of the SQLSTART EXEC. Contact the designated database manager support group for your installation.

256 An error exists in the logic of the SQLSTART EXEC. Contact the designated database manager support group for your installation.

508 The database manager ended. This return code is returned when an SQLEND QUICK (SHUTDOWN QUICK) has been issued or when a problem such as a trace file close error occurred.

512 The database manager ended, but a limit error occurred (see message ARI0039E). Execution of any subsequent DB2 Server for VM applications may result in a failure if they depend upon the successful completion of a current DB2 Server for VM application.

516 The database manager ended, but an error occurred. (See message ARI0040E). This code is also returned if the operator enters an HX command. Execution of any subsequent DB2 Server for VM applications may result in a failure if they depend upon the successful completion of a current DB2 Server for VM application.

520 The database manager ended but a hardware error occurred. (See message ARI0041E). Execution of any subsequent DB2 Server for VM applications may result in a failure if they depend upon the successful completion of a current DB2 Server for VM application.

For nonzero return codes, see previous database manager error messages.

Note: When the database manager is operating in single user mode: If this message is accompanied by message ARI0046I, the return code passed back to the system will be the code from the database system.

System Action: Database manager termination continues.

ARI0044D If you want a dump, enter 1(Yes). Otherwise enter 0(No).

Explanation: The abnormal termination routine has been entered due to an operator request. If a dump is required, the operator must reply with a 1 or YES to the message. If 0 or NO is entered, no dump will be taken. Any other reply will cause message ARI0052E to be displayed. The dump to be taken is determined by the DUMPTYPE initialization parameter (DUMPTYPE=P, F, or N). If DUMPTYPE=N was specified, the default DUMPTYPE=F is assumed.

System Action: Processing continues as determined by the operator reply.

Operator Response: Reply 1 or YES if a dump is required. Reply 0 or NO if no dump is required.

ARI0045I Ready for operator communications.

Explanation: This informational message is issued only in multiple user mode. It informs the DB2 Server for VM virtual machine operator that the database manager is ready to accept DB2 Server for VM operator commands.

Operator Response: Enter any operator commands desired.

ARI0046I *xxxxxxxx* return code is *n1* - X'*n2*'.

Explanation: This message occurs only when the database manager is operating in single user mode. It reports a nonzero application return code (passed back in register 15) when the application program returns to the database manager.

xxxxxxxx is the name of the application program (specified in the PROGNAME initialization parameter).

n1 is the decimal representation of the return code.

n2 is the hexadecimal representation of the return code.

Some application programs (such as those written in COBOL or FORTRAN) do not pass back a zero return code upon successful completion. If the application program does not make the return code zero, whatever value is in register 15 will be displayed as the return code. This value is normally the entry point address of the application program; it is typically a large value.

IBM-supplied DB2 Server for VM programs (for example, a preprocessor or the DBS utility) that run in single user mode will also have their return codes displayed in this message.

If the name of the application program given in the message is one of the preprocessors:

ARIPRPA Assembler Preprocessor
ARIPRPB C Preprocessor
ARIPRPC COBOL Preprocessor
ARIPRPF FORTRAN Preprocessor
ARIPRPP PL/I Preprocessor

the following return codes apply:

0 No errors and no warnings.

1 The preprocessor BLOCK option was specified and
- One or more cursor operations are ineligible for blocking, or
- One or more SELECT statements were encountered (SELECT statements cannot be blocked).

4 Preprocessor or SQL warning. Either:
- A preprocessor warning message was issued.
- An SQL warning message was issued during the processing of an SQL command.

8 Preprocessor or SQL error. Either:
- A preprocessor error message was issued.
- An SQL error message was issued during the processing of an SQL command; a negative SQLCODE was returned. Preprocessing continues.

12 Preprocessor or SQL error. Preprocessor cannot continue. Probable causes are:
- File I/O error (on a GET, PUT, OPEN, or CLOSE).
- Get/release virtual storage failure.
- Preprocessor invocation parameter error.
- SQL error, such as preprocessor cannot connect to the database manager using the user ID or password provided to it (either explicitly or implicitly), or preprocessor experienced an internal SQL error during preprocessor initialization or shutdown.

16 Severe SQL error. Preprocessor cannot continue. Probable causes are:
- Package was not in the catalog or was not usable when an attempt was made to use it.
- Communication path to the database manager has been disabled.
- Error occurred while trying to free dynamically allocated virtual storage.
- Unrecoverable system error occurred.
- Attempt to process an SQL command before the database manager was started or ready.
- Error during attempt to open or write to a FORTRAN list file.

Return codes 12 and 16 are called preprocessor terminating errors. The preprocessor halts at the point of the error and performs its cleanup processing:
- Flushes the input source file.
- Frees any previously allocated virtual storage.
- Merges workfiles into output files (for COBOL only).
- Closes any open files.
- Restores the previous state of the package.
- Handles the preprocessor return code.

If the name of the application program given in the message is the DBS utility (**ARIDBS**), the following return codes apply:

0 All commands processed successfully.

4 An error occurred during DBS utility termination. However, all requested processing completed successfully. All changes were committed to the database. No SQL or DBS utility commands need to be reprocessed.

6 Errors have occurred during ERRORMODE CONTINUE processing. Some commands issued while SET ERRORMODE CONTINUE was in effect were not processed successfully.

8 DBS utility processing error occurred. From the point of the error, no further commands were executed, but later DBS utility commands were checked for syntax errors.

12 DBS control command file open error occurred. No commands were processed.

16 DBS message file open error occurred. No commands were processed.

20 Initialization error occurred. Not enough virtual storage was available for DBS utility working storage areas. No commands were processed.

When the database manager is started for database generation (STARTUP=C is specified), the program name is always ARIGCAT.

When the database manager is started for adding DBSPACEs (SQLADBSP EXEC), the program name is always ARISEGB.

Unless the return code is 512 or greater, the application return code is passed back to the system.

System Action: Database manager termination continues.

Operator Response: None, unless otherwise requested by the application programmer.

System Programmer Response: None, unless otherwise requested by the application programmer or by preceding messages if the application program is the DB2 Server for VM program.

ARI0047E *xxxxxxxx* **- Reason Code =** *nnn***.**

Explanation: This message only occurs with message ARI0026E. It is issued when an error other than "not found" or "insufficient storage" occurs while the database manager is trying to load the program indicated in message ARI0026E.

xxxxxxxx is the type of load operation being tried by the database manager when the error occurred. This value can be LOADMOD, NUCXLOAD, SEGMENT LOAD, FETCH (CMS/DOS), or LOAD.

nnn is the return code indicating the cause of the failure. Refer to the VM manual that documents the load operation in question. (For NUCXLOAD and SEGMENT macro, also see below.)

This message is accompanied by a DB2 Server for VM dump (as specified by the DUMPTYPE initialization parameter) and message ARI0040E, except when the program identified is ARICMOD. The return code in message ARI0043I is 516.

This message is issued once if the failure occurred while trying to load ARICMOD, ARIGCAT, or ARISEGB (and the error was other than "not found" or "insufficient storage"). For all other programs, the message is issued either four or five times. The database manager tries to load the application program through each of the VM load types in this order: LOADMOD, NUCXLOAD, SEGMENT LOAD, FETCH (only if CMS/DOS is on), and finally LOAD. The system tries each one until a successful load occurs or all fail:

- If *all* load attempts fail because of a "not found" condition, the database manager issues messages ARI0026E, ARI0039E, and ARI0042I. The reason code is 4, and the return code is 512.
- If *any* load attempt fails because of the "insufficient storage" condition, the database manager issues messages ARI0026E, ARI0039E, and ARI0042I. The reason code is 8, and the return code is 512.
- For other failure situations, the database manager issues message ARI0026E and four or five occurrences of message ARI0047E. (Five occur if CMS/DOS mode is in effect.) Message ARI0047E gives the reason code for each type of load failure. Finally, the database manager issues message ARI0040E. The return code is 516.

The NUCXLOAD and SEGMENT LOAD cases involve more than the execution of NUCXLOAD or SEGMENT LOAD alone.

For NUCXLOAD, the database manager may go through a NUCEXT QUERY, NUCXDROP, NUCXLOAD, and NUCEXT QUERY process to load the program. As a result, the reason code displayed in message ARI0047E may be a return code from NUCEXT QUERY or NUCXDROP, not NUCXLOAD. Consult the *VM/ESA: CMS Command Reference* manual for a description of the return codes from these commands as well as NUCXLOAD.

Similarly, if the program is to be loaded into a saved segment area, the database manager issues a SEGMENT LOAD macro. The reason code displayed in message ARI0047E may thus have resulted from the FINDSYS or PURGESYS, not the SEGMENT LOAD.

For a full description of this load search hierarchy, and the possible error conditions, refer to the information about starting an application server in single user mode in the *DB2 Server for VM System Administration* manual.

System Action: The database manager ends.

Operator Response: Notify your system programmer.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

ARI0048E Accounting specified, but the CP ACCT option was not specified in the directory entry of the database machine.

Explanation: DB2 Server for VM accounting records are written to the VM system accounting file by the CP DIAGNOSE instruction code X'4C'. This DIAGNOSE requires that the VM directory entry of the virtual machine issuing the DIAGNOSE (DB2 Server for VM application server) must specify the CP ACCT option. The ACCT is an operand of the VM OPTION directory control statement. Otherwise, the DIAGNOSE returns a condition code of 1.

System Action: If this condition occurs, accounting support is shut off and the database manager is ended.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the VM

directory entry of the application server. See the *VM/ESA: Planning and Administration* manual for details. Then restart the application server.

ARI0049E CP DIAGNOSE Instruction Code X'70' failure. Issue CP command SET ECMODE ON, reissue IPL CMS, and restart the database manager.

Explanation: DB2 Server for VM accounting support uses the CP DIAGNOSE instruction code X'70' to capture CPU time on a user basis. This DIAGNOSE requires that ECMODE is on. If ECMODE is on and the DIAGNOSE still fails, consult the *VM/ESA: CP Programming Services* manual for further details.

System Action: If this condition occurs, accounting support is shut off and the database manager is ended.

Operator Response: Refer this message to your system programmer.

System Programmer Response: If the problem occurs, and ECMODE is on, the virtual machine must be reset. A virtual machine is reset by issuing any of these CP commands: IPL, SET ECMODE, SYSTEM RESET, SYSTEM CLEAR, DEFINE STORAGE, DEFINE CHANNELS, and LOGOFF. After the virtual machine is reset, restart the database manager.

ARI0050E CP DIAGNOSE instruction code X'4C' failure.

Explanation: DB2 Server for VM accounting records are written by the CP DIAGNOSE instruction code X'4C'. While attempting to write a DB2 Server for VM accounting record, this DIAGNOSE failed. This is a DB2 Server for VM system error condition.

System Action: If this condition occurs, the accounting support is shut off, and the system is ended.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

ARI0052E *response* **is incorrect. Please reenter the correct response.**

Explanation: An incorrect value (*response* in text) was entered in response to the previous message. If *response* is blank, a null response (no response or blanks) was entered.

System Action: The previous message will be re-displayed.

Operator Response: Determine the valid response to the previous message, and enter the valid response when the prompting message is re-displayed.

ARI0053E An unexpected error occurred accessing the application server at initialization. DBSS Return Code = *code***.**

Explanation: During the database manager startup, the RDS module ARIXERO encountered unexpected problems when attempting to access the application server.

code is the DBSS return code from the open. Refer to "Chapter 18. DBSS Return Codes" on page 271.

System Action: The database manager is ended.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Note the DBSS return code, make a record of what went wrong, and contact the designated support group for your installation.

ARI0054E Unable to do COMMIT. DBSS Return Code = *return-code***.**

Explanation: During the database manager startup, ARIXERO was unable to issue a COMMIT after opening the record catalog table RIDs, TIDs, and IIDs.

return-code is the DBSS return code from the open. Refer to "Chapter 18. DBSS Return Codes" on page 271.

System Action: The database manager is ended.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0055A Ready trace output volume. Enter CUU or CANCEL.

Explanation: You specified one or more of the DB2 Server for VM initialization parameters TRACCONV, TRACDBSS, TRACRDS, TRACDSC, TRACWUM, TRACDRRM, and TRACSTG. DB2 Server for VM initialization is requesting the database virtual machine operator to:

- If trace output is to tape:
  1. Select a tape unit for trace output.
  2. Make the tape unit ready with a scratch tape mounted and properly positioned.
  3. Enter the cuu of the selected tape unit.
- If trace output is to a CMS file:
  - Enter the cuu of the CMS file minidisk.

You can reply CANCEL if no tape or CMS file is available or tracing is not desired.

If this message is re-displayed after message ARI0094E (indicating a trace file open error), the system is

requesting that corrective action for message ARI0094E be taken. After a scratch tape (for tape output) has been mounted and positioned correctly, enter the cuu of the selected tape unit. Reply CANCEL if the error condition cannot be corrected, no scratch tape is available, no tape unit is available, no CMS file is available, or the error condition represented by message ARI0094E cannot be corrected.

System Action: If the cuu is entered, the tape file or CMS file indicated on the CMS FILEDEF command for ARITRAC is opened. DB2 Server for VM initialization continues with DB2 Server for VM trace active.

If CANCEL is entered, message ARI0056I is displayed. DB2 Server for VM initialization continues with DB2 Server for VM trace inactive ("output disabled").

Operator Response: If this message was re-displayed after message ARI0094E, take the corrective action suggested for that message.

You must use the virtual tape unit or minidisk specified on the CMS FILEDEF command for ddname ARITRAC.

If output is to tape, notify the CP operator (using a CP MSG OP command) to ready (at the load point) a writable scratch tape volume on a tape unit, attach that tape to your virtual machine with the ARITRAC FILEDEF-specified cuu, and notify you when the tape is ready for use. Then enter the cuu of the virtual tape unit. If no tape unit or volume is available, or the error represented by message ARI0094E cannot be corrected, enter CANCEL. DB2 Server for VM initialization will proceed without tracing.

Note: The virtual tape unit or CMS minidisk specified on the CMS FILEDEF command for ddname ARITRAC will be used for the file regardless of the cuu you enter.

ARI0056I Tracing terminates because of operator request.

Explanation: During database initialization, you specified the parameters TRACCONV, TRACDBSS, TRACRDS, TRACDSC, TRACWUM, TRACDRRM, or TRACSTG (or any combination). The initialization dialog prompted the operator for the tape unit or CMS minidisk cuu for the trace output file (message ARI0055A) and the operator responded with CANCEL to the prompting message. (The operator refers to the user of the database virtual machine console.) The message indicates that database initialization is continuing but with trace inactive ("output disabled").

System Action: Database initialization continues, but with the trace inactive, the TRACCONV, TRACDBSS, TRACRDS, TRACDSC, TRACWUM, TRACDRRM, or TRACSTG parameters are overridden.

Note: Since trace is "output disabled", the TRACE OFF operator command must be entered before the TRACE ON operator command can be entered.

ARI0057E Message ARI*nnnn* has formatting error.

Explanation: This is a database error. The message identified by *nnnn* (that is, message prefix ARI) either:

- Was too long to fit into the internal message buffer (68 bytes for the database manager or 79 bytes for ISQL). Part of the message was truncated and not displayed.
- Had one or more variable substitution requests which could not be performed. Either the variable was not supplied by the requesting module or the variable had an invalid length. The message line contains **&n.** or **&na.** (where **n** is the variable number in the internal structure VARLIST and **a** is an optional conversion alphabetic character). These values identify the unresolved variables.

For a multi-line message, the error may have occurred on any (or all) lines. The message in error precedes this message on the display terminal or printer.

System Action: Normal processing continues.

Operator Response: Save the message output (this message and the message identified by **nnnn**) and notify your system programmer.

Programmer Response: Same as Operator Response.

System Programmer Response: Save the message output and report this problem to the designated support group for your installation. If the application program using the database manager is known, save information about it so that the problem can be recreated if necessary.

ARI0058E Message editing error {MSG ID = *nnnn* | SQLCODE = *nnnn*}. Reason Code = *nn*.

Explanation: This is a system error. A database component requested that a message be edited and displayed. If **SQLCODE** appears in the message text, the request was for the SQLCODE message associated with SQLCODE nnn. If **MSG ID** appears in the message text, the request was for the database message ARInnnn (*nnnn* from message text).

The exact cause of the message editing error is identified by the **Reason Code=***nn* value in the message text where:

01 A message number is outside a valid range.

02 SQL code number is out of range.

03 This is an invalid DBCS substitution string.

04 Message is not found in the repository.

05 The repository is not found.

06 Sequence number contains wrong or non-existing message line number.

07 Invalid buffer length.

You may be able to identify the lost message from the displayed message number or SQLCODE and take appropriate action.

System Action: Processing continues as though the lost message had been displayed. Subsequent messages may clarify system activity.

User Response: If the reason code is 04 and the database machine is operating at a later release than your user machine, use the message number or SQLCODE in the message text to help you determine the release level of the database machine.

Operator Response: Save this message and, if possible, any additional information about the application program or the database manager program which was running and notify your system programmer.

Programmer Response: See Operator Response.

System Programmer Response: Save the message output and any other information given to you by the operator or programmer and report this problem to the designated support group for your installation. If the application program using the database manager is known, save information about it so that the problem can be recreated if necessary.

ARI0059E Program cancelled due to insufficient virtual storage.

Explanation: An attempt to acquire virtual storage (using the CMS free storage macro) in the virtual machine failed.

System Action: The program will be canceled because of insufficient virtual storage.

Operator Response:

If a return code other than 1 from the free storage macro is displayed (see *Note*), this indicates a VM or database system error. The system programmer should verify that a user program is not damaging CMS storage pointers (free storage macro return codes 2 and 3). Assuming that a user error did not occur, the system programmer should determine the service level of the database manager and report this problem to the designated support structure for your installation.

Otherwise, assume that the virtual storage is not enough and proceed as follows. Either run the failing program in a larger virtual machine or change the program parameters so that it requires less virtual storage. If you are running a program with the database manager in single user mode, you may be able to run it in a smaller virtual machine while the database manager is running in multiple user mode. Consult with your system programmer to help you. (You can use the CP QUERY STORAGE command to display the virtual storage size of your virtual machine. You can set the size of your virtual machine by using the CP DEFINE STORAGE command before you use the CP IPL CMS command.)

Note: An attempt is made to provide information about:

- Calling module name (or address).
- Requested virtual storage size (in bytes).
- Reason code containing the operating system return code (from the free storage macro). (See message ARI0042I.)
- Database return code. (See message ARI0043I.)

A dump of the virtual machine is issued if this attempt fails because of a lack of storage. The dump is displayed on the virtual printer.

System Programmer Response: See Operator Response.

ARI0060I Database manager initialization complete.

Explanation: The database manager has completed the initialization process in multiple user mode. It is ready to accept work requests from other virtual machines.

System Action: The database manager enters wait state, waiting for work requests from other virtual machines and for database manager operator requests to enter operator commands.

Operator Response: Database application programs including preprocessors, ISQL, and the Database Services Utility program may be started in other virtual machines.

Database operator commands may be entered from the operator console.

ARI0061E The operator command value *value* is too long.

Explanation: The database manager operator or ISQL terminal user has entered a database operator command which has a command name or parameter which exceeds eighteen characters in length. Database operator command names and parameters may not exceed eighteen characters. The first eighteen characters of the invalid value are displayed in the message. This error may be caused by misspelling an operator command name or parameter or by omitting the blank required between the parameters of the command or between the command name and the first parameter.

System Action: Operator command processing ends.

Operator Response: Reenter the command with the correct value. For a description of the database operator

commands, see the *DB2 Server for VSE & VM Operation* manual.

User Response: Reenter the command with the correct value. See the *DB2 Server for VSE & VM Interactive SQL Guide and Reference* manual.

ARI0063I You did not enter an operator command.

Explanation: The database manager operator has entered a null line which the database manager has attempted to process as a database operator command, or ISQL has transmitted a blank line for database operator command processing. This should not occur for ISQL (ISQL internal error).

System Action: Operator command processing ends.

User Response: You may enter a database operator command or an ISQL or SQL command. If you are in ISQL, this message indicates an internal error in ISQL. In this case, notify your system programmer.

System Programmer Response: If message ARI0063I is received by an ISQL terminal user, ISQL is malfunctioning. Determine the service level of the database manager and report this problem to the designated support group for your installation.

ARI0064E You entered an invalid operator command.

Explanation: The operator or the ISQL terminal user has entered an operator command. Either:

- The command name is not a valid operator command, or
- The command name is a name other than COUNTER or SHOW (ISQL command input only), or
- The ARCHIVE or LARCHIVE operator command was entered with operands.

Notes:

1. ISQL should process only COUNTER and SHOW as operator commands. Therefore, ISQL has an internal error.
2. The command RESET CRR LOGNAME is not valid when SYNCPNT=N.

System Action: Operator command processing ends.

Operator Response: Initiate communication with the database manager and reenter the command with the correct name and operands. For the descriptions of the database operator commands, see the *DB2 Server for VSE & VM Operation* manual.

User Response: You may enter a database operator command or an ISQL or SQL command. This message indicates an internal error in ISQL. Notify your system programmer.

System Programmer Response: If message ARI0064E is received by an ISQL terminal user, ISQL is malfunctioning. Determine the service level of the database manager and report this problem to the designated support group for your installation.

ARI0065I Operator command processing is complete.

Explanation: This message notifies the operator or the ISQL terminal user that a database operator command has successfully completed processing.

System Action: Normal processing continues.

User Response: You may enter another database operator command or an ISQL or SQL command.

ARI0066E An error occurred processing the operator command.

Explanation: A database operator command was routed to the module which processes the command. The module detected an error and ended command processing. If the error was an invalid command parameter, the module issued a message identifying the error. If the module invoked database services which detected an error (for example, I/O error), then the operator received an error message due to the error. Note that in the last case, the operator command may have come from an ISQL terminal but the specific error message is displayed to the operator.

System Action: Operator command processing ends.

Operator Response: Refer to the previous error message for the cause of the error and for corrective action.

User Response: If there is a previous error message indicating the cause of command failure, take the corrective action indicated by that message. If there is no previous error message, the command was not executed due to some error detected by the database manager (such as an I/O error) and the operator has been notified of the error. If the problem persists, you may wish to notify the system programmer of your problem.

System Programmer Response: Obtain the ISQL terminal user and operator error message output and take the corrective action recommended or suggested by the error messages.

ARI0067E Invalid SHOW command entered.

Explanation: This message was caused by the operator or ISQL terminal entering:

- The operator SHOW command with an invalid or omitted first operand
- A SHOW command with an incorrect number of operands.

System Action: Operator command processing ends.

Operator Response: Reenter the SHOW command with the correct operands. For a description of this

command, refer to the *DB2 Server for VSE & VM Operation* manual.

User Response: Reenter the SHOW command with the correct operands. For the description of SHOW commands, refer to the *DB2 Server for VSE & VM Operation* manual. You may alternatively enter an ISQL or SQL command.

ARI0068E Invalid SET command entered.

Explanation: The operator SET command operand was invalid or omitted.

System Action: Operator command processing ends.

Operator Response: Reenter the SET command with the correct operand. For the description of this command, refer to the *DB2 Server for VSE & VM Operation* manual.

ARI0069E READ FILE *fn ft fm* failed, REASON=*nn* CODE=*nn*.

Explanation: During reading of the SYSIN file, a READ FILE statement was encountered. (The READ FILE statement is described in the *VM/ESA: CMS Command Reference* manual.) When the system attempted to access and read the CMS file, an error occurred.

fn ft fm identifies the file name, file type, and file mode of the CMS file. REASON is the Primary Error Code, and CODE is the Secondary Error Code.

For further explanation, look up the displayed error codes under either "Chapter 24. Include File Access Errors" on page 411, or "INCLUDE FILE ERRORS" in the HELP text.

System Action: See subsequent database messages. Normally this error condition causes the database manager program to end.

User Response: Save this error message and give it to the program submittor or the system programmer for problem resolution.

System Programmer Response: For further explanation, look up the displayed error codes under either "Chapter 24. Include File Access Errors" on page 411, or "INCLUDE FILE ERRORS" in the HELP text.

ARI0081E Write error trace file. Reason Code = *nn*. Error Code = *nn*.

Explanation: When the system attempted to write trace data to the trace output file, an access error occurred. Trace status is now "output disabled". The trace file has the ddname ARITRAC. The Reason Code value is the Primary Error Code. The Error Code value is the Secondary Error Code. For further explanation, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Action: Any further database trace output is disabled. This error condition prevents closing the trace output file during TRACE OFF command processing, database manager shutdown, or application termination.

User Response: Save this message and give it to your system programmer. Do not try to process the trace output file.

See Operator Response for message ARI0097I for possible action to salvage the accumulated trace output data.

Operator Response: See User Response.

System Programmer Response: For corrective action, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text. Note that although you may have partial trace output, it cannot be cleanly processed because the tape or CMS file has not been closed.

ARI0082E TRACE command parameter *n* [*parameter*] invalid or missing.

Explanation: The TRACE operator command was entered and is being processed. One of the following conditions was detected:

- A required parameter contains an invalid value.
- A required parameter was omitted.
- An optional parameter contains an invalid value.
- An undefined extra parameter was entered.

Parameter *n* identifies which parameter was omitted or contains the invalid value. *parameter*, if present in the message text, identifies the invalid parameter value. If not present, it indicates that a required parameter was omitted.

System Action: TRACE command processing ends. There is no change in the status of the database trace facility.

Operator Response: Consult the TRACE command section of the *DB2 Server for VSE & VM Operation* manual, and compare the TRACE command description with the TRACE command that you entered and the text of this message. Determine which parameter was erroneously specified or omitted and reenter the TRACE command.

ARI0083I Trace command specified {ON|OFF}, but trace already {ON|OFF CLOSED}.

Explanation: One of the following occurred:

- The TRACE ON command was specified and the database trace facility is already active (ON) or "output disabled" due to successful processing of a

previous TRACE ON operator command (and no subsequent successful TRACE OFF operator command).

- The TRACE ON command was specified and the database trace facility is already active (ON) or "output disabled" due to successful processing of trace parameters (TRACCONV=, TRACDBSS=, TRACRDS=, TRACDSC=, TRACWUM=, TRACDRRM=, TRACSTG=) specified with the initialization parameters (and no subsequent successful TRACE OFF operator command).
- The TRACE OFF command was specified and the database trace facility is already inactive (OFF) and no trace file is open (trace is in OFF CLOSE state). The message text specifies OFF and OFF CLOSED.

TRACE ON cannot be validly specified unless trace is inactive (OFF). TRACE OFF cannot be validly specified unless trace is active (ON) or inactive but with the trace file open (OFF NOCLOSE).

System Action: TRACE command processing ends. There is no change in status of the database trace facility.

Operator Response: None is required since trace is already in the state specified by the TRACE command. If you want to specify new tracing (TRACE ON) parameters, you must first successfully enter the TRACE OFF command (the NOCLOSE option allows the trace file to be kept open for further output in the same trace file).

ARI0084D Enter one of userid, AGENT n, * (for all), or CANCEL.

Explanation: The TRACE ON operator command is being processed. The database manager is prompting the operator to specify which database user ID or agent number (or * for all agent numbers) is to be traced.

System Action:

- If a user ID is entered, trace output will occur only for the database Agents processing on behalf of the specified user ID.
- If AGENT n is entered, trace output will occur only for processing within the specified Agent number.
- If * is entered, trace output will occur for processing in all DB2 Server for VM Agents.
- If CANCEL is entered, TRACE command processing ends.

Operator Response: Enter one of the following:

User ID
Indicates tracing by database user ID. Although lower case characters may be entered, they are translated to upper case by the database manager. Also, imbedded blanks may not be entered since the blank (space) character is the database parameter delimiter character. You should attempt to trace by a user ID only if the user ID consists entirely of numeric and uppercase alphabetic characters with no imbedded blanks (low order blanks are no problem).

AGENT n
AGENT indicates trace by agent number; **n** is a valid database Agent number (blank between AGENT and n).

AGENT 1	Operator Agent.
AGENT 2	Checkpoint/Archive Agent.
AGENT 3	In single user mode is General Purpose User Agent. In multiple user mode is Ready/Recovery Agent.
AGENTS 4-n	In multiple user mode are General Purpose User Agents.

Note: On SHOW commands, user agents 4 through **n** are displayed as agents 1, 2, ..., NCUSERS initialization parameter value.

* Indicates that all database Agents are to be traced.

CANCEL
Ends TRACE command processing.

ARI0085E Agent number [*value*] invalid or missing.

Explanation: An incorrect or null (blank) agent number value was entered in response to TRACE ON command prompting message ARI0084D.

- If *value* is missing from the message text, the keyword AGENT was entered but the agent number value was omitted.
- If *value* is present in the message text, an agent number value was entered which was not an unsigned number, was a zero, or exceeded the number of agents present in the database manager. The number of agents can be determined by the SHOW ACTIVE operator command.

System Action: TRACE ON prompting message ARI0084D is re-displayed to allow the correct response to be entered.

Operator Response: Determine the valid agent number value you will enter in response to prompting message ARI0084D, and enter the valid response when the prompting message is re-displayed.

ARI0086E Invalid response {BLANK|*value*} to prompt.

Explanation: An incorrect value (*value* in message text) or no value (**BLANK** in message text) was entered in response to a TRACE ON command or trace

initialization prompting message.

- For message ARI0055A, a blank or null response was entered (**BLANK** in text), an incorrect cuu value was entered (not three characters or not valid hex characters), or the response or the response CANCEL was misspelled.
- For message ARI0084D, a blank or null response was entered.
- For message ARI0087D, either only a blank or null response was entered (**BLANK** in text), or a value other than CONV, DBSS, RDS, WUM, DRRM, DSC, or STG was entered (*value* in text).
- For message ARI0088D, either only a blank or null response was entered (**BLANK** in text), or a value for function name other than *, EXEC, PA, OPT, AG, INT, or AU was entered (*value* in text).
- For message ARI0090D, either only a blank or null response was entered (**BLANK** in text), or a value for function name other than *, ENTRY, EXIT, LOG, LOCK, LUW, DC, DM, STOR, SORT, INDEX, or STAT was entered (*value* in text).
- For message ARI0091D, either only a blank or null response was entered (**BLANK** in text), or a value that was not an unsigned numeric, or that was not in the range 0 to 14999 or 20000 to 24999, was entered (*value* in text).
- For message ARI0140D, a blank or null response was entered (**BLANK** in text), or a value for function name other than *, DICT, GEN, PARSE, or RDIIN was entered (*value* in text).
- For message ARI0141D, either only a blank or null response was entered (**BLANK** in text).
- For message ARI0142D, either only a blank or null response was entered (**BLANK** in text), or a value for function name other than *, AGENT, or COM was entered (value in text).
- For message ARI0160D, a blank or null response was entered (**BLANK** in text).
- For message ARI0162D, a blank or null response was entered (**BLANK** in text).

System Action: The TRACE prompting message (to which the invalid response was entered) is redisplayed.

Operator Response: Determine the valid response to the trace prompting message, and enter the valid response when the prompting message is re-displayed.

ARI0087D **Enter one or more component names or CANCEL. Valid component names are DBSS, RDS, [WUM, DRRM,] DSC, CONV, and STG.**

Explanation: The TRACE ON operator command is being processed. The database manager is prompting the operator to specify the major components (DBSS, RDS, WUM, DRRM, DSC, CONV, or STG) that are to be traced.

System Action:

- If the operator enters DBSS, message ARI0090D appears and prompts the operator for the functions of DBSS to trace.
- If the operator enters RDS, message ARI0088D appears and prompts the operator for the functions of RDS to trace.
- If the operator enters WUM, message ARI0141D appears and prompts the operator for the trace level.
- If the operator enters DRRM, message ARI0140D appears and prompts the operator for the functions of DRRM to trace.
- If the operator enters DSC, message ARI0142D appears and prompts the operator for the functions of DSC to trace.
- If the operator enters CONV, message ARI0160D appears and prompts the operator for the trace level.
- If the operator enters STG, message ARI0162D appears and prompts the operator for the trace level.
- If the operator enters CANCEL, TRACE command processing ends.

Operator Response:

- If you want DBSS tracing, enter DBSS.
- If you want RDS tracing, enter RDS.
- If you want WUM tracing, enter WUM.
- If you want DRRM tracing, enter DRRM.
- If you want DSC tracing, enter DSC.
- If you want CONV tracing, enter CONV.
- If you want STG tracing, enter STG.
- If you want to end the TRACE command, enter CANCEL.

Enter any of DBSS, RDS, WUM, DRRM, DSC, CONV, or STG (with one or more blanks between them). You can enter them in any order.

Note: You can see this message in all environments. In environments where components are not active, however, the components tracing does not take place, even if the operator specifies them.

ARI0088D **Enter RDS function name and trace-level pairs.**

Valid function names are *, EXEC, PA, OPT, AG, INT, AU, SG, and CANCEL.
Valid trace-level values are 0, 1, and 2.

Explanation: The TRACE ON command is being processed. The operator has responded RDS to prompting message ARI0087D. The database manager is prompting the operator to specify which functions of RDS to be traced and the desired trace level for each function.

System Action:

- If * n is entered, all RDS functions will be traced at trace level n.

- If EXEC n, or PA n, or OPT n, or AG n, or INT n, or AU n, or SG n are entered, the specified functions of RDS will be traced at the specified trace level (n).
- If CANCEL is entered, TRACE command processing ends.

Note: If trace level 0 is entered for a function, it will not be traced (equivalent to not entering the function at all). Therefore, if you want to trace all RDS functions (at level 2, for example) except INT, you can enter * 2 INT 0. Since INT appears after the * specification, it overrides the implicit INT 2 implied by * 2. Likewise, entering : * 1 EXEC 2 will cause all RDS functions except EXEC to be traced at level 1 and EXEC to be traced at level 2.

Operator Response:

- If all RDS functions are to be traced, enter * plus 1 or 2 for desired trace level (with blank between * and number).
- If only selected RDS functions are to be traced, enter first function name, then its trace level value, then, if desired, second function name, then its trace level value, etc. Note that one or more blanks must appear between each function name and each trace level value entered. Note also that if you enter a function name more than once, the last entry is effective.
- If TRACE command termination is desired, enter CANCEL.
- See *Note* in the System Action section of this message for possible use of * in combination with other function names.

ARI0089E **Trace-level [*value*] for {RDS | CONV | DBSS | DSC | WUM | DRRM | STG} {* | *function name*} invalid or missing.**

Explanation: You entered one of the following:

- An incorrect trace-level value ([*value*] displayed in message text)
- No trace-level value ([*value*] omitted from message text).

Your entry was in response to one of the following TRACE ON command prompting messages:

- ARI0088D (RDS in message text)
- ARI0160D (CONV in message text)
- ARI0090D (DBSS in message text)
- ARI0140D (DRRM in message text)
- ARI0141D (WUM in message text)
- ARI0142D (DSC in message text)
- ARI0162D (STG in message text).

{ | *function name}* in message text identifies the valid RDS, DBSS, DSC, or DRRM function name that precedes the omitted or incorrect trace-level value.

System Action: The TRACE ON prompting message (to which the incorrect response was entered) is re-displayed.

Operator Response: Determine the valid response to the trace prompting message (each function name must be followed by a valid trace-level value of 0, 1, or 2 with a blank between them) and enter the valid response when the prompting message is re-displayed.

ARI0090D **Enter DBSS function name and trace-level pairs.**

Valid function names are *, ENTRY, EXIT, LOG, LOCK, LUW, DC, DM, STOR, SORT, INDEX, STAT, and CANCEL.
Valid trace-level values are 0, 1, and 2.

Explanation: The TRACE ON command is being processed. The operator has responded DBSS to prompting message ARI0087D. The database manager is prompting the operator to specify which functions of DBSS are to be traced and the desired trace level for each function.

System Action:

- If * n is entered, all DBSS functions will be traced at trace level n.
- If ENTRY n, or EXIT n, or LOG n, or LOCK n, or LUW n, or DC n, or DM n, or STOR n, or SORT n, or INDEX n, or STAT n are entered, the specified functions of DBSS will be traced at the specified trace level (n).
- If CANCEL is entered, TRACE command processing ends.

Note: If trace level 0 is entered for a function, it will not be traced (equivalent to not entering the function at all). Therefore, if you want to trace all DBSS functions (at level 2, for example) except INDEX, you can enter: * 2 INDEX 0. Since INDEX appears after the * specification, it overrides the implicit INDEX 2 implied by * 2. Likewise, entering: * 1 ENTRY 2 will cause all DBSS functions except ENTRY to be traced at level 1 and ENTRY to be traced at level 2.

Operator Response:

- If all DBSS functions are to be traced, enter * plus 1 or 2 for desired trace level (with blank between * and number).
- If only selected DBSS functions are to be traced, enter first function name, then its trace level value, then, if desired, second function name, then its trace level value, etc. Note that one or more blanks must appear between each function name and each trace level value entered. Note also that if you enter a function name more than once, the last entry is effective.

- If TRACE command termination is desired, enter CANCEL.
- See the note in the SYSTEM ACTION section of this message for possible use of * in combination with other function names.

ARI0091D Enter the dump trace point number, or CANCEL to terminate trace command processing.

Explanation: The TRACE ON command is being processed. The operator specified the DUMP parameter in the TRACE ON command. The database manager is prompting the operator to specify the CONV, DBSS, RDS, DSC, WUM, DRRM, or STG trace point number that is to cause a database snap dump to occur when the trace point is first activated. The snap dump will dump either all of the database virtual machine (full dump), or only that part of the virtual machine which does not contain database load modules (partial dump). The database initialization parameter DUMPTYPE will control whether a full dump is given (DUMPTYPE=F) or a partial dump is given (DUMPTYPE=P or N).

System Action: The entered trace point number is stored. When that trace point is first activated for tracing, a snap dump is produced on the virtual printer. After the first activation, no more snap dumps are produced.

If CANCEL is entered, TRACE command processing ends.

Operator Response:

- Enter the CONV, DBSS, RDS, DSC, WUM, DRRM, or STG trace point number that you want. The number must be unsigned and in the range of 0 to 14999 or 20000 to 24999.
- If TRACE command termination is desired, enter CANCEL.

ARI0093I TRACE command terminated by request.

Explanation: The TRACE ON command was being processed. The operator responded CANCEL to a prompting message.

System Action: TRACE command processing ends. There is no change in status to the database trace facility.

Operator Response: The operator may reenter the TRACE ON command.

ARI0094E Trace file *cuu* assign or open failed. Reason Code = *nn*. Return Code = *nn*.

Explanation: The database manager is processing the TRACE ON command or initializing a database trace during database or Resource Adapter initialization. An error occurred while the Resource Adapter was attempting to open the trace output file on tape unit or CMS minidisk *cuu*. The Reason Code value is the Primary Error Code. The Return Code value is the Secondary Error Code. For further explanation, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Action: If this error occurs in the database machine, then either:

- TRACE command processing ends (For TRACE command processing, the trace facility remains inactive.), or
- Database initialization reissues message ARI0055A. This allows the virtual machine terminal user to correct the problem indicated by this error message, ready the trace output volume, and retry the trace file open processing.

If this error occurs in a user virtual machine, the application abends.

User Response: For corrective action, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

Operator Response: See User Response.

System Programmer Response: See User Response.

ARI0095I Trace now on with output on unit *cuu*.

Explanation: The TRACE ON command has been successfully processed, or database initialization has successfully initialized trace ON. Database trace is now active with trace output on tape unit or CMS minidisk *cuu*.

System Action:

- Trace is active.

  TRACE ON command processing is completed. Trace is active as specified by the values entered for the various TRACE ON command prompting messages, or initialization trace initialization is completed. The TRACCONV, TRACDBSS, TRACRDS, TRACDSC, TRACWUM, TRACDRRM, or TRACSTG initialization parameters specify how TRACE is active.
- Trace output is directed to tape unit or CMS minidisk *cuu*.

ARI0096I TRACE now OFF. Trace file remains open on unit *cuu*.

Explanation: The TRACE OFF NOCLOSE command was successfully processed. Trace is inactive and the trace output file remains open on tape unit or CMS minidisk *cuu* for subsequent trace output (after using a TRACE ON command). This message will occur if trace was active when the command was entered or if trace was already inactive but with the trace output file open

(due to a prior TRACE OFF NOCLOSE command).

System Action: TRACE OFF command processing has completed. Trace is completely inactive.

Operator Response: None required. A TRACE ON command may now be entered. Note that a TRACE OFF command with the NOCLOSE parameter omitted (or database manager termination which closes the tape or minidisk file) must be executed before the trace output file can be processed.

ARI0097I TRACE was output disabled. TRACE is now OFF. It did not close unit {*cuu* | UA}.

Explanation: The database manager is being normally or abnormally terminated, or the TRACE OFF command (with or without the CLOSE or NOCLOSE parameters) was processed. Trace was previously ON (active) due to the TRACE ON operator command or the initialization parameters TRACCONV, TRACDBSS, TRACRDS, TRACDSC, TRACWUM, TRACDRRM, or TRACSTG.

Trace became "output disabled" (trace output suppressed) because of either:

- A failure to open the trace output file during database initialization (CANCEL reply to message ARI0055A); or
- A trace output file write error (message ARI0081E).

The TRACE OFF command or database manager termination caused trace to be turned off (fully disabled), but the trace output file was not closed due to the previous problem (open failure or write error).

If *cuu* is in the message text, the trace output file was opened and may contain trace output data, but due to the write error and failure to close the file, the file may not be usable.

If **UA** is in the message text, the trace output file was not successfully opened during database initialization.

System Action: Trace has been turned off (disabled).

Operator Response: None required. If the message was in response to the TRACE OFF command, a TRACE ON command may be entered. If trace output is to tape, you may attempt to salvage the trace data on the output tape by using the CMS TAPE WTM TAPn command. In any case, you can attempt to run the DB2 Server for VM Trace Formatter Utility Program (ARIMTRA). It will format and output all trace data or selected trace data up to the point of failure. At that point, it will probably end due to invalid data or an I/O error on the tape or minidisk file.

ARI0098E Trace file *cuu* unassigned or CLOSE failed. Reason Code = *nn*. Return Code = *nn*. TRACE OFF. Processing completed.

Explanation: The TRACE OFF command (CLOSE is the default) is being processed or the database manager is being normally or abnormally ended. In attempting to close the trace output file on tape unit or CMS minidisk cuu, a terminating error occurred. The Reason Code value is the Primary Error Code. The Return Code value is the Secondary Error Code. For further explanation, look up the displayed error codes under either “Chapter 23. Sequential File Errors” on page 407, or “SEQ FILE ERRORS” in the HELP text.

System Action: TRACE OFF command processing or trace shutdown is completed. Trace is in the OFF state. Assume that the trace tape or minidisk file was not closed and is not fully processable (see below).

Operator Response: Report the problem (message) to your system programmer. Include any CMS minidisk or tape unit error messages displayed by CMS.

You may attempt to run the DB2 Server for VM Trace Formatter Utility Program (ARIMTRA). It will format and output all trace data up to the point of failure. At that point it may end due to lack of tape/minidisk CLOSE recording or an unreadable record on the tape/minidisk.

Note for tape output: You may wish to notify the CP operator to demount the tape volume, save it, and detach the real tape unit from the virtual machine.

System Programmer Response: For corrective action, look up the displayed error codes under either “Chapter 23. Sequential File Errors” on page 407, or “SEQ FILE ERRORS” in the HELP text.

ARI0099I TRACE OFF. Output file closed on unit *cuu*.

Explanation: Database trace has been successfully ended due to entry of the TRACE OFF CLOSE (CLOSE is the default) operator command or due to database manager normal or abnormal termination. Trace is off and the trace output file has been successfully closed.

System Action: TRACE OFF command processing or trace shutdown is completed. The trace file is closed.

Operator Response: None required. The trace output tape or minidisk is now available for print display by the DB2 Server for VM Trace Formatter Utility Program (ARIMTRA).

Note for tape output: You may wish to notify the CP operator to demount the tape volume, save it, and detach the real tape unit from the virtual machine.

ARI0100E **Agent/user ID {*agent-id* | *user-id*} is not assigned.**

Explanation: A SHOW LOCK operator command could not be processed. Sources of the problem include:

- An invalid user ID was entered
- An invalid agent number was entered (not numeric or not in the range 1 through NCUSERS)
- The user, identified by *user-id*, was not assigned to a real agent at the instant the SHOW command was entered.

System Action: The SHOW command processing is ended.

Operator Response: Be sure that the specified *agent-id* or *user-id* is valid and reenter the SHOW command.

Note: You may get the same result if the requested agent/user does not occupy a real agent at the time the command is reentered.

ARI0101E **DBSPACE *dbspace-number* does not exist.**

Explanation: The *dbspace-number* entered on the SHOW LOCK DBSPACE operator command was invalid:

- Not numeric or
- Not in the range of 1 to n where n is the number of PUBLIC and PRIVATE DBSPACEs defined in the database (and with rows in catalog table SYSTEM.SYSDBSPACES).

System Action: The SHOW command processing is ended.

Operator Response: Correct the DBSPACE number (be sure that the DBSPACE number is between 1 and the maximum valid value for DBSPACE numbers) and reenter the command.

ARI0102E **There is no lock contention for {any | this} {DBSPACE | agent/user}.**

Explanation: The database manager could not find a **DBSPACE** or an *agent/user* that had locking contention at the instant the SHOW command was entered. This message occurred as a result of entering a SHOW LOCK DBSPACE, SHOW LOCK USER, SHOW LOCK WANTLOCK, SHOW LOCK ACTIVE, or SHOW LOCK GRAPH command.

System Action: The SHOW command processing is ended.

Operator Response: Reenter the command.

Note: You may get the same result if the requested agent, user, or DBSPACE has no lock contention at the time the command is reentered.

ARI0103E **A column named *name1* does not exist in the system catalog table named *name2*.**

Explanation: The row for the column name *(name1)* was not found in the system table *name2*. *name1* is either TLABEL or CLABEL. The TLABEL column resides in SYSTEM.SYSCATALOG and contains correlation names. *name2* is either SYSTEM.SYSCATALOG or SYSTEM.SYSCOLUMNS. The CLABEL column resides in SYSTEM.SYSCOLUMNS and contains column labels.

The database manager expects there to be a column named TLABEL in SYSTEM.SYSCATALOG and CLABEL in SYSTEM.SYSCOLUMNS when looking for a row describing those columns. It could find neither of them.

System Action: Processing continues.

Operator Response: Refer this message to your database administrator.

Database Administrator Response:

1. Determine why the row was deleted, or why it was updated with invalid values, and what values should be inserted.
2. Insert or update a row for TLABEL or CLABEL in SYSTEM.SYSCOLUMNS with the appropriate values.
3. Restart the database manager.

ARI0104I **The row, where SQLOPTION = *option1* does not exist, or has incorrect data in the VALUE column in the catalog table named SYSOPTIONS. The database manager will assume a value of *option2*.**

Explanation: You omitted a data type value or gave an incorrect value. The system assumed a value instead.

The SQLOPTION = *option1* possibilities are:

SQLOPTION = DBCS
The row for DBCS option in SYSTEM.SYSOPTIONS (where SQLOPTION = DBCS) is missing or contains an invalid value. The correct value is either YES or NO.

SQLOPTION = CHARSUB
The row for the CHARSUB option in SYSTEM.SYSOPTIONS (where SQLOPTION = CHARSUB) is missing or contains an invalid value. The correct values are SBCS or MIXED.

SQLOPTION = CCSIDSBCS
The row for the CCSIDSBCS option in SYSTEM.SYSOPTIONS (where SQLOPTION = CCSIDSBCS) is missing or contains an invalid CCSID value. The correct values are recorded in the SYSTEM.SYSCCSIDS catalog table.

SQLOPTION = CCSIDMIXED
The row for the CCSIDMIXED option in SYSTEM.SYSOPTIONS (where SQLOPTION = CCSIDMIXED) is missing or contains an invalid CCSID value. The correct values are recorded in the SYSTEM.SYSCCSIDS catalog table.

SQLOPTION = CCSIDGRAPHIC
The row for the CCSIDGRAPHIC option in SYSTEM.SYSOPTIONS (where SQLOPTION = CCSIDGRAPHIC is missing or contains an invalid CCSID value. The correct values are recorded in the SYSTEM.SYSCCSIDS catalog table.

SQLOPTION = MCCSIDSBCS
The row for the MCCSIDSBCS option in SYSTEM.SYSOPTIONS (where SQLOPTION = MCCSIDSBCS) is missing or contains an invalid CCSID value.

SQLOPTION = MCCSIDMIXED
The row for the MCCSIDMIXED option in SYSTEM.SYSOPTIONS (where SQLOPTION = MCCSIDMIXED) is missing or contains an invalid CCSID value.

SQLOPTION = MCCSIDGRAPHIC
The row for the MCCSIDGRAPHIC option in SYSTEM.SYSOPTIONS (where SQLOPTION = MCCSIDGRAPHIC) is missing or contains an invalid CCSID value.

System Action: Processing continues. The implied value of *option2* is:
- For SQLOPTION = DBCS, DBCS option will be NO.
- For SQLOPTION = CHARSUB, CHARSUB option will be SBCS.
- For SQLOPTION = CCSIDSBCS, the CCSIDSBCS option will be determined from the current CHARNAME value.
- For SQLOPTION = CCSIDMIXED, the CCSIDMIXED option will be determined from the current CHARNAME value.
- For SQLOPTION = CCSIDGRAPHIC, the CCSIDGRAPHIC option will be determined from the current CHARNAME value.
- For SQLOPTION = MCCSIDSBCS, the MCCSIDSBCS option will be 37.
- For SQLOPTION = MCCSIDMIXED, the MCCSIDMIXED option will be `0`.
- For SQLOPTION = MCCSIDGRAPHIC, the MCCSIDGRAPHIC option will be `0`.

Operator Response: Refer this message to your database administrator.

Database Administrator Response: Display the SYSTEM.SYSOPTIONS catalog table and find out whether:
- The DBCS option row is missing or contains a value other than YES or NO. If the row is missing, insert a row with the correct value. If the row is invalid, update the row with the appropriate DBCS option.
- The CHARSUB option row is missing or contains a value other than SBCS or MIXED. If the row is missing, insert a row with the correct value. If the row is invalid, update the row with the appropriate CHARSUB option.
- One or more of the CCSID option rows (CCSIDSBCS, CCSIDMIXED, CCSIDGRAPHIC, MCCSIDSBCS, MCCSIDMIXED, or MCCSIDGRAPHIC) is missing or contains an invalid CCSID value. Insert a row or update the row with the correct CCSID value. The CCSID values appropriate for this installation should be planned and implemented at the time of installation or migration.

ARI0105I **The row, where *column* = *name*, does not exist in the system catalog table *table-name*. The database manager assumes the internal default of *charname*.**

Explanation: The row for the character set, where *column* = *name* does not exist in the catalog table *table-name*. The character set specified in the CHARNAME parameter on SQLSTART is not correctly defined in all catalog tables.

System Action: Processing continues. The database manager uses the *charname* character set, which is usually be the last CHARNAME correctly specified.

Operator Response: Refer this message to your database administrator.

Database Administrator Response: Determine whether a wrong character set name is specified as the CHARNAME initialization parameter.
- If the initialization parameter is incorrect, restart the database system with a correct initialization parameter value.
- If the row is missing from the specified catalog table *table-name*, insert a row with the correct values into the table.

  Run the ARISDBMA EXEC. For more information on ARISDBMA, see the *DB2 Server for VM Program Directory*

  After inserting the row, restart the database manager, because some application programs may not run correctly when using another character set.

  For more information, see the *DB2 Server for VM System Administration* manual.

ARI0106E **The database manager found an incompatibility in the application server at initialization. The expected FORMAT ID is *format-id1*. The FORMAT ID found in the application server is *format-id2*.**

Explanation: The code running on the database manager is incompatible with the release level of the

database. *format-id1* is the expected format ID, and *format-id2* is the format ID of the current application server.

Level of Code	Application Server Format ID
Version 7 Release 1	9
Version 6 Release 1	9
Version 5 Release 1	8
Version 3 Release 5	7
Version 3 Release 4	7
Version 3 Release 3	7
Version 3 Release 2	7
Version 3 Release 1	6
Version 2 Release 2	5

System Action: The database manager is terminated.

Operator Response: Refer this message to your system programmer.

System Programmer Response: The procedures for installing the database manager code and updating application servers are described in the *DB2 Server for VSE Program Directory*. Installations that are simultaneously using multiple application servers and different database manager releases may accidentally pair a release of the database manager with an incompatible application server.

ARI0107E **The ROW, where the NAME = *name*, contains incorrect data in the *column* column in the system catalog table SYSCHARSETS. Data positions that are in error are indicated by the letter E in the strings that follow. Either the current system CHARNAME or INTERNATIONAL will be used.**

Explanation: For the character set specified in the CHARNAME initialization parameter, the EBCDIC character set classification table or the EBCDIC character set translation table in SYSTEM.SYSCHARSETS contains invalid values.

In the strings following the message, the positions in error are marked by an E. These positions are represented in hexadecimal numbers.

System Action: Processing continues. If the application server default CHARNAME is being changed and the new CHARNAME is in error, the value of the current application server default CHARNAME is used. If the current application server default CHARNAME is in error, then INTERNATIONAL will be used as the CHARNAME value. Message ARI0143I indicates the value of the application server default CHARNAME.

Operator Response: Refer this message to your database administrator.

Database Administrator Response: Display the row for the specified EBCDIC character set in the SYSTEM.SYSCHARSETS catalog table. Determine the correct values and update the row with correct values.

After correcting the row, restart the database manager because some application programs may not run correctly using another character set.

ARI0110E **CMS TODACCNT function failed. Return Code =** nn**.**

Explanation: DB2 Server for VM accounting support uses the CP DIAGNOSE instruction code X'70' to capture CPU time on a user basis. The database manager invokes CMS to issue the CP DIAGNOSE using the CMS TODACCNT function. This function has failed while attempting to issue the DIAGNOSE X'70'. The return code is the return code (in decimal) issued by the CMS TODACCNT function.

System Action: If this condition occurs, accounting support is shut off, and the database manager is ended.

Operator Response: Refer this message to your system programmer or database administrator.

ARI0113I **{LUWID** *luwid-value* **| DBSPACE** *dbspace-number* **| USERID** *userid***} has already been specified.**

Explanation: The specified LUWID, DBSPACE, or USERID was entered in a previous command of the same type (duplicate command) in the EXTEND input file.

System Action: The duplicate specification is ignored. Processing continues.

ARI0114E *keyword* **was specified more than once in the same statement.**

Explanation: A single EXTEND input file command specified a control keyword more than once.

keyword is a control keyword—ALL, TO, DBSPACE, USERID, or LUWID.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Remove duplicate keywords in the EXTEND input file command, then restart extended database initialization.

ARI0115E Unexpected string *string* was found.

Explanation: A line of an EXTEND input file command contains unrecognizable keywords, more than one control keyword, or extra input on a record.

string is an unrecognizable string, a keyword, or a control keyword found while parsing an EXTEND input file command.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the EXTEND input file command. Restart extended database initialization.

ARI0116E *string1* was expected but *string2* was found.

Explanation: While parsing EXTEND input file commands, one of the following caused an error:

- An unexpected string was found where a keyword, option, or reserved word should have been found.
- An end of record was expected and a string was found.
- An invalid keyword-option combination was found.

  For example, TIME or DATE was specified in a BYPASS UNDO WORK command, or ALL was specified in a ROLLBACK COMMITTED WORK command.

string1 is:

KEYWORD	One of the following keywords was expected: BYPASS, ROLLBACK, DISABLE, or ENABLE.
OPTION	One of the following options was expected: ALL, LUWID, TIME, DATE, DBSPACE, or USERID.
VALUE	One of the following values was expected: LUWID value, TIME value, DATE value, DBSPACE value, or USERID value.
reserved word	One of the following reserved words was expected: WHERE, UNDO, WORK, COMMITTED, DBSPACE.

If *string1* was either VALUE, UNDO, WORK, COMMITTED, or DBSPACE, *string2* may be END OF RECORD because an incomplete command was entered. If *string1* is either OPTION or WHERE, *string2* may be END OF FILE because an incomplete command was entered.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the EXTEND input file. Restart extended database initialization.

ARI0117E Range error: *number* is greater than the maximum number of possible *condition* (*max*).

Explanation: While parsing EXTEND input file commands, one of two conditions existed. *condition* identifies the condition.

- *condition* is LUWIDS. The total number of logical units of work included in the EXTEND input file (a composite of all commands) exceeded the maximum. Since BYPASS UNDO WORK affects log recovery, only 252 logical units of work are allowed.

  number is the number of logical units of work included in the EXTEND input file.

  max is 252, the maximum number of active logical units of work permitted at any one time.
- *condition* is DBSPACES. The number of DBSPACEs exceeded the maximum number of PUBLIC and PRIVATE DBSPACEs in the database.

  number is the number of a DBSPACE.

  max is the number of PUBLIC and PRIVATE DBSPACEs defined in the database.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the EXTEND input file. Restart extended database initialization.

ARI0118E {LUWID *luwid-value* | DBSPACE *dbspace-number* | USERID *userid*} contains invalid {HEX CHARACTERS | NUMBERS | ALPHANUMERICS} or is too long.

Explanation: While parsing EXTEND input file commands, one of the following conditions occurred:

- A logical unit of work (*luwid-value* is the logical unit of work identifier) contains invalid hexadecimal characters.
- A DBSPACE value contains characters other than positive numbers or zero.
- The logical unit of work, DBSPACE, or user ID value was too long.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the EXTEND input file. Restart extended database initialization.

ARI0119E Cannot {bypass | disable} DBSPACE 1.

Explanation: DBSPACE 1 can be neither disabled nor bypassed. The database manager must be able to use the database catalog located in DBSPACE 1.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Modify the EXTEND input file so that DBSPACE 1 is not disabled or bypassed. Restart extended database initialization.

ARI0120I DBSPACE *dbspace-number* **is already {ENABLED | DISABLED}.**

Explanation: A DISABLE or ENABLE DBSPACE command was specified for a DBSPACE that was previously disabled or enabled.

System Action: Initialization continues. The status of the specified DBSPACE is not changed.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Take note of the message.

ARI0121I DBSPACE nnn **is disabled.**

Explanation: This message is issued for each disabled DBSPACE in the database. DBSPACEs are disabled by the DISABLE DBSPACE command.

System Action: Processing continues.

System Programmer Response: You may wish to take note of which DBSPACEs are disabled.

ARI0122E The time *time-string* **and date** *date-string* **specified are later than the current time.**

Explanation: The starting date and time specified (or defaulted to) in the ROLLBACK COMMITTED WORK command are later than the current date and time. There can be nothing in the database log with such a date and time.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Determine the correct time and date. Correct the EXTEND input file. Restart extended database initialization.

ARI0123E The TO time and date combination is earlier than the FROM time and date combination.

Explanation: In a ROLLBACK COMMITTED WORK command, the TO time and date (taken in combination) must be greater than the FROM time and date (taken in combination).

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the EXTEND input file. Restart extended database initialization.

ARI0124E {OPEN | GET | CLOSE} error *input-file* **Reason Code =** nn**. Return Code =** cc**.**

Explanation: An error occurred in an attempt to access or close the EXTEND input file. The Reason Code value is the primary error code; the Return Code value is the secondary error code. For further explanation, look up the displayed er under either “Chapter 23. Sequential File Errors” on page 407 in this manual or SEQ FILE ERRORS in the HELP text.

input-file is the file with ddname ARIEXTND, the input file associated with the use of the EXTEND initialization parameter during database initialization.

System Action: For OPEN or GET errors, the database manager ends. For CLOSE errors, processing continues.

Operator Response: For OPEN or GET errors, look up displayed error codes to take corrective action. For CLOSE errors, no action is required.

System Programmer Response: Same as Operator Response.

ARI0125E The extend file is empty.

Explanation: The input file, ddname ARIEXTND, associated with the use of the EXTEND initialization parameter during database initialization contains no commands.

System Action: The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: You must either provide the required file commands or restart database initialization with EXTEND=N.

ARI0126E DBSS termination during {FORWARD | ROLLBACK | UNDO | REDO} processing.

Explanation:

- If FORWARD appears in the message, the database manager failed in the DBSS while accessing data in the database (DBSPACE data). The database manager was performing normal database activity for an application program or a terminal user. The following display output provides more information about the failing DBSS operation and identifies the database user who requested the operation.
- If ROLLBACK appears in the message, the database manager failed in the DBSS while trying to undo a database update previously made by the logical unit of work identified in the following display output. The database manager is performing log recovery for a logical unit of work that failed because the application, the terminal user, or the database manager initiated the ROLLBACK WORK process. The following display output provides more information about the failing DBSS operation and identifies the database user who requested the operation.
- If UNDO appears in the message, the database manager has failed in the DBSS during log recovery phase of warm start or restore from archive while trying to undo a database update previously made by the uncommitted logical unit of work identified in the following display output. The following display output provides more information about the failing DBSS operation and identifies the database user who requested the operation.

  Note: This failure can occur if a committed logical unit of work is rolled back by the ROLLBACK COMMITTED WORK command (refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for information about recovering from DBSS errors).
- If REDO appears in the message, the database manager has failed in the DBSS during the log recovery phase of warm start or restore from archive while trying to "redo" a database update previously made by the committed logical unit of work identified in the following display output. The following display output provides more information about the failing DBSS operation and identifies the database user who requested the operation.

System Action: The database manager ends.

Operator Response: Save this message and the subsequent display output and contact your system programmer immediately.

System Programmer Response: Make note of the information given in the message and the following display output, and refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for information about recovery procedures for DBSS errors.

ARI0127E The {TIME | DATE} parameter *string* contains a syntax error.

Explanation: The TIME or DATE parameter specified in a ROLLBACK COMMITTED WORK command contained a syntax error.

TIME Must be a total of 8 characters (hh:mm:ss). You must have two digits each for hour, minutes, and seconds with a delimiter (:) between each. Add leading zeros if necessary.

DATE Must be a total of 8 characters (mm-dd-yy). You must have two digits each for month, day, and year with a delimiter (-) between each. Add leading zeros if necessary.

string is the incorrect time or date character string.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the EXTEND input file. Restart extended database initialization.

ARI0128E A nonnumeric or invalid character was found in the {TIME | DATE} parameter *string*.

Explanation: An invalid character was found in the TIME or DATE parameter specified in a ROLLBACK COMMITTED WORK command.

TIME A non-numeric value for the hour, minutes, or seconds was specified. Or, an incorrect numeric value was specified. Valid numeric ranges for time are:

hh 00-23 mm 00-59 ss 00-59

DATE A non-numeric value for the month, day, or year was specified. Or, an incorrect numeric value was specified. Valid numeric ranges for date are:

mm 01-12 dd 01-31 yy 00-99

Note: Value for day must be valid for the month or year specified.

string is the invalid time or date string.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the EXTEND input file. Restart extended database initialization.

ARI0129E The combination of the ALL option with any other option is not valid.

Explanation: One of two conditions occurred:

- The ALL option and some other option (LUWID, USERID, or DBSPACE), were specified in the same BYPASS UNDO WORK command. No other option can be specified with ALL.
- More than one BYPASS UNDO WORK command was specified in the same EXTEND input file and one of these commands specified the ALL option. If ALL is used, only one BYPASS UNDO WORK command is allowed in the EXTEND input file.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the BYPASS UNDO WORK command. Restart extended database initialization.

ARI0130I The parameters specified are a subset of those used in a previous statement of the same type.

Explanation: The parameters ANDed together in a BYPASS UNDO WORK or ROLLBACK COMMITTED WORK command are a subset of parameters specified in previous commands of the same type. For example, the following duplicate commands will cause this message to be issued:

```
BYPASS UNDO WORK WHERE
USERID SMITH JONES
DBSPACE 2 4 5
   .
   .
   .
BYPASS UNDO WORK WHERE
USERID JONES
DBSPACE 4
```

System Action: The duplicate specification is ignored. Processing continues.

ARI0131E The TO option was found but no starting time or date was specified.

Explanation: A ROLLBACK COMMITTED WORK command was entered and the TO option was specified. No starting (from) TIME or DATE was specified.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the EXTEND input file. Restart extended database initialization.

ARI0132E DBSPACE *dbspace-number* has been selected for both enabling and disabling.

Explanation: The EXTEND input file contained conflicting commands to enable and disable the same DBSPACE.

System Action: EXTEND input file parsing completes. The database manager ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the EXTEND input file. Restart extended database initialization.

ARI0133I *yyyyyyyy*'s ownership of the application server *xxxxxxxx* has been revoked by CP. No new users can connect to this application server. The database manager will continue to process current users.

Explanation: This message is issued if two disjoint collections merge, and another virtual machine has identified *xxxxxxxx* as a resource, or if an authorized user has revoked the resource.

System Action: The database manager continues to process current users. No new users can access the application server (resource) *xxxxxxxx*.

Operator Response: Refer this problem to the SQL database administrator. The database manager should be shut down. A normal shutdown can still be done.

System Programmer Response: Resolve the conflict over the use of *xxxxxxxx* as a global resource name.

ARI0134I Application server xxxxxxxx **has been identified as a {global | local} resource.**

Explanation: The database manager has identified the specified application server as a global or local resource (specified with the DBMODE parameter). If no DBMODE parameter was specified at startup, then the database manager has used the IUCV *IDENT directory entry for **xxxxxxxx** as the default startup mode.

System Action: Database initialization continues.

ARI0135E The application server is not authorized to identify itself as resource xxxxxxxx**.**

Explanation: The database machine is not authorized to identify itself as the resource *xxxxxxxx* in an APPC/VM environment. The database machine's VM directory entry for IUCV *IDENT is incorrect. The causes are:

- The directory does not contain an IUCV *IDENT entry for the database (DBNAME(*xxxxxxxx*)) being used or, there is no IUCV *IDENT entry for RESANY.

- An attempt has been made to identify the database as a global resource (DBMODE=G) when it should be identified as a local resource.

System Action: The database manager has ended. Message ARI0042I is issued with a reason code.

Operator Response: Refer this message to your system programmer.

System Programmer Response: See "Chapter 21. DB2 Server for VM Reason Codes" on page 399 for additional information about the reason codes. Ensure that the IUCV *IDENT directory entry for the DB2 Server for VM database machine exists and is correct. Sample entries are:

IUCV *IDENT **xxxxxxxx** GLOBAL
IUCV *IDENT **yyyyyyyy** LOCAL
IUCV *IDENT RESANY LOCAL
IUCV *IDENT RESANY GLOBAL

If the entry only allows LOCAL identification for resource (database) **yyyyyyyy**, then an attempt to start the database manager as a GLOBAL resource (DBMODE=G) will fail.

ARI0136E Another virtual machine already owns the resource identifier xxxxxxxx**.**

Explanation: Another virtual machine is managing resource **xxxxxxxx**. Resource names must be unique.

System Action: The database manager has ended. Message ARI0042I is issued with a reason code.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Ensure that the resource name *xxxxxxxx* is unique within your collection.

See "Chapter 21. DB2 Server for VM Reason Codes" on page 399 for additional information about the reason codes.

ARI0137I Invalid installation defined date/time defaults on name.

Explanation: The specified date/time default *name* contains an invalid value. During installation, four date/time values were stored into the SYSTEM.SYSOPTIONS table. Their names and supported values are as follows:

DATE
The date format. One of the following must be used: ISO, JIS, USA, EUR, LOCAL.

TIME
The time format. One of the following must be used: ISO, JIS, USA, EUR, LOCAL.

LDATELEN
The local date length. It must be 0, or a number greater than 9 and less than 255.

LTIMELEN
The local time length. It must be 0, or a number greater than 7 and less than 255.

System Action: Processing continues with the default datetime values.

The default values supplied by the system are:

DATE	ISO
TIME	ISO
LDATELEN	0
LTIMELEN	0

Operator Response: Refer this message to your system programmer.

System Programmer Response: Update the SYSTEM.SYSOPTIONS table with the correct datetime values.

ARI0140D Enter DRRM function name and trace-level pair. Valid function names are *, DICT, GEN, PARSE, RDIIN, and CANCEL. Valid trace-level values are 0, 1, and 2.

Explanation: The TRACE ON command is being processed. The operator responded DRRM to prompting message ARI0087D. The database manager is prompting the operator to specify the functions of DRRM to be traced and the trace level for each function.

System Action:

- If * *n* is entered, where *n* is one of 0, 1, or 2, all DRRM functions will be traced at trace-level *n*.
- If any or all of DICT *n*, GEN *n*, PARSE *n*, RDIIN *n* are entered, the specified functions of DRRM will be traced at the specified trace-level *n* (where *n* is 1 or 2).
- If trace-level 0 is entered for a function it will not be traced (equivalent to not entering the function at all).
- If CANCEL is entered, TRACE command processing ends.

Note: This message is displayed in all environments. In environments in which the DRRM component is not active, however, the operator's response to this message is ignored.

Operator Response:

- If all DRRM functions are to be traced, enter * plus 1 or 2 for the trace level that you want (with a blank between * and number). For example, enter * 1 PARSE 2 to trace all DRRM functions except PARSE at level 1 and traces PARSE at level 2.
- If only selected DRRM functions are to be traced, enter first a function name followed by its trace-level value, and then a second function name, if you want one, with its trace-level value. One or more blanks

must be keyed between each function name and each trace-level value entered. If you enter a function name more than once, only the last entry is effective. If you want to terminate the TRACE command, enter CANCEL.

ARI0141D **Enter WUM trace-level or CANCEL. Valid trace-level values are 0, 1, and 2.**

Explanation: The TRACE ON command is being processed. The operator has responded WUM to prompting message ARI0087D. The database manager is prompting the operator to specify the desired trace level for WUM.

System Action:

- If *n* is entered, where *n* is 1 or 2, the WUM component will be traced at trace-level *n*.
- If trace-level `0` is entered for the component, it will not be traced (equivalent to not entering WUM in response to the message ARI0087D).
- If CANCEL is entered, TRACE command processing ends.

Notes:

1. This message is displayed in all environments. In environments where the WUM component is inactive, however, the operator's response to this message is not used.

Operator Response:

- Enter the desired trace level.
- If TRACE command termination is desired, enter CANCEL.

ARI0142D **Enter DSC function name and trace-level pair. Valid function names are *, AGENT, COM, and CANCEL. Valid trace-level values are 0, 1, and 2.**

Explanation: The TRACE ON command is being processed. The operator responded DSC to prompting message ARI0087D. The database manager prompts the operator to specify the DSC function to trace and the desired trace level.

System Action:

- If you enter AGENT, the agent handling function of the database manager is traced at the specified trace level (n).
- If you enter COM, the communications function of the database manager is traced at the specified trace level (n).
- If you enter CANCEL, TRACE command processing ends.

Note: If you enter trace level 0 for a function, the function is not traced. Entering trace level 0 is equivalent to not specifying the function at all.

Operator Response: If you want to terminate the TRACE command, enter CANCEL.

ARI0143I **The application server has been initialized with the following values: CHARNAME = *option1*, DBCS = *option2*, CHARSUB = *option3*, CCSIDSBCS = *option4*, CCSIDMIXED = *option5*, and CCSIDGRAPHIC = *option6*.**

Explanation: The CHARNAME startup parameter (if specified) or the CHARNAME value used previously, and the values in SYSTEM.SYSOPTIONS for DBCS and CHARSUB, are used to initialize the values reported by this message. The values in SYSTEM.SYSOPTIONS are updated so that the next time the application server is started and the CHARNAME startup parameter is not specified, the values reported by this message are then similar to those received now.

System Action: Initialization continues.

ARI0145E **DBSPACE name *dbspace-number* does not exist or is invalid.**

Explanation: The user failed to pass in a valid DBSPACE name (for example, System DBSPACE name) for processing.

System Action: The IDENTIFICATION process ends unsuccessfully.

User Response: Rerun the IDENTIFICATION process and provide a valid DBSPACE name as input.

ARI0146I **Warning: The command line was truncated at column 80.**

Explanation: ARIYE04 only parses 80 characters. On VM, the command can be up to 130 characters. On VSE, an operator command can only be 80 characters. Rather than extending ARIYE04 for VM, a warning message is generated on VM if the operator enters more than 80 characters.

System Action: Process the truncated command.

Operator Response: Shorten the command line if valid data appears beyond column 80.

ARI0150E **There is no connection for** showid**.**

Explanation: A connection to the application server for the specified *showid* that is an agent, user ID, or LUWID cannot be found. Sources of the problem include:

- An incorrect user ID was entered.
- An incorrect agent number was entered (not numeric or not in the range 1 through NCUSERS).
- An incorrect LUWID was entered.
- The agent, user ID, or LUWID is not connected to the application server at the instant the SHOW CONNECT command was entered.

System Action: The SHOW command processing is ended.

Operator Response: Be sure that the specified agent, user ID, or LUWID is correct and then reenter the SHOW CONNECT command.

Note: You may receive the same result if the requested agent, user ID, or LUWID is not connected to the application server when the command is reentered.

ARI0152I **Processing DBSPACE:** *owner.dbspace-name***.**

Explanation: This message is written out to the screen to indicate that the specified DBSPACE *owner.dbspace-name* is being processed.

System Action: Processing continues.

ARI0153I **The database manager ignores TRACEBUF =** *size* **parameter when trace is not activated.**

Explanation: A nonzero *size* for the TRACEBUF startup parameter has been specified, and no subcomponent have been chosen to be traced (TRACxxxx startup parameters must indicate 1 or 2 for a specific subcomponent to be traced).

System Action: The system does not use the TRACEBUF parameter, and no trace buffer is allocated. The startup continues.

Operator Response: No response is required if the operator does not want any trace output. If trace output is expected, then the database manager must be restarted with the proper values of the TRACxxxx startup parameters. Alternatively, the **TRACEBUF n** parameter in the *TRACE ON* operator command can be used to start the trace in MUM.

ARI0154I **A** *size* **kilobyte trace buffer has been allocated. Maximum contiguous memory left =** *amount* **kilobytes.**

Explanation: The trace buffer of the *size* as shown has been created.

System Action: The trace is started. Trace records are stored into the trace buffer.

Operator Response: If the *amount* maximum contiguous memory left is too small (see the note below) then the operator can consider one of the following actions:

- Turn the trace off using the TRACE OFF command and then reissue the TRACE ON with a smaller amount of storage specified for the TRACEBUF parameter.
- Turn the trace off using the TRACE OFF command and then reissue the TRACE ON with TRACEBUF = 0 or without TRACEBUF (the trace buffer is not used and the trace records are written directly into the trace file).
- Start the database manager with more memory available.

Note: Even after the database manager has been started, the storage requirements are dynamic and workload dependent. See the *DB2 Server for VM System Administration* manual for virtual storage requirements of the DB2 Server for VM system.

ARI0155E **There is not enough memory for trace buffer allocation. Contiguous memory requested =** *needed_storage* **kilobytes. Maximum contiguous memory available at this time =** *available_storage* **kilobytes. Trace is not started.**

Explanation: The memory *needed_storage* required for the trace buffer allocation is not available.

System Action: If TRACEBUF was specified in the TRACE ON operator command, the trace is not started.

If TRACEBUF was specified as a database startup parameter, the database manager is ended.

Operator Response: There are several possible actions:

- Reissue the TRACE ON command with smaller amount of storage specified for the TRACEBUF parameter.
- Reissue the TRACE ON command with TRACEBUF = 0 (the trace buffer is not used and the trace records are written directly into the trace file).
- Start the database manager with more memory available.
- If TRACEBUF was specified as a database startup parameter, start the database manager with a smaller amount of storage specified for TRACEBUF, or remove TRACEBUF from the parameter list.

ARI0156I **The trace buffer was written to the trace file.**

Explanation: The contents of the trace buffer was flushed out to the trace file. This can happen as a result of the TRACE OFF operator command or during a database system shutdown.

ARI0157I **The trace buffer is empty.**

Explanation: An attempt to write the trace buffer into the trace file failed. The trace buffer is empty, because an incorrect TRACE specification (component, subcomponent, authorization ID, agent number or a level) was entered, or the workload that was run did not hit an activated trace point.

Operator Response: Reissue the TRACE ON

command in MUM, or restart the database manager with the correct values of startup parameters. Repeat the workload to be traced.

ARI0158E **RESID X'*resid*' is an invalid registered *resid-class* TPN.**

Explanation: The database manager has detected an attempt to use an invalid registered TPN to identify a resource managed by this application server. The following are acceptable values for *resid*:
- Any registered DRDA TPN.
- Any registered DDM TPN.

Note: Please refer to the information on SNA service transaction programs list in the *SNA Transaction Programmer's Reference* manual for LU Type 6.2 and the information on DRDA rules in the *Distributed Relational Database Architecture Reference* manual for more details. The following are acceptable values for **resid-class**:
- C'DRDA'
- C'DDM'

System Action: The database manager is terminated.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Note the information in this message and refer to the *DB2 Server for VM System Administration* manual for information about changing a database name and resource identifier.

ARI0159D **The specified CHARNAME is different than the current system CHARNAME. This will force the system CCSID values to be changed. Do you wish to change the system CHARNAME?**

Enter 0(No) to continue using the current CHARNAME of old-charname**, or**
1(Yes) to change the system CHARNAME to new-charname**, or**
111(Quit) to cancel your request and end SQLSTART processing.

Explanation: The value of the CHARNAME parameter specified in SQLSTART is different from the current CHARNAME value. Changing the current CHARNAME will change the current default CCSID values. Also, the CCSID attribute of the character columns of the system catalog tables will be updated to the new default CCSID. The actual data stored in the system catalog tables will ***not*** be converted from the current default CCSID to the new default CCSID. Only the attribute of the character columns will change. The CCSID attribute of the character columns of any user tables will ***not*** be updated to the new default CCSID.

The default CHARNAME of a system should be changed only once: at installation or migration. If it is changed after system objects have been created in the system catalog tables, object names not conforming to the rules for ordinary identifiers may contain characters that do not have the same hexadecimal representation in the new CCSID and some data may be displayed incorrectly. For the rules about ordinary identifiers, see the *DB2 Server for VSE & VM SQL Reference* manual. For more information on the CHARNAME parameter, see the *DB2 Server for VM System Administration* manual.

System Action:
- If you answer `0` (No), the current default CHARNAME and CCSID values are used and SQLSTART processing continues.
- If you answer 1 (Yes), the default CHARNAME will change. CCSID values associated with the new CHARNAME will:
  - Become the new default CCSID values for the database manager
  - Replace the current CCSID attribute for the character columns in the system catalog tables.
- If you answer 111(Quit), SQLSTART processing ends, and no changes are made.

Database Administrator Response: Unless you are installing or migrating a database or reconfiguring the database to work with different CCSIDs, enter `0`(No) to continue database initialization using the current CHARNAME.

ARI0160D **Enter CONV trace-level or CANCEL. Valid trace-level values are 0, 1, and 2.**

Explanation: The TRACE ON command is being processed. The operator has responded CONV to prompting message ARI0087D. The database manager is prompting the operator to specify the desired trace level for CONV.

System Action:
- If *n* is entered, where *n* is 1 or 2, the CONV component will be traced at trace-level *n*.
- If trace-level `0` is entered for the component, it will not be traced (equivalent to not entering CONV in response to the message ARI0087D).
- If CANCEL is entered, TRACE command processing ends.

Operator Response: Enter the desired trace level. If TRACE command termination is desired, enter CANCEL.

ARI0161E **PROTOCOL=AUTO cannot be specified because the DRDA facility has not been installed for the application server.**

Explanation: PROTOCOL=AUTO was specified as an SQLSTART parameter.

System Action: The database manager is ended.

Operator Response: Refer this message to your database administrator.

Database Administrator Response: Determine whether or not you require your application server to service DRDA requesters. If so, install the DRDA facility following the steps in the *DB2 Server for VM System Administration* manual. If not, do not specify PROTOCOL=AUTO as an SQLSTART parameter.

ARI0162D Enter STG trace-level or CANCEL. Valid trace level values are 0 and 1.

Explanation: The TRACE ON command is being processed. The operator responded STG to prompting message ARI0087D. The database manager is prompting the operator to specify the trace level for STG.

System Action:
- If 1 is entered, the STG component is traced at level 1.
- If 0 is entered, the STG component is not traced.
- If CANCEL is entered, TRACE command processing ends.

Operator Response: Choose your response from "System Action", and enter `1` or `0` or `CANCEL`.

ARI0164I An error occurred when dropping routine *routine-name*. NUCXDROP reason code = *reason-code*.

Explanation: The routine *routine-name* cannot be dropped. Refer to the *VM/ESA: CMS Command Reference* for the explanation of the NUCXDROP *reason-code*.

User Response: Report the problem to your system programmer.

System Programmer Response: Drop the routine.

ARI0165I All parameters are ignored for *string*.

Explanation: The command, statement, or operation that you entered, such as SHOW DBCONFIG, contained parameters, but none were required. The system does not use the parameters.

System Action: Processing continues, and all parameters are ignored.

ARI0172E A Sync Point Manager is not available.

Explanation: The parameter "SYNCPNT=Y" was specified at startup; however, the CRR recovery server was not available. This message is following by message ARI0176I.

System Action: The database manager resets the SYNCPNT parameter to "N" and the database manager is unable to use VM/CRR for handling DRDA sync point operations. As a result, this database may not be a participant in DRDA logical units of work that update multiple remote databases. However, the database manager is **still** able to process SQLDS protocol requests and DRDA requests that do not involve a sync point manager (for example, DRDA logical units of work that only update a single remote database).

System Programmer Response: If this database manager is to be a participant in DRDA logical units of work that update multiple remote databases, a CRR recovery server must be installed. See the *VM/ESA: Installation Guide* for instructions on installing a CRR recovery server. After the CRR recovery server is intalled, restart the database manager with SYNCPNT=Y.

ARI0173I The CRR recovery server's TPN and log name and the database manager's log name have been reset. On the next resynchronization, the database manager's log status will be cold.

Explanation: The **RESET CRR LOGNAMES** command was entered to initialize the following fields stored in the database manager's log to hex zeros:
- The CRR recovery server's TPN
- The CRR recovery server's log name
- The database manager's log name

The next time the database manager is started with the SYNCPNT parameter set to "Y" and *resynchronization initialization* is performed, the database manager will consider the log status to be *cold*.

For more information about *resynchronization initialization*, see the *DB2 Server for VM System Administration* manual.

System Action: The RESET CRR LOGNAMES command processing is ended.

Database Administrator Action: When the database manager is next started with the SYNCPNT parameter set to "Y", *resynchronization initialization* will fail if there are any DRDA2 in-doubt logical units of work. To clean these up, issue the SHOW INDOUBT command to list them and then use the FORCE and/or RESET INDOUBT commands to clear them.

ARI0174I Resynchronization initialization has started. Date: *date* Time: *time*

Explanation: The database manager was started with the SYNCPNT parameter set to "Y". It has started the *resynchronization initialization* process with the CRR recovery server at the date and time specified. This process involves an exchange of log names and notifies the CRR recovery server of its readiness to accept resynchronization communications.

For more information about *resynchronization*

initialization, see the *DB2 Server for VM System Administration* manual.

System Action: Processing continues.

System Programmer Response: Watch to make sure that no error messages appear after message ARI0174I.

ARI0175I Resynchronization initialization has completed. Date: *date* **Time:** *time*

Explanation: The database manager previously started the *resynchronization initialization* process with the CRR recovery server. This message signals the operator that this processing has ended at the date and time specified. For more information about *resynchronization initialization*, see the *DB2 Server for VM System Administration* manual.

System Action: If no errors occurred during *resynchronization initialization*, DRDA2 activity may commence with this database, including units of work that require a sync point manager. If message ARI0176I is displayed before this message, then errors have occurred during *resynchronization initialization*.

System Programmer Response: If errors occurred during *resynchronization initialization*, then an error message will have been displayed that indicated the error. See the explanation for that message and resolve the error.

ARI0176I The SYNCPNT parameter has been reset to N.

Explanation: One of three situations has occurred:

1. The database is performing *resynchronization initialization* and encountered an error that resulted in the SYNCPNT parameter being changed from "Y" to "N". If this is the case, then message ARI0174I will have been issued previously and another message follows it that indicates the error that occurred.
2. The database is processing a DRDA2 logical unit of work and is attempting to register the unit of work with VM/CRR using the DMSREG CSL routine, but the registration failed. If this is the case, then message ARI0180E (with function *DMSREG*) will have been issued previously.

System Action: The database manager is unable to use VM/CRR for handling DRDA sync point operations. As a result, this database may not be a participant in DRDA logical units of work that update multiple remote databases. However, the database manager is **still** able to process SQLDS protocol requests and DRDA requests that do not involve a sync point manager (for example, DRDA logical units of work that only update a single a single remote database).

System Programmer Response: Message ARI0176I is preceded by a message indicating the error that occurred. Fix the problem indicated in this prior message and restart the database.

ARI0177E CRR recovery server at TPN *tpn* **has provided a new log name resulting from a cold start. Some LUWID(s) cannot be automatically resolved by resynchronization.**

Explanation: The database manager is performing *resynchronization initialization* and has determined that the CRR recovery server has been cold started. Furthermore, the database manager has some in-doubt logical units of work that require resynchronizing with the CRR recovery server.

tpn indicates the transaction program name for the CRR recovery server with which the database manager is communicating.

For more information about *resynchronization initialization*, see the *DB2 Server for VM System Administration* manual.

System Action: The database manager is unable to use VM/CRR for handling DRDA sync point operations. As a result, this database may not be a participant in DRDA logical units of work that update multiple remote databases. However, the database manager is **still** able to process SQLDS protocol requests and DRDA requests that do not involve a sync point manager (for example, DRDA logical units of work that only update a single remote database).

Database Administrator Action: Contact the CRR recovery server operator to determine the reason for the cold start. Then resolve the disposition of each in-doubt unit of work as follows:

- Issue the SHOW CONNECT command to determine the location of the remote DRDA application requester.
- Contact the other Database Administrator at that remote site to determine the proper state of each logical unit.
- Use the FORCE and RESET INDOUBT commands to clear up the in-doubt logical units of work and then restart the database.

Once all in-doubt units of work have been resolved, restart the database manager.

Note: If CRR has coldstarted and a DRDA2 requester has any outstanding in-doubt logical units of work, those in-doubt units of work at the requester will need to be resolved before the requester can make a DRDA2 connection to the DB2 Server for VM server.

ARI0178E **An Exchange Log Name's *{Request | Reply}* sent by CRR recovery server at TPN *tpn* contained a log name which does not match the current *{database manager | CRR recovery server}* log name. Log name in *{Request | Reply}*: *log name* Current Log name: *log name***

Explanation: The database manager has detected a log name mismatch. This has occurred in one of the following circumstances:

1. The database manager is processing *resynchronization initialization* and has received an exchange log names reply from the CRR recovery server. A log name given by the CRR recovery server in the reply does not match a corresponding log name found in the database manager's log. In this case:
   - *Reply* indicates that the database manager received an exchange log names reply.
   - *tpn* indicates the transaction program name for the CRR recovery server with which the database manager is communicating.
   - *{database manager | CRR recovery server}* indicates if the unmatched name is for the database manager or for the CRR recovery server.
   - log name in *Reply* indicates the log name found in the reply.
   - Current log name indicates the log name found in the database manager's log.
2. The database manager is processing *resynchronization recovery* and has received an exchange log names request from the CRR recovery server. The log name given by the CRR recovery server in the request does not match the log name found in the database manager's log. In this case:
   - *Request* indicates that the database manager received an exchange log names request.
   - *tpn* indicates the transaction program name for the CRR recovery server with which the database manager is communicating.
   - *{database manager | CRR recovery server}* indicates if the unmatched name is for the database manager or for the CRR recovery server.
   - log name in *Request* indicates the log name found in the request.
   - Current log name indicates the log name found in the database manager's log.

For more information about *resynchronization initialization* or *resynchronization recovery*, see the *DB2 Server for VM System Administration* manual.

System Action: If the message comes during *resynchronization initialization*, then the SYNCPNT parameter is reset from "Y" to "N" and the database manager is unable to use VM/CRR for handling DRDA sync point operations. As a result, this database may not be a participant in DRDA logical units of work that update multiple remote databases. However, the database manager is **still** able to process SQLDS protocol requests and DRDA requests that do not involve a sync point manager (for example, DRDA logical units of work that only update a single remote database).

If the message comes during *resynchronization recovery*, then any logical unit of work remains unresolved. That is, it remains IN-DOUBT.

Database Administrator Action: If the message comes during *resynchronization initialization*, then the database manager's participation in sync points must be delayed until the error condition is resolved. Contact the recovery server operator to determine the reason for the log name mismatch. The database may have been restored from an archive which resulted in the log name mismatch, or the recovery server might be using the wrong log, and should be restarted with the correct log. If the correct log cannot be supplied, both partners must be coldstarted. Note the following:

- The "RESET CRR LOGNAMES" operator command may be used to reset the CRR recovery server's luname, tpn, and log name and force a DB2 Server for VM status of cold.
- If the "FILESERV CRRLOG" CRR command is used to force a CRR coldstart and a DRDA2 requester has outstanding in-doubt logical units of work, those in-doubt units of work at the reguester will need to be resolved before the requester can make a DRDA2 connection to the DB2 Server for VM server.

If the message comes during *resynchronization recovery*, contact the recovery server operator to determine the reason for the log name mismatch. A couple of possible reasons are:

- The database manager's NETID, LUNAME or TPN may have changed, which resulted in a different log name.
- An archive from another DB2 Server for VM database may have been restored in this database.

If this is the case then the database must be coldstarted. Note that the "RESET CRR LOGNAMES" operator command may be used to reset the CRR recovery server's luname, tpn and log name and force a DB2 Server for VM log status of cold. If the recovery server is using the wrong log and cannot locate the correct log, some units of work may need to be manually FORCED at the database manager.

ARI0179E **An Exchange Log Name's Reply sent by CRR recovery server at TPN *tpn* contained an error status.**

Explanation: The database manager is processing *resynchronization initialization* and has received an exchange log names reply from the CRR recovery server. This reply contains an error status rather than a normal status. The CRR recovery server has processed the exchange log names request sent by the database

manager and one of the following situations has occurred:

1. The database manager sent an exchange log names request with a log status of cold, but the CRR recovery server detected some units of work that were in-doubt and required resolving. In this case, the CRR recovery server issues CMS message 3311E to the recovery server operator.
2. The database manager sent an exchange log names request with a log status of warm, but the CRR recovery server detected that a log name in the rquest does not match its current log name. In this case, the CRR recovery server issues CMS message 3312E to the recovery server operator.

tpn indicates the transaction program name for the CRR recovery server with which the database manager is communicating.

System Action: The SYNCPNT parameter is reset from "Y" to "N" and the database manager is unable to use VM/CRR for handling DRDA sync point operations. As a result, this database may not be a participant in DRDA logical units of work that update multiple remote databases. However, the database manager is **still** able to process SQLDS protocol requests and DRDA requests that do not involve a sync point manager (for example, DRDA logical units of work that only update a single remote database).

Database Administrator Action: Contact the recovery server operator to determine if the CRR recovery server issued CMS message 3311E or 3312E. With their help, resolve any in-doubt units of work or the log name mismatch. The "RESET CRR LOGNAMES" operator command may be used to reset the CRR recovery server's luname, tpn and log name and force a DB2 Server for VM log status of cold.

Note: If CRR has coldstarted and a DRDA2 requester has any outstanding in-doubt logical units of work, those in-doubt units of work at the requester will need to be resolved before the requester can make a DRDA2 connection to the DB2 Server for VM server.

ARI0180E **CRR function** *function* **failed. Return Code:** *return code* **Reason Code:** *reason code* **Identifier:** *identifier***.**

Explanation: The database manager has issued a CMS command or callable services library (CSL) routine while interfacing with VM/CRR, and that command or routine has returned an error status.

function indicates the CMS command or CSL routine for which the error occurred. If function has the format "DMSxxxxx", then it was a CSL routine. Otherwise, it was a CMS command.

return code indicates the return code from the CMS command or CSL routine.

reason code indicates the reason code from the CSL routine. This has no meaning for CMS commands.

identifier indicates which occurrence of the CMS command or CSL routine for which this error occurred.

System Action: If the function was DMSCHREG, the change CRR registration has failed and the unit of work is rolled back. Otherwise, the SYNCPNT parameter is reset from "Y" to "N" and the database manager is unable to use VM/CRR for handling DRDA sync point operations. As a result, this database may not be a participant in DRDA logical units of work that update multiple remote databases. However, the database manager is **still** able to process SQLDS protocol requests and DRDA requests that do not involve a sync point manager (for example, DRDA logical units of work that only update a single remote database).

System Programmer Response: If it was a a callable services library (CSL) routine that failed, look up the return code and reason code in the *VM/ESA: CMS Application Development Reference* manual and resolve the error. Otherwise, it was a CMS command. In this case, look up the return code in the *VM/ESA: CMS Command Reference* manual and resolve the error. Note that the Reason code has no meaning for CMS commands.

Database Administrator Action: Once the error has been resolved, set the SYNCPNT parameter back to "Y" by restarting the database manager.

ARI0181I **The CRR recovery server is resynchronizing work with the database manager. Date:***date* **Time:***time*

Explanation: Resynchronization occurs if two-phase commit processing is interrupted by a resource failure. The CRR recovery server initiates the *resynchronization recovery* function to ensure consistent completion of the sync point by all registered resources for which data was logged. Using information stored in its log, the CRR recovery server determines which resource managers (such as DB2 Server for VM) should be included in the recovery and allocates APPC conversations with them.

Message ARI0181I is a signal to the database manager operator that the CRR recovery server has initiated *resynchronization recovery* with the database manager and has allocated an APPC conversation with it. The date and time indicate when this occurred.

This message is followed by message ARI0182I which identifies the *transaction id* and *LUW id* for the unit of work being recovered.

For more information about *resynchronization recovery*, see the *DB2 Server for VM System Administration* manual.

System Action: *Resynchronization recovery* processing continues.

Database Administrator Action: Watch to make sure that no error messages appear after message ARI0181I.

ARI0182I **Transaction ID:** *transaction id* **LUWID:** *LUW id*

Explanation: Message ARI0182I is an informational message that identifies the *transaction id* and *LUW id* for the unit of work involved in *resynchronization recovery*. It follows messages ARI0181I and ARI4023I.

System Action: Database manager processing continues.

Database Administrator Action: Watch to make sure that no error messages appear after message ARI0182I.

ARI0183E **The Sync Point Manager has asked to {COMMIT | ROLLBACK} this LUW but the database manager has no memory of it.**

Explanation:

On VM, the database manager is working with the CRR recovery server to recover the LUWID identified in message ARI0181I. Specifically, the CRR recovery server is asking the database manager to either COMMIT or ROLLBACK this unit of work, as identified in the message. However, the database manager could not find any indication that this unit of work needs recovery.

Note: If the operator used the "RESET INDOUBT" command to erase the database manager's memory of a unit of work, and then the CRR recovery server involves this unit of work in *resynchronization recovery*, then message ARI0183E can come up.

Note: Message ARI0183E can come up in the following circumstances:

- If the operator used the "RESET INDOUBT" command to erase the database manager's memory of a unit of work, and then the CRR recovery server involves this unit of work in resynchronization recovery.
- A unit of work at the database manager is "read-only" and a DRDA2 commit is interrupted for any reason.

System Action: On VM, the database manager sends back a normal completion reply indicating a ROLLBACK, and *resynchronization recovery* is terminated for this unit of work.

Database Administrator Action: Contact the CRR operator and have them issue the "CRR QUERY LUWID" command to obtain more information about this unit of work. Then, make sure that this unit of work was properly recovered.

ARI0184A **The Sync Point Manager has asked to {COMMIT | ROLLBACK} this LUW but the FORCE command was previously used to {ROLLBACK | COMMIT} it.**

Explanation: Coordinated Resource Recovery (CRR) is attempting to resynchronize the status of a logical unit of work with the database manager. It is asking the database to either commit or rollback this LUW. However the operator had previously used the database manager FORCE command to force the LUW into the opposite state.

System Action: The LUW remains unsynchronized with the sync point manager. The LUW still appears when the SHOW INDOUBT command is issued.

Operator Response: In order to clear out this LUW from the SHOW INDOUBT display , it is necessary to use the RESET INDOUBT command.

Database Administrator Action: Manual intervention is necessary to ensure that the LUW is in a consistent state at all sites where the LUW has been distributed. This may require intervention at this database manager, or possibly at another database manager. Manual intervention could mean manually fixing the data or possibly restoring an archive.

Refer this message to your DB2 Server for VM database administrator.

ARI0197E **A communication error occurred.**

Function: *function-code*
Return Code: *return-code*
Reason Code: *reason-code*
CMSIUCV Reg15: *X'reg15'*
CC: *condition-code*
IPRCODE: *X'iprcode'*
IPCODE: *X'ipcode'*
IPWHATRC: *X'ipwhatrc'*
IPAUDIT: *X'audit'*

Explanation: The database manager is engaged in a conversation with the CRR recovery server while performing *resynchronization initialization* or *resynchronization recovery*, but an APPC communications error occurred. For further information on determining the cause of the error, see the following sections:

- See “VM IUCV Communication Codes” on page 400
- See “Chapter 22. Debugging Communication Errors” on page 403

System Action: The processing of *resynchronization initialization* or *resynchronization recovery* ends.

System Programmer Response: Take corrective action based on the **Explanation**.

ARI0198E Reset CRR Lognames cannot be used when resynchronization <*Initialization* | *Recovery*> **is active.**

Explanation: The operator entered the "RESET CRR LOGNAMES" command to reset CRR log name information to hex zeros. However, the database is currently performing *resynchronization initialization* or *resynchronization recovery* which depends on this CRR log name information. Therefore, the operator command is disallowed.

<*Initialization* | *Recovery*> indicates whether the database was processing *resynchronization initialization* or *resynchronization recovery*.

System Action: Command processing is terminated.

Operator Response: Retry the command after message ARI0175I is displayed, signalling *resynchronization initialization* completion, or ARI4023I is displayed, signalling *resynchronization recovery* completion.

Chapter 3. ARI0200–ARI0299, ARI2000–ARI2048 DBSS Messages

ARI0200E Sizes of dual logs are unequal.

Explanation: During database generation (invoking the application server with the SQLDBINS or SQLDBGEN EXEC) or during log file redefinition/reformatting (invoking the application server with the SQLLOG EXEC), dual logging was defined (initialization parameter DUALLOG=Y). The database manager has determined that the two log minidisks are not the same size. This is not permitted.

System Action: The application server is ended.

Operator Response: Return the console output with the output listing.

System Programmer Response: You can circumvent this problem by invoking the application server with a single log. To do so:

- Reinvoke the SQLDBINS or SQLDBGEN EXEC and reply that you want only one log minidisk when prompted, or
- Reinvoke the SQLLOG EXEC with the LOG1 parameter and omit the LOG2 parameter.

You can correct this problem by proceeding as follows:

Redefine the VM Block I/O minidisks for the logs so that they are equal in size. They are always equal in size if they are on the same device type and you specify the same number of cylinders (count-key-data devices) or blocks (FB-512 devices). Then reexecute the SQLLOG EXEC with the LOG1 and LOG2 parameters or the SQLDBINS or SQLDBGEN EXEC and reply that you want two log minidisks when prompted.

ARI0201E Storage pool *n* is full.

Explanation: The physical storage in storage pool *n* has reached a level at which the system cannot function. If this error was the result of a DROP TABLE or DBSPACE command, then the storage pool did not contain enough shadow pages to process the DROP command.

System Action: In most cases, processing of the SQL statement ends. In some cases, the application server ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Define an additional Block I/O minidisk for a new DBEXTENT. Then run the ADD DBEXTENT (SQLADBEX EXEC) function to add the DBEXTENT to the storage pool that is full.

Notes:

1. If ARI0201E occurred during DROP TABLE or DBSPACE processing and the application server ended, you need to check the catalog table SYSDROP after you warm start the application server. If there is an entry in SYSDROP for the table or dbspace that was being dropped when the application server was terminated, any subsequent drop commands may cause the application server to be terminated with message ARI0201E.

   To correct this problem, you need either to add a DBEXTENT to the storage pool in which the table's DBSPACE resides, or to reorganize the DBSPACE.

   When you add a DBEXTENT to correct this problem, you should next issue an ACQUIRE and then a DROP of any size DBSPACE. This will cause the database manager to clean up any entries in SYSTEM.SYSDROP that were left over during the original DROP TABLE or DBSPACE failure.

   For information on how to reorganize the DBSPACE, see the *DB2 Server for VSE & VM Database Administration* manual.
2. You may not have to enlarge the storage pool. Under certain conditions, the message ARI0201E occurs when the processing of the current operation has used only half of the storage pool.

   A bypass is possible if LOGMODE is not *N* and the failing operation is one of:

CREATE INDEX	DROP TABLE
DROP DBSPACE	UPDATE
INSERT	DELETE
DATALOAD	RELOAD TABLE
RELOAD DBSPACE	RELOAD PROGRAM
RELOAD PACKAGE	REORGANIZE INDEX
SCHEMA	DROP INDEX

   Try running the command in single user mode with *LOGMODE=N*.

ARI0202I Warning: Storage pool *n* is short of storage.

Explanation: Storage pool *n* has reached the cushion specified at startup using the database SOSLEVEL initialization parameter (SOSLEVEL x TOTAL PAGES IN POOL / 100). Either logging was inactive, or logging is active but further checkpoints will not correct this situation.

System Action: Processing continues.

© Copyright IBM Corp. 1987, 2000

Operator Response: Notify your system programmer.

System Programmer Response: Enter the SHOW DBEXTENT command to determine the amount of space left in the storage pool. You may need to add more DBEXTENTs to the storage pool specified in the message. (If needed, use the ADD DBEXTENT function (SQLADBEX EXEC).)

ARI0203E {PAGE|DIRECTORY} buffer pool is full.

Explanation: When running the database manager, more pages of data or DIRECTORY blocks were simultaneously required in the buffers than there were buffers in the pool.

System Action: The application server is ended.

Operator Response: Save the console output for the system programmer.

System Programmer Response: Increase the number specified for the initialization parameter NPAGBUF (for page buffers) or NDIRBUF (for directory buffers), and restart the application server.

ARI0204D Restore from archive has been invoked. Current database will be destroyed.

Enter RESTORE - to continue, or CANCEL - to cancel.

Explanation: The application server was started with STARTUP=R. This causes the contents of the database to be replaced (restored) by the database copy on a database manager archive file. The restored database is then updated for any logging which has occurred since the last database archive.

System Action: The system waits for your reply.

If the reply is RESTORE, the database manager continues and replaces the database with the database copy on the archive file.

If the reply is CANCEL, the application server is ended without modification to the database.

Any other reply causes the message to be reissued.

Operator Response: If the database is to be restored from a database manager archive file, reply RESTORE.

If the database is not to be restored from a database manager archive file, reply CANCEL to end the application server.

System Programmer Response: Inform the operator of your intentions when restoring a database, including any mounting of database manager archive tapes.

ARI0205I You must user-archive the directory and *n* DBEXTENT(S) but not the logs!

Explanation: The database operator has issued an SQLEND UARCHIVE command. This tells the database manager that you are shutting down normally and then backing up your database with the backup facility of your choosing.

n is the number of DBEXTENTs in your database. You are also warned not to backup your logs. Backing up and then restoring your logs would make it impossible to restore your database changes since the user-archive was taken.

System Action: The database manager will write a record on the log indicating that a user-archive is being attempted and then shut down normally.

Operator Response: After the application server terminates, backup your directory and **n** DBEXTENTs with any suitable backup facility. Take special care to note whether you have backed up successfully or not as you will be prompted with message ARI0206D at the next database startup.

ARI0206D Did the user-archive of the directory and *n* DBEXTENT(S) execute successfully? Enter 0(No) or 1(Yes).

Explanation: The SQLEND UARCHIVE command was used the last time the application server was shut down. You must tell the database manager if you have successfully archived your database. *n* is the number of DBEXTENTS in your database.

System Action: If you reply 1 for YES, the database manager records on the log that a user-archive was successfully done, and it frees log space. If you reply `0` for NO, the database manager will prompt you with message ARI0207D. If you do not reply 1 or `0`, message ARI0052E will be displayed and message ARI0206D will be redisplayed.

Operator Response: Reply 1 if you are sure that you backed up your directory and **n** DBEXTENTs. Reply `0` if you are unsure about your archive and you will then be prompted with message ARI0207D.

ARI0207D Do you want to continue (without user-archive) or cancel (redo the user-archive)? Enter CONTINUE or CANCEL.

Explanation: You replied `0` to message ARI0206D. You may terminate the application server if you wish to redo your user-archive.

System Action: If you reply CONTINUE, the database manager will write a record to the log indicating that no user-archive was done and processing will continue.

If you reply CANCEL, the database manager will not write any record to the log. The last user-archive log

record is still in effect. The application server will terminate to allow you to redo your user-archive. At the next application server startup, you will be prompted with message ARI0206D.

If you do not reply CONTINUE or CANCEL, message ARI0052E is displayed and message ARI0207D is redisplayed.

Operator Response: Reply CONTINUE if you wish to continue using the database manager without user-archiving the database. Reply CANCEL if you wish to terminate and redo the user-archive.

ARI0208D Did the user-restore of the directory and *n* DBEXTENT(S) execute successfully? Enter 0(No) or 1(Yes).

Explanation: The application server was started with STARTUP=U to indicate that a user-restore of the database was done before startup. This message is issued to verify that you performed the restore.

System Action: If you reply 1, the database manager will reapply the log activity since the user-archive was taken. All committed logical units of work will be reflected in the database.

If you reply 0, the application server will terminate to allow you to redo the user-restore or startup with the correct STARTUP parameter.

If you do not reply 1 or 0, message ARI0052E is displayed and message ARI0208D is redisplayed.

Operator Response: Reply 1 if you are sure that your user-restore was successful. Otherwise, reply 0. If the STARTUP parameter was incorrectly set to U, simply restart the application server with the correct STARTUP parameter. If you are unsure of the user-restore, you may redo it after the application server ends and then restart the application server with STARTUP=U.

ARI0210I The extend file input did not generate summary statistics.

Explanation: Either the control keywords specified in the EXTEND input file had no match in the log or the control keywords were matched but the logical units of work found were read-only (queries).

System Action: The database manager displays message ARI0213D.

Operator Response: Respond CANCEL to ARI0213D. Refer this message to your system programmer.

System Programmer Response: Check the control keywords in the EXTEND input file and adjust as needed. Restart extended database initialization.

ARI0211I The requested BYPASS or ROLLBACK operation may affect the following logical units of work:

Explanation: If the ROLLBACK COMMITTED WORK or BYPASS UNDO WORK operation completes, the specified logical units of work may operate on the same tables as did the rolled back or bypassed logical unit of work. Depending on which operation is performed on the DBSPACE or the table, this may lead to other problems such as updating a table or row which no longer exists.

System Action: The database manager displays message ARI0214I (impacted summary) and message ARI0213D (pause). Refer this message to your system programmer.

System Programmer Response: Use the summary information to determine if the specified action is to be done. Instruct the operator whether to reply CONTINUE or CANCEL to message ARI0213D.

Sample Output Sequence:

```
ARI0211I The requested BYPASS or ROLLBACK
         operation may affect the
         following logical units of work:
ARI0214I LUWID=7BA USER ID=JONES
         DBSPACE=8 TABLE ID=-32765(8003)
```

ARI0212I Summary information of the records to {be bypassed | roll back}:

LUWID = *&2B*. **User ID =** *&3*.
Date = *&4*. **Time =** *&5*.

Explanation: Summary information will be displayed for each logical unit of work bypassed or rolled back due to a BYPASS UNDO WORK or ROLLBACK COMMITTED WORK command. This message displays statistics for this logical unit of work. It is followed by message ARI0237I which displays the operations and the number of their occurrences which will be rolled back or bypassed if instructed to continue.

However, summary information will not be displayed for the logical units of work which have already been rolled back or bypassed due to a previous BYPASS UNDO WORK or ROLLBACK COMMITTED WORK command.

System Action: The database manager issues:

- Messages ARI0237I (operations performed), ARI0214I (affected logical units of work), and ARI0213D (operator decisions), or
- Messages ARI0237I (operations performed) and ARI0213D (operator decisions).

Operator Response: Refer this message to your system programmer.

System Programmer Response: Use the summary information to determine if the specified action is to be done. Instruct the operator whether to reply

CONTINUE or CANCEL to message ARI0213D.

Sample Output:

```
ARI0212I Summary information of the
         records to roll back:
         LUWID=7B9      User ID=SMITH
         Date=12-01-85  Time=13:12:45
```

ARI0213D Enter CONTINUE to continue initializing or CANCEL to end.

Explanation: Summary information, if any, has been displayed on the console concerning the logical units of work that will be bypassed or rolled back due to a BYPASS UNDO WORK or ROLLBACK COMMITTED WORK command if the database manager is instructed to continue.

System Action: The database manager waits for the operator reply. Message ARI0052E is displayed if you do not reply CONTINUE or CANCEL. This message displays the first 8 characters of the reply and states that the response was not valid. Message ARI0213D will be repeatedly displayed until CONTINUE or CANCEL is entered.

Operator Response: Analyze the summary information and respond accordingly. The response CONTINUE causes database initialization to continue with the specified EXTEND input file commands carried out. The effects of the commands will be written into the log before applying the records. They will remain in the log even if a failure occurs while applying the records. The response CANCEL causes application server termination, leaving the database as it was before startup.

If no summary information was displayed and an EXTEND input file was included, the EXTEND input file commands will *not* be executed. Something may be wrong with the EXTEND input file commands. In this case, respond CANCEL to bring down the application server and check the EXTEND input file commands. Refer the summary messages to your system programmer.

System Programmer Response: Analyze summary messages and decide whether the actions are what was expected. Determine whether to continue or cancel initialization.

ARI0214I LUWID=*&1B*. User ID=*&2*. DBSPACE=*&3*. TABLE ID=*&4A*. (*&5B*.)

Explanation: This message is displayed immediately after message ARI0211I. It displays the logical unit of work value, the user ID value, the DBSPACE number, and the table identification number (if applicable) of a logical unit of work which may be affected if the specified BYPASS UNDO WORK or ROLLBACK COMMITTED WORK command operation is continued.

The user ID value can be blank. For example, a logical unit of work was started after a system checkpoint. A second logical unit of work begins and commits before the first logical unit of work does. The first logical unit of work acts on the changes done by the second logical unit of work and commits. If the second logical unit of work is rolled back for any reason, the first logical unit of work will be affected and USERID will be blank because the logical unit of work started after the checkpoint.

&1B. is a logical unit of work value.

&2. is a user ID value or blank.

&3. is a DBSPACE number.

&4A. is a table identification number or 0 if the table identification number is not applicable (no table updated).

&5B. is a hexadecimal table identification number or 0 if the table identification number is not applicable (no table updated).

System Action: Message ARI0213D (operator decision) is displayed.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Use the summary information to determine if the specified action is to be done. Instruct the operator whether to reply CONTINUE or CANCEL to message ARI0213D.

ARI0215E Operator command attempts to access a disabled DBSPACE.

Explanation: The operator or an ISQL user is issuing an operator command which is attempting to access a disabled DBSPACE. Disabled DBSPACEs cannot be accessed. The operator command that might cause this message is SHOW DBSPACE. DBSPACEs are disabled by DISABLE DBSPACE commands in an EXTEND input file.

System Action: Operator command processing ends.

Operator Response: Change the command so it does not attempt to access a disabled DBSPACE.

User Response: Same as Operator Response.

ARI0216I Directory verify function completed {with | without} discrepancies.

Explanation: This message is issued after the completion of the directory verify function (an SQLEND command option). This message will be preceded by specific error messages for each discrepancy found (if any).

System Action: Termination process continues.

Operator Response: If the directory verify function ended with discrepancies, refer this message and the preceding messages to your system programmer. If it completed without discrepancies, termination will

continue and you will be prompted for operator action as needed.

Note: If ARCHIVE was specified and discrepancies were found, no database archive will be taken. If UARCHIVE was specified and discrepancies were found, a user-archive should not be taken. Any implicit log archives will be performed due to ARCHIVE or UARCHIVE. If LARCHIVE was specified, log archiving will be performed, but you should correct the discrepancies as soon as possible.

System Programmer Response: If the database directory verify function completed with discrepancies, note the information in the messages preceding this message and refer to directory verification in *DB2 Server for VSE & VM Diagnosis Guide and Reference* for recovery procedures. If it completed without discrepancies, no action is required of the system programmer.

ARI0217E Physical page X'*n1*' assigned to DBSPACE *n2*. Page X'*n3*' was not allocated in the allocation bit map.

Explanation: The directory verify function has detected that the physical page number identified by **X'***n1*' that has been allocated to logical page **X'***n3*' of DBSPACE number *n2* is reflected as not being allocated in the allocation bit map. This is an internal database error.

Note: The DBSPACE number *n2* corresponds to the DBSPACENO column in the catalog table SYSTEM.SYSDBSPACES.

System Action: The database manager will continue verifying all PUBLIC and PRIVATE DBSPACEs and end normally.

Operator Response: Refer this message to your system programmer.

Note: If ARCHIVE was specified, the DB2 Server for VM database archive function will not be invoked. If UARCHIVE was specified, a user-archive should not be taken. Any implicit log archives will be performed due to ARCHIVE or UARCHIVE. If LARCHIVE was specified, the log will be archived, but you should correct the discrepancies as soon as possible.

System Programmer Response: Make note of the information given in this message and refer to directory verification in *DB2 Server for VSE & VM Diagnosis Guide and Reference* for recovery procedures.

ARI0218E DBSPACE *n1*, page X'*n2*' is allocated to physical page X'*n3*' of storage pool *n4* instead of storage pool *n5*.

Explanation: The directory verify function has detected that physical page **X'***n3*' of storage pool *n4* was allocated to physical page **X'***n2*' of DBSPACE *n1*, which belongs to storage pool *n5*. This is an internal database error.

Note: The DBSPACE number *n1* corresponds to the DBSPACENO column in the catalog table SYSTEM.SYSDBSPACES.

System Action: The database manager will continue verifying all PUBLIC and PRIVATE DBSPACEs and end normally.

Operator Response: Refer this message to your system programmer.

Note: If ARCHIVE was specified, the archive function will not be invoked. If UARCHIVE was specified, a user-archive should not be taken. Any implicit log archives will be performed due to ARCHIVE or UARCHIVE. If LARCHIVE was specified, the log will be archived, but you should correct the discrepancies as soon as possible.

System Programmer Response: Make note of the information given in this message and refer to directory verification in *DB2 Server for VSE & VM Diagnosis Guide and Reference* for recovery procedures.

ARI0219E An unexpected internal error occurred during SYSDROP processing.

Explanation: The process which cleans up entries on the SYSDROP catalog table has been invoked. An unexpected error has occurred during this process, most likely a database error. This message is displayed on the system console.

Note: In addition to this error, ARI0504I and ARI0505I are also displayed to report error information in the SQLCA. See "Chapter 18. DBSS Return Codes" on page 271 for definitions of the DBSS return codes.

System Action: The application server terminates the SYSDROP processing but continues executing.

Operator Response: Refer this message and its contents to your system programmer.

System Programmer Response: The SYSDROP process can be FORCEd by another logical unit of work which attempts to commit a DROP TABLE or DROP DBSPACE. If this failed to release the resources, report this message and its contents to the designated support group for your installation.

ARI0220E FORCE command has insufficient number of parameters.

Explanation: The database FORCE operator command was entered without parameters.

System Action: The command processing is ended.

Operator Response: Reenter the FORCE operator command with the appropriate operands.

ARI0221E Agent identifier *n* in FORCE command is invalid.

Explanation: The agent identifier **n** does not exist in the system. It must be an integer value which ranges from 1 to the database initialization parameter NCUSERS.

System Action: The command processing is ended.

Operator Response: Verify the agent identifier by using the SHOW ACTIVE operator command output. Then reenter the FORCE operator command with the correct agent identifier.

ARI0222E Parameter *parm value* in FORCE command is invalid.

Explanation: The parameter value identified by *parm value* specified in the FORCE operator command is not valid. Acceptable parameters are:

ROLLBACK
DISABLE

The DISABLE parameter is optional, but, if used, it must follow the ROLLBACK parameter (that is, ROLLBACK DISABLE).

System Action: The command processing is ended.

Operator Response: Check the specified parameter, and reenter the FORCE operator command with correct values.

For more information about the FORCE operator command, refer to the *DB2 Server for VSE & VM Operation* manual.

ARI0223I Agent *n* in FORCE command has no work to force.

Explanation: The agent identified by *n* does not now have any work to be forced off the system.

Note: The SHOW ACTIVE operator command may have indicated that there was work to force. However, the work ended before the FORCE operator command.

System Action: The command processing is ended.

Operator Response: Do a SHOW ACTIVE to verify the status of agent **n**. The agent will be either not in work (NIW) or active with a new logical unit of work.

ARI0225E System operator must issue SHOW ACTIVE, SHOW CONNECT or SHOW SYSTEM command prior to FORCE command.

Explanation: This message is issued for one of these reasons:

- The database FORCE command was entered without first doing a SHOW ACTIVE, SHOW CONNECT or SHOW SYSTEM from the database operator console. One of these operator commands must be issued to obtain the correct agent numbers to specify in the FORCE command.

  or

- The FORCE command specified an agent that completed its LUW after the SHOW ACTIVE, SHOW CONNECT or SHOW SYSTEM command was entered.

Note that if parameters are used on the SHOW CONNECT command, they must result in the display of active agents to enable the FORCE command to work for the displayed agents.

System Action: The command processing is ended.

Operator Response: Enter the SHOW ACTIVE, SHOW CONNECT or SHOW SYSTEM command from the operator console to obtain the correct agent numbers, then reenter the FORCE command for those agents desired.

ARI0226I FORCE {ROLLBACK | COMMIT} is already scheduled for {agent | Rmtuser} *n*.

Explanation: The agent or Rmtuser identified by *n* is being rolled back or committed by the system. This action may have been scheduled because of an application request, a previous FORCE command, or system action.

System Action: The command processing is ended.

ARI0229E Too many FORCE command input parameters.

Explanation: The FORCE command was entered with more than three parameters.

System Action: The command processing is ended.

Operator Response: Reenter the FORCE command with correct number of parameters.

ARI0230I FORCE {ROLLBACK | COMMIT} {*with* | *without*} disable scheduled for agent *n* because of {*log full* | *operator request* | *resynchronization recovery*}.

Explanation: The database manager FORCE command was initiated for the agent identified by *n* due to the condition indicated by the message ("log full," "operator

request," or "resynchronization recovery"). If the agent was in-doubt, the action could be COMMIT or ROLLBACK. Otherwise the action is ROLLBACK.

System Action: The COMMIT or ROLLBACK is scheduled. The actual COMMIT or ROLLBACK may not occur immediately. When the agent begins processing and detects that the FORCE was scheduled, it will then begin the requested process. DISABLE, if specified on the FORCE command, will cause the link to the application to be disconnected after the COMMIT or ROLLBACK process completes.

ARI0231E The database log contains unresolved units. Please start the database manager with NCUSERS = *&1*..

Explanation: In a VSE guest sharing environment, the DB2 Server for VM recovery process detected some logical units of work which were prepared for COMMIT or ROLLBACK by a subsystem (for example, the DB2 Server for VSE Online Resource Adapter). Because of a system or subsystem failure, the subsystem did not COMMIT or ROLLBACK the logical units of work. These logical units of work cannot be recovered (committed or rolled back) because either:

- The application server has been started in single user mode, or
- The application server has been started in multiple user mode but the number of logical units of work is greater than the number of concurrent users (specified by the database initialization parameter NCUSERS).

&1. is the number of logical units of work prepared for COMMIT or ROLLBACK and is the minimum value required for initialization parameter NCUSERS.

System Action: The system ends with a limit error (message ARI0039E).

Operator Response: Restart the application server in multiple user mode (SYSMODE=M) and NCUSERS greater than or equal to &1..

ARI0232E The log contains unresolved work for user ID *&1*..

Explanation: In a VSE guest sharing environment, the application server recovery process detected that a logical unit of work for user ID &1. was prepared by a subsystem for COMMIT or ROLLBACK (for example, the DB2 Server for VSE Online Resource Adapter). The COMMIT or ROLLBACK request did not occur because of a system or subsystem failure. Operator intervention may be required to resolve the logical unit of work.

&1. is the user ID which was established for the logical unit of work.

System Action: The database manager rebuilds the agent as it was prior to the system or subsystem failure. The agent is ready to be resolved by the FORCE command or by the DB2 Server for VSE online resource adapter restart resynchronization support. After the agent is resolved, it is made available for general use by applications or subsystems.

Operator Response: The agents requiring resolution will be resolved during the next execution of the DB2 Server for VSE online resource manager CIRA or CIRB transactions.

If the automatic restart resynchronization environment is not available, after database initialization is completed, use the SHOW ACTIVE command to obtain more information about the logical unit of work. Check with the installation database administrator to determine whether a FORCE n COMMIT command or a FORCE n ROLLBACK command should be entered.

System Programmer Response: In general, a FORCE n COMMIT command should be entered. This is because the application declared its intent as commit either by an SQL COMMIT WORK or by a CICS/VSE SYNCPOINT (no ROLLBACK). The only exception to the general rule is if the application was updating multiple resources (for example, the DB2 Server for VM database and a DL/I database). In that case, the ROLLBACK option is available for when the update was not committed by another subsystem like DL/I.

To determine the nature of the application, the SHOW ACTIVE command displays the following information:

- USERID = &1. is prepared for COMMIT OR ROLLBACK.
- COORDINATOR is the CICS/VSE subsystem id.
- RESOURCE ADAPTER is the DB2 Server for VSE resource adapter id.
- TRANSACTION is the CICS/VSE TRANSID for the application.
- CICS SIGNON is the CICS/VSE signon userid (if any).
- TERMINAL is the CICS/VSE terminal id (if any).

If the TRANSACTION identifier is ISQL, only the DB2 Server for VM database was being updated. Thus, FORCE COMMIT should be entered because the ISQL user requested COMMIT and ISQL updates only the DB2 Server for VM database.

ARI0233E Invalid FREECLASS for page X'*n1*' of DBSPACE *n2* (Block X'*n3*').

Explanation: The database manager has detected that the FREECLASS setting for page **X'***n1*'** of DBSPACE *n2* is not within the valid range. This is an internal database error.

Note: The DBSPACE number *n2* corresponds to the DBSPACENO column in catalog table SYSTEM.SYSDBSPACES.

Note: The BLOCK number *n3* refers to the Page Map Table block that is being processed.

System Action: The database manager will continue verifying all PUBLIC and PRIVATE DBSPACEs and end normally.

Operator Response: Refer this message to your system programmer.

Note: If ARCHIVE was specified, the archive function will not be invoked. If UARCHIVE was specified, a user-archive should not be taken. Any implicit log archives are performed due to ARCHIVE or UARCHIVE. If LARCHIVE was specified, the log will be archived, but you should correct the discrepancies as soon as possible.

System Programmer Response: Make note of the information given in this message and refer to directory verification in *DB2 Server for VSE & VM Diagnosis Guide and Reference* for recovery procedures.

ARI0234E Invalid physical page X'*n1*' for page X'*n2*' of DBSPACE *n3*.

Explanation: The database manager has detected that physical page value **X'***n1*'** for logical page **X'***n2*'** of DBSPACE *n3* is outside the currently defined physical pages of the database. This is an internal database error.

Note: The DBSPACE number *n3* corresponds to the DBSPACENO column in catalog table SYSTEM.SYSDBSPACES.

System Action: The database manager will continue verifying all PUBLIC and PRIVATE DBSPACEs and end normally.

Operator Response: Refer this message to your system programmer.

Note: If ARCHIVE was specified, the archive function will not be invoked. If UARCHIVE was specified, a user-archive should not be taken. Any implicit log archives are performed due to ARCHIVE or UARCHIVE. If LARCHIVE was specified, the log will be archived, but you should correct the discrepancies as soon as possible.

System Programmer Response: Make note of the information given in this message and refer to directory verification in *DB2 Server for VSE & VM Diagnosis Guide and Reference* for recovery procedures.

ARI0235I LARCHIVE command rejected because a {log | database} archive is being taken.

Explanation: An LARCHIVE command was entered by the database operator while either a log archive or a database archive was being taken.

System Action: The log archive or database archive is completed and the LARCHIVE command is ignored.

ARI0236I The database manager rejects log archive request because log archiving is not enabled.

Explanation: An SQLEND LARCHIVE or an LARCHIVE command was entered by the operator, but the database manager is not running with LOGMODE=L.

Note: Although it is valid to enter SQLEND ARCHIVE if database archiving is not enabled (LOGMODE=Y), it is not valid to enter SQLEND LARCHIVE if log archiving is not enabled.

Note: SHUTDOWN is a valid synonym for SQLEND.

System Action: The database manager will process the SQLEND command ignoring the LARCHIVE request. If the LARCHIVE command was entered, command processing ends.

Operator Response: If you were instructed to enter the LARCHIVE command or SQLEND LARCHIVE, notify your system programmer. Otherwise, no action is required.

System Programmer Response: Ensure that the operator is not requested to log archive when LOGMODE is not L.

ARI0237I *&1. &2.*

Explanation: This message is issued after message ARI0212I. It specifies the affected DBSPACEs and the operations which will be bypassed or rolled back.

Sample Output:

```
ARI0237I  DBSPACE 7
ARI0237I   5 INSERTS
ARI0237I   2 CREATE TABLES
ARI0237I  DBSPACE 12
ARI0237I   10 UPDATES
```

System Action: Message ARI0214I (affected logical units) or message ARI0213D (operator decision) will be issued.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Use the summary information to determine if the specified action is to be done. Instruct the operator whether to reply CONTINUE or CANCEL to message ARI0213D.

ARI0238I *database* **archive** *mm-dd-yy hh:mm:ss*

Explanation: The database manager is listing the restore set you should apply to the current database archive file. The message identifies the external labeling of the database archive file for the restore set.

Message ARI0239I displayed this information when the database archive file was created.

System Action: ARI0261I is issued for each log archive to be applied to this database.

ARI0239I **External labeling of this archive is: Type: {database | log} archive Timestamp:** *mm-dd-yy hh:mm:ss*

Explanation: Because there are three different kinds of archives, and each may require multiple tape volumes, you should externally label the tapes for use in restoring the database. When the database manager requests the operator to mount the tapes to contain the archive, this message is displayed to help in this external labeling.

This message may be followed by ARI0252I, which may indicate that the LOG ARCHIVE is being written to disk. In that case, it is impossible to externally label the archive medium.

The database manager also uses the label internally to determine the log archives which follow a database archive.

This message is displayed at the following times:

- When the database manager creates a database archive file
- When the database manager creates a log archive file
- When the application server is ending to allow the user to create a database archive (using the SQLEND UARCHIVE operator command)
- When the database manager is ready to read the next log archive file during the restore process.

The database manager will display this same information to identify the database archive file and the required log archive files during the restore process. See messages ARI0238I and ARI0261I.

System Action: The archive or restore process continues.

Operator Response: When creating archives, use the displayed information to externally label the archive tapes. If you are doing a LOG ARCHIVE TO DISK, you may want to record the archive details in an operational record book.

When restoring the database, use the displayed information to assist you in mounting the correct log archive tape. (This step is unnecessary if restoring a LOG ARCHIVE FROM DISK.)

ARI0240I **Restoring log disk.**

Explanation: The application server has been started with STARTUP=R or STARTUP=U; the log archive input volume has been readied; and the log files are being restored.

System Action: The database manager proceeds with the restore process.

ARI0241E **The log archive being read is not the log archive requested.**

Explanation: The volume mounted after message ARI0239I was displayed is not the log archive tape file requested. Possible causes include:

- The mounted (and opened) volume for ddname ARILARC does not contain a log archive file.
- The mounted (and opened) volume for ddname ARILARC is a log archive volume, but is not the log archive volume described in message ARI0239I. It may not be the first volume of the log archive file.

If the log archive is on disk, the disk file read after message ARI0239I was displayed is not the log archive disk file requested. Possible causes include:

- A file with the same identifier as the log archive disk file exists ahead of the archive file in the CMS search order of the database machine.
- A duplicate file identifier was used in the creation of a log archive on disk. The log requested archive was overwritten by a newer one.

System Action:

The mounted (and opened) volume is closed. If the close fails, message ARI0298E is displayed and the application server is terminated. Otherwise, message ARI0295A is redisplayed to allow the operator to mount the required log archive file and continue the restore process.

Operator Response:

1. Determine why the correct volume was not mounted.
2. Ensure that the information given in message ARI0239I matches the information used to externally label the log archive volume at log archive creation time.

   Note: This only applies if the operator labeled the archive tapes with the information displayed at archive creation time.
3. Ensure that the correct log archive volume is mounted. This is the log archive file identified by the previous message ARI0239I.
4. Reply with the correct cuu to message ARI0295A to continue the log archive restore process.

System Programmer Response: If using log archiving to tape, ensure that your database operator and system

operator follow procedures that allow proper identification of the log archive volume at creation time and proper log archive volume mounting at log archive restore time.

If using log archiving to disk, ensure that your database operator follows procedures that guarantee a unique file identifier is used for each archive file.

ARI0242E **Previously LOGMODE was L and now LOGMODE is {Y|N}. To start with LOGMODE = {Y|N}, perform SQLEND with LARCHIVE, perform COLDLOG, and restart with LOGMODE = {Y|N}. Note: Only in a restore process do log archives taken up to the switch in LOGMODE from L to {Y|N} remain useful.**

Explanation: If the operator wants to change from LOGMODE=L to LOGMODE=Y or N, this message outlines the procedure to follow. In this case, the message also alerts the operator that after SQLEND LARCHIVE archives the current log and the operator does a COLDLOG, a restore process that restores a database archive (which may include the log archive just taken in its restore set) cannot restore log archives taken past this log archive.

If the operator wants to continue running the database manager with LOGMODE=L, the operator can ignore the procedure outlined in this message and restart the application server with LOGMODE=L.

System Action: The application server is ended.

Operator Response: Either restart the application server with LOGMODE=L so that log archives continue to be useful in a restore process, or restart the application server with LOGMODE=Y or LOGMODE=N after performing the procedure outlined in the message. Note that COLDLOG is performed with the SQLLOG EXEC. See *DB2 Server for VM System Administration* for more information on the SQLLOG EXEC.

ARI0243I **Previously LOGMODE was L and now LOGMODE=A. Note that log archives taken up to this point are useful until another database archive is taken.**

Explanation: If the LOGMODE switch from L to A is desired, this message alerts the operator that once a database archive is taken with LOGMODE=A, a restore from a back-level database archive may only apply log archives up to the last log archive before this switch to LOGMODE=A.

System Action: Database initialization continues.

Operator Response: No action is required if the LOGMODE switch from L to A is desired. Otherwise, end the application server and restart it with the desired LOGMODE.

ARI0244E **To continue with LOGMODE=L, you must first perform an initial database archive.**

Explanation: You have been running with a LOGMODE other than L. If you want LOGMODE=L, you must first perform an initial database archive since the log indicates that there is no database archive upon which a log archive can be based. A cold log resets LOGMODE to Y because a COLDLOG ends all current restore sets.

System Action: The application server ends.

Operator Response: Start the application server with LOGMODE=Y and take a database archive. Then, restart the application server with LOGMODE=L.

ARI0245E **Warning: Do not do a user-archive until you correct discrepancies in the directory.**

Explanation: The SQLEND UARCHIVE command was entered with the DVERIFY parameter. The preceding messages display the discrepancies found in the directory.

System Action: The application server ends.

Operator Response: Refer the message to your system programmer.

System Programmer Response: Make note of the information given in the preceding messages and refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* for recovery procedures for directory verification errors.

ARI0246D **The above information describes the log archive about to be done. Enter either CONTINUE to proceed using the output medium indicated, or CHANGE to change this medium.**

Explanation: A log archive is being created for one of the following reasons:

- You entered the LARCHIVE operator command.
- You entered the SQLEND LARCHIVE operator command.
- You entered the SQLEND UARCHIVE operator command, and the database manager is creating a log archive (LOGMODE=L).
- The database manager has implicitly initiated the creation of a log archive because of the ARCHPCT initialization parameter.
- You are archiving or restoring the database with LOGMODE=L, and the database manager is archiving the current log.

This message is displayed after ARI0239I and ARI0252I. If ARI0252I displays question marks (? characters) for the value of any portion of the medium, message

ARI0263D is displayed after this message to indicate which archive medium should be used. In other instances, the database manager prompts the operator to confirm whether or not the log archive should be written to the tape address or the disk file specified in ARI0252I.

System Action: When the operator types CONTINUE at the `CONTINUE or CHANGE` prompt, the log archive is written to the tape address or the disk file specified in ARI0252I. The database manager assumes that the tape is mounted or the disk is accessed.

If the operator types CHANGE, ARI0263D is displayed allowing the destination of the archive to be changed.

If the database manager does not prompt the operator for a response, ARI0263D is always displayed after this message to force the operator to change the destination of the archive.

Operator Response: Take one of the following actions when the database manager prompts you to type CONTINUE or CHANGE:

If you want the archive to be written to a destination other than the one indicated in ARI0252I, type CHANGE.

If you want the archive to be written where indicated in ARI0252I:
- If to disk, type CONTINUE.
- If to tape, request (with the CP MSG OP command) that the CP operator select an available tape unit, mount and ready a scratch tape volume, and attach the tape unit to your virtual machine with the required virtual tape unit address. (TAPn is unit 180-7 if n= 0 to 7, and TAPn is unit 288-F if n= 8 to F). When the CP operator notifies you that these actions have been taken, reply with CONTINUE.

If this is a redisplayed message and the system error message DEVICE *device name* DOES NOT EXIST is displayed before this redisplayed message, issue a CP LINK command from the database operator console to link back to the detached disk to store the log archives.

ARI0247D You are in the restore process. There are log archives that you could use in this process but LOGMODE=A. Do you want LOGMODE=L instead? Enter 1(Yes) to change LOGMODE to L, or 0(No) to keep LOGMODE=A.

Explanation: You have started the application server with STARTUP equal to R or U and LOGMODE equal A. The database has been restored. There are log archives which could be applied to the database. The database manager is giving you the chance to switch the LOGMODE value to L so that the log archives may be applied.

System Action: If you reply 1(YES), the database manager will switch the LOGMODE to L and apply all applicable log archives as if you had started the application server with LOGMODE=L. If you reply 0(No), the database manager will issue message ARI0264D.

Operator Response: Reply 1(YES) if you want any log archives applied to the database just restored. Reply 0(No) if you do not want these log archives applied.

ARI0248E No operand on command.

Explanation: The operator or ISQL terminal user entered a COUNTER or RESET command without specifying either * or the counter names.

System Action: The command processing is ended.

User Response: Reenter the command and specify either * for all counters or specific counter names for the desired counters.

Operator Response: See User Response.

ARI0249E *parameter* **is an invalid RESET or COUNTER parameter.**

Explanation: The operator or ISQL terminal user entered a COUNTER command or RESET command with an operand value that was not a system counter name or a valid reset parameter.

System Action: The name is bypassed by the command processor. Any valid counters following the invalid name will be displayed or reset.

User Response: Check the spelling of the submitted name. Reenter the command with the correctly spelled counter name or reset parameter. Issue COUNTER * for a list of all acceptable counter names.

You cannot reset the DASDIO, DASDREAD, and DASDWRIT counters because they are sums of other counters.

To reset the values displayed by the SHOW STORAGE command, enter `RESET HIGHSTOR`. (HIGHSTOR is not a counter and is not reset by `RESET *`.)

If your database manager has DB2 VM Data Spaces Support installed, you can obtain the valid storage pool counter names for the RESET POOL command. Enter `COUNTER POOL *`. To display internal counter names, enter `COUNTER INTERNAL *`.

Operator Response: See User Response.

ARI0250D The above information describes the next log archive to be used in the restore process. Enter either: CONTINUE to restore this log archive, STOP SYSTEM to interrupt this restore process or END RESTORE to end this restore process.

Explanation: Messages ARI0239I and ARI0252I contain a description of the next log archive file to be applied.

This is the same information given at the time the log archive file was created. You are given three options with this message:

1. Reply CONTINUE: The log archive identified in message ARI0239I will be applied to the database. If the archive is on tape, and ARI0252I specifies a tape cuu, mount the tape before replying.
2. Reply STOP SYSTEM: The restore process is suspended and the application server stops. The restore process will start where it left off if you warm start the application server with STARTUP=W and LOGMODE=L.
3. Reply END RESTORE: The restore process is ended. Also, you will never be able to restore the remaining log archives to the database archive you have just restored. This is because the remaining log archives are no longer a part of the restore set for this database archive in the history area. However, if you go to a back level database archive, these log archives may be applied as usual.

System Action: One of the following actions will occur:

- Reply was CONTINUE: The restore process continues.
- Reply was STOP SYSTEM: The restore process is suspended. The application server stops.
- Reply was END RESTORE: If you have not applied the first log archive file to your database archive file, you will receive message ARI0264D. If you have applied at least one log archive file to your database archive file, the database manager will complete initialization and begin application processing.

Operator Response: In all but very special cases, you will reply CONTINUE. As mentioned in the Explanation, you may need to mount a tape before making this reply. You may want to reply STOP SYSTEM if you are using an EXTEND input file and need to change the EXTEND commands. There is no danger in using STOP SYSTEM. The only case in which you may want to use END RESTORE is if you want to use the log archives up to a certain point and never restore the remaining log archives. END RESTORE should only be used in extreme cases as it could result in a permanent loss of data. Do not reply END RESTORE without consulting your system programmer.

System Programmer Response: Be sure that you understand the consequences of replying END RESTORE.

ARI0251D The database archive currently mounted is not the most recent database archive known to the database manager. Enter either: IGNORE to continue, or CANCEL to restart the restore process with the most recent database archive.

Explanation: The database archive that was just restored is not the latest database archive record in the history area. This prompt gives you the chance to terminate the database manager if you wish to mount a different database archive.

System Action: If you reply IGNORE and there are no log archive files that go with this database archive file, message ARI0264D is issued. If you reply IGNORE and there are log archive files to go with this database archive file, these log archive files are applied. If you reply CANCEL, the application server terminates to allow you to mount a different database archive file. Any other reply causes message ARI0052E to be displayed and message ARI0251D to be redisplayed.

Note: At this time, the database archive mounted has replaced your database minidisks.

Operator Response: Reply IGNORE if you wish to continue with the restore of the database archive volume currently mounted. Reply CANCEL if you wish to terminate the application server and mount a different database archive to be used in the restore process.

ARI0252I Medium: {Tape cuu **| Disk** fn ft fm**}.**

Explanation: When displayed, ARI0252I always follows ARI0239I. A log or database archive is about to be created, or the database manager is ready to read the next log archive file during the restore process.

During a log archive:

ARI0252I tells the operator where the database manager plans to write the archive. The medium (tape or disk) the system chooses is based on the FILEDEF for ARILARC at startup, or the medium used for the most recent log archive (if any) since then.

If the medium is tape, the address shown is the most recent tape address specified for ARILARC at startup, log restore, or log archive; if unspecified so far, it is "???".

If the medium is disk, a default file identifier is presented (fn ft fm) to assist the operator. The file name (fn) is the database name. The file type (ft) is

```
mmddyynn, where
mmddyy  is the date the archive is taken (in the
        system's date format),
    nn  designates the file as a log (archive),
        and is the archive sequence (01-99) on
        that date.
```

If the database manager is unable to establish a default file type, message ARI0271A will have been displayed and the file type will be "????????".

The default file mode (fm) is the file mode previously specified for log archive disk output, either in the FILEDEF at startup, or in response to messages. If the file mode has not been specified yet, it is displayed as "??".

During a log restore:

ARI0252I tells the operator where the database manager plans to read the log archive from. If the log is on disk, the file name and file type are taken from the history area of the log, and the file mode shown is "*".

If the log is on tape, the address shown is the most recent tape address specified for ARILARC at startup, log restore, or log archive; if unspecified so far, it is "???".

System Action: The archive or restore process continues.

Operator Response: Respond to the messages following ARI0252I.

ARI0253D The database archive mounted is not known to the database manager. Enter either: IGNORE to continue, or CANCEL to restart the restore process.

Explanation: The database archive that was just restored does not have a record in the history area. This could be caused by:

1. The record of this database archive was written over. If it was your most current database archive, you should have gotten message ARI0258E as a warning of this. If it was a back level database archive, you would get no warning.
2. You did a COLDLOG (STARTUP=L) sometime after taking this database archive which destroyed the history area for this database.

   Note: When running the SQLLOG EXEC, if you answer Y to message ARI0688D, you will also destroy the history area for the database.

This message gives you the chance to terminate the application server if you wish to mount a different database archive file.

System Action: If you reply IGNORE, message ARI0264D is issued. If you reply CANCEL, the application server terminates.

Operator Response: Reply IGNORE if you wish to continue with the restore of the database archive volume currently mounted. Reply CANCEL if you wish to terminate the application server and mount a different database archive to be used in the restore process.

ARI0254I The database manager is initiating a log archive. When the log archive is complete, the database manager will process the database archive request.

Explanation: A database archive has been requested and LOGMODE=L but the current log has not been archived. Therefore, the database manager has initiated a log archive to ensure that log archives will remain useful in recovering to this point.

Note: This log archive will become the last archive in the restore set for the database archive taken prior to this database archive request.

System Action: The database manager has automatically initiated an on-line log archive operation. It will then process the database archive request.

ARI0255I The database manager is initiating a log archive. When it is completed, the restore process will continue.

Explanation: A database restore has been requested and LOGMODE=L but the current log has not been archived. Therefore, the database manager has initiated a log archive.

Note: This log archive will become the last log archive in the current restore set.

System Action: The database manager will automatically initiate an on-line log archive operation. The database manager will then resume restore processing.

ARI0256E You must specify ROLLBACK by DBSPACE ***dbspace-number*** **in the extend file.**

Explanation: A ROLLBACK COMMITTED WORK command was specified for a logical unit of work containing one or more DROP DBSPACE or DROP TABLE commands and one or more of the following occurred:

- The DBSPACE control keyword was submitted but did not specify all of the DBSPACEs which are affected by the DROP commands.
- The DBSPACE control keyword was not submitted with the command.
- The ROLLBACK COMMITTED WORK command contained the DBSPACE control keyword with other ANDed control keywords (USERID, TIME, or DATE).

This message may be issued more than once for the same DBSPACE.

Note: DBSPACE 1 (catalog DBSPACE) need not be specified on the command although the DROP commands update the catalog.

dbspace-number is a DBSPACE number of the DBSPACE affected by a DROP DBSPACE or DROP TABLE command.

System Action: Processing of the log is completed and summary statistics are displayed. Message ARI0213D is not displayed and processing is ended.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Use the summary information to determine if the specified actions are to be done. If so, include each DBSPACE specified in this message (ARI0256E) in a ROLLBACK COMMITTED WORK command. Then, restart extended database initialization.

ARI0257I **New LUWs and the log archive process cannot continue until** n **non-R/O active LUWs are completed.**

Explanation: A log archive has been requested either through a command or due to the ARCHPCT value being exceeded. The log archive process requires that there be no logical units of work active at the time that the log archive begins. This is so that the logical units of work do not span log archives. The database manager displays this message to inform the operator of how many logical units of work are active at the time of the log archive request.

System Action: When a log archive is requested, the database manager prevents new logical units of work from starting. Users that try to start a new logical unit of work will be in a lock wait until the log archive begins. Logical units of work that are active when the log archive is requested are allowed to complete. In most cases, there is only a slight delay before the log archive begins. If, however, there are long running logical units of work, the delay will be longer. A logical unit of work may be long enough so that the SLOGCUSH value is reached. In this case, the database manager will start log overflow processing and rollback the longest-running logical units of work to free log space.

Operator Response: If you find that a long running logical unit of work is holding up the log archive, you can use the SHOW operator commands to determine which user is delaying the start of the log archive. You may wish to use the FORCE command to end those logical units of work.

ARI0258E **The log indicates that you are in danger of losing the record of your latest database archive. To prevent this, you must do a database archive.**

Explanation: The database manager keeps track of information such as database archives, log archives, logmode switches, coldlogs, and restores. This is so that the database manager can be sure that the database and log archive files used for restore are valid and in the correct order. However, there is a limit on the number of records the database manager can store at any one time. When the limit is reached, the oldest records are lost. This is not a problem unless the record of the last database archive taken is lost. This message is issued when the latest database archive is within 10 records of being written over and every time a record is stored after that until the record of the latest database archive is actually written over. At this point, message ARI0262E is issued.

System Action: Processing continues.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Be sure to take a database archive before receiving this message a total of 10 times. After a record of your latest database archive is lost, you will not be able to restore your database to its current state until you have taken a database archive.

ARI0259I **No logical unit of work started at or after the time specified in the ROLLBACK command.**

Explanation: This information message may be displayed when TIME is used as a control keyword in the EXTEND input file. It will be followed by message ARI0210I or message ARI0212I.

System Action: Processing continues.

System Programmer Response: Check the control keywords in the EXTEND input file and adjust if necessary. See message explanations for the accompanying messages.

ARI0260I **The restore set for this database archive is as follows:**

Explanation: This message is displayed every time a database restore completes. It is followed by messages ARI0238I and ARI0261I which identify the set of database archives and log archives to be used with the database archive just restored.

System Action: Message ARI0238I follows.

ARI0261I **[-->] log archive - {TAPE|DISK** *fn ft***}** *mm-dd-yy hh:mm:ss*

Explanation: The message identifies the next log archive file in the restore set.

Message ARI0239I displayed this information when the log archive file was created. The information displayed should have been recorded externally on the tape. Message ARI0239I will display this information again when the restoration process is ready to read this log archive file.

If the arrow (-->) is displayed before log archive, ARI0261I is followed by ARI0267D, which explains the meaning of the arrow.

System Action: ARI0261I is issued for each log archive file to be applied to this database.

ARI0262E The record of your latest database archive has just been lost. You must archive the database to retore your database to its current state.

Explanation: The database manager keeps track of information such as database archives, log archives, logmode switches, coldlogs, and restores. This is so that the database manager can be sure that the database and log archive files used for restore are valid and in the correct order. However, there is a limit on the number of records the database manager can store at any one time. When the limit is reached, the oldest records are lost. This is not a problem unless the record of the last database archive taken is lost. This message is issued when the latest database archive has been written over.

System Action: Processing continues.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Take a database archive as soon as possible to ensure that if you need to restore the database that you will be able to.

ARI0263D To direct the log archive to tape, enter TAPE followed by the tape address (CUU) to be used. To direct the log archive to disk, enter DISK followed by the disk file's file name, file type and file mode. If you choose DISK, the default file is: *filename filetype filemode***.**

Explanation: A log archive is being created and the operator is prompted to change, or has indicated a change is necessary, to the destination of the archive specified in ARI0252I. ARI0263D requests that this new destination be specified, either as a tape address or a disk file identifier. In case a disk archive is wanted, a default file identifier is presented (fn ft fm) to assist the operator. The default file identifier is described under ARI0252I.

System Action: If the operator response is valid, the database manager displays it in ARI0252I (preceded by ARI0239I), allowing the operator to confirm that this is the intended destination. If the response was invalid, an error message is displayed and ARI0263D is redisplayed.

Operator Response: To have the log archive written to tape, enter TAPE, followed by the tape address (cuu) to which the archive will be written. This address is normally 183.

To have the log archive written to disk, enter DISK, followed by the file identifier (file name, file type, and file mode) of the disk file to contain it. You can specify an equal sign (=) as a component of the file identifier to indicate that you want that component's value taken from ARI0263D. The file mode letter must specify a currently accessed CMS minidisk or Shared File System directory.

The file identifier cannot have been used already for a log archive. Use the default naming convention.

If this is a redisplayed message and the system error message DEVICE *device name* DOES NOT EXIST is displayed before this redisplayed message, issue a CP LINK command from the database operator console to link back to the detached disk to store the log archives.

System Programmer Response: If you have log archives written to disk files, ensure that a convention to name these files uniquely is established and followed.

ARI0264D If a COLDLOG was not performed immediately preceding this restore process, enter CANCEL, perform COLDLOG, and start the database manager with STARTUP=W. Otherwise, enter IGNORE.

Explanation:

- You replied IGNORE to message ARI0251D or message ARI0253D, or
- You replied END RESTORE to message ARI0250D before applying the first log archive file in your restore set, or
- An end of restore set was found before the end of the current log.

Unless you executed a COLDLOG immediately preceding the STARTUP=R, your current log is out of synchronization with the database just restored. This message gives you a chance to COLDLOG before the log is applied to the database.

System Action: If you reply IGNORE, the database manager will assume that you have performed a COLDLOG prior to restore and log recovery, and the restore process will continue.

If you reply CANCEL, the application server will terminate.

Operator Response: If you have performed a COLDLOG immediately preceding the restore, then reply IGNORE. If you have not, then reply CANCEL, execute the COLDLOG function and then start the application server with STARTUP=W. The COLDLOG function is performed by the SQLLOG EXEC. See the *DB2 Server for VM System Administration* manual for more information on the SQLLOG EXEC.

ARI0265I The block size for *ddname* **is invalid. A block size of 4K bytes will be used.**

Explanation: The block size specified on the FILEDEF for *ddname* is not a multiple of 4K or it is greater than 28K. *ddname* is ARILARC for the LOG ARCHIVE, or ARIARCH for the DATABASE ARCHIVE.

System Action: Database initialization continues using the default block size of 4K.

Operator Response: If a block size of anything other than 4K is desired, the operator must end database manager operation by issuing the SQLEND command, change the block size in question to be a multiple of 4K, and restart the database manager after initialization is complete.

ARI0266A The specified file is an existing log archive.

Explanation: The disk file identifier entered in response to ARI0263D has a file name and file type which have already been used for a log archive file. This fact is based on a search of the history area of the log file.

System Action: ARI0263D is displayed to allow an unused log archive file identifier to be specified.

Operator Response: Enter a valid, unused log archive file identifier in response to ARI0263D.

ARI0267D The log archive disk files indicated by an arrow (-->) cannot be found on any currently accessed disk. Enter either: CONTINUE to continue this restore process, or STOP SYSTEM to interrupt this process, or END RESTORE to end this process.

Explanation: When displayed, this message follows ARI0261I. It indicates that all currently accessed disks have been searched for each disk log archive in the restore set, but at least one of the archives could not be found. The missing archives are indicated by an arrow in ARI0261I.

You have three options with this message:

1. Reply CONTINUE: The restore process continues.
2. Reply STOP SYSTEM: The restore process is suspended and the application server stops. The restore process will start where it left off if you warm start the application server with STARTUP=W.
3. Reply END RESTORE: The restore process is ended. Also, you will never be able to restore the remaining log archives to the database archive you have just restored. This is because the remaining log archives are no longer a part of the restore set for this database archive in the history area. However, if you go to a back level database archive, these log archives may be applied as usual.

   The only case in which you should reply END RESTORE is if you do not want to restore any of the log archives in the restore set. END RESTORE should only be used in extreme cases as it could result in a permanent loss of data.

System Action: One of the following actions will occur:

- Reply was CONTINUE: The restore process continues. The application server begins processing the first log archive.
- Reply was STOP SYSTEM: The restore process is suspended. The application server stops.
- Reply was END RESTORE: You will receive message ARI0264D.

Operator Response: You will normally want to reply STOP SYSTEM. This lets you interrupt the restore process, determine why the archives could not be found, correct the problem, and re-start the restore. You may also want to use this reply if you are using an EXTEND input file and need to change the EXTEND commands. There is no danger in using STOP SYSTEM.

There are two cases in which you may want to reply CONTINUE:

- You want to restore some of the log archives in the restore set, then interrupt the restore and fix the problem of the missing log archive before continuing the restore.
- You want to restore some of the log archives in the restore set, and plan to END RESTORE when the missing log archive is to be processed.

Do not reply END RESTORE without consulting your system programmer.

System Programmer Response: Be sure that you understand the consequences of replying END RESTORE.

ARI0268D The above log archive disk file cannot be found on any currently accessed disk. Enter either: STOP SYSTEM to interrupt this restore process, or END RESTORE to end this process.

Explanation: When displayed, this message follows ARI0239I and ARI0252I. It indicates that the log archive file described in these two messages is not found on any of the currently accessed disks.

You have two options with this message:

1. Reply STOP SYSTEM: The restore process is suspended and the application server terminates. The restore process will start where it left off if you start the application server with STARTUP=W, LOGMODE=L.
2. Reply END RESTORE: The restore process is ended. Also, you will never be able to restore the remaining log archives to the database archive you have just restored. This is because the remaining log archives are no longer a part of the restore set for this database archive in the history area. However, if you go to a back level database archive, these log archives may be applied as usual.

System Action: One of the following actions will occur:

- Reply was STOP SYSTEM: The restore process is suspended. The application server terminates.
- Reply was END RESTORE: If you have not applied the first log archive file to your database archive file, you will receive message ARI0264D. If you have applied at least one log archive file to your database archive file, the database manager will complete initialization and begin application processing.

Operator Response: You will normally want to reply STOP SYSTEM. This lets you interrupt the restore process, determine why the archive could not be found, correct the problem, and re-start the restore. You may also want to use this reply if you are using an EXTEND input file and need to change the EXTEND commands. There is no danger in using STOP SYSTEM.

The only case in which you should reply END RESTORE is if you do not want to restore this log archive or any later one in the restore set. END RESTORE should only be used in extreme cases as it could result in a permanent loss of data. Do not reply END RESTORE without consulting your system programmer.

System Programmer Response: Be sure that you understand the consequences of replying END RESTORE.

ARI0269D The log archive will overlay an existing disk file. Enter either OVERLAY to proceed, or CHANGE to change the destination of the log archive.

Explanation: The system has been instructed to write a log archive to the disk file previously displayed or entered by the operator. There is an existing file with the same identifier, however, and the log archive, if continued, would overlay the existing file.

System Action: The database manager waits for the operator's reply. If the reply is OVERLAY, the archive will proceed to overlay the existing file. If the reply is CHANGE, message ARI0263D will be displayed to allow the destination of the archive to be changed.

Operator Response: If you are certain the existing file is nonessential and should be destroyed, reply OVERLAY. Otherwise, reply CHANGE and change the archive destination in response to later messages.

ARI0270A The database manager does not have write access to the output disk.

Explanation: The database manager has been instructed to write a log archive to the disk file previously displayed, or entered by the operator. However, the database machine does not have the disk accessed in read/write mode. (It may not have the disk accessed at all.

If the output disk is a Shared File System directory (VM/ESA Version 2 Release 1 or later), a some VM commands (such as XEDIT) are able to write to it even if accessed in read-only mode. This capability does not extend to the database manager.

System Action: ARI0263D is re-displayed, allowing the operator to specify a different disk file mode, or to choose a tape archive instead.

Operator Response: Respond to ARI0263D, requesting a different disk or tape.

System Programmer Response: Ensure that, before the application server is started each time, the database machine has read/write access to all the disks to which it may need to write a log archive.

ARI0271A The database manager could not establish a default file type for the log archive disk file.

Explanation: For log archiving to disk, the database manager will name the output disk files and allow the operator to confirm them. The naming convention followed by the database manager allows 99 log archives per day. ARI0271A indicates that the log archive just started will exceed this limit.

System Action: Instead of displaying the default log archive file type in message ARI0263D, the file type displayed is "????????".

Operator Response: Respond to ARI0263D by choosing a valid file type, preferably similar to the default. The naming convention for the default is discussed under ARI0252I.

ARI0272E FILEDEF for *file* must have one of these device types: *device1, device2, device3, device4,* or *device5*.

Explanation: You have executed a FILEDEF command for the specified *file* before starting the application server, but the device type used in the FILEDEF is not valid for that file. *device1* to *device5* are the valid device types for *file*.

System Action: The application server startup is terminated.

User Response: Execute a FILEDEF for *file*, using either:
- One of the device types shown in the message, or
- The CLEAR option (to have the default FILEDEF used).

Then re-execute the command that caused this message to be issued.

ARI0274I Implicit COMMIT failed for *userid* to avoid deadlock or because of system request. A ROLLBACK of the LUW will occur.

Explanation: User *userid* did not issue a commit work command before ending, and the database manager started implicit commit processing for that user. During this processing, it was found that the work performed by the user had to be rolled back due to a system request (such as log full, out of lock request blocks, etc.) or to avoid a deadlock situation.

System Action: All database changes for this logical unit of work have been rolled back.

Operator Response: Inform user *user-id* that the work has not been committed, but rolled back. User *user-id* should inform the application developer that this condition occurred.

User Response: Recover the work that had been rolled back. Also, it is strongly recommended that a commit work command be issued by applications prior to application termination. If this is done, the application could indicate to the user when a COMMIT WORK is actually completed.

ARI0275D Archive file open error. Enter either: RETRY to try again, or CANCEL to cancel. Note: entering CANCEL will cancel the { database manager | LARCHIVE}.

Explanation: When trying to open the database manager archive tape or disk output file, an error was detected during CMS OS QSAM OPEN processing as indicated by message ARI0296A and by a CMS DMS error message. A likely cause is that the output file is for tape, but there is no tape mounted.

If the message text says 'CANCEL the database manager', the database manager will end if the operator replies CANCEL. This will occur for all database archives and for all log archives except a log archive initiated by the database LARCHIVE operator command.

If the message text says CANCEL LARCHIVE, only the log archive initiated by the LARCHIVE operator command will be canceled. The database manager will continue normal operations.

System Action: The system waits for your reply.

Operator Response: Correct the reason for the open error as indicated in previous messages. Reply RETRY so that the database archive file open can be tried again. For a database archive, message ARI0299A is displayed before the retry. Reply CANCEL to:

- Stop the database manager if 'CANCEL the database manager' appears in the message text.
- Cancel the log archive request if CANCEL LARCHIVE appears in the message text. You can reenter the LARCHIVE command after correcting the error condition as indicated by message ARI0296A.

ARI0276E Log is full.

Explanation: The database log is full. Overflow procedures, designed to prevent this from happening, have failed.

System Action: The application server is ended with a limit error. Logical units of work in progress at the time of the termination will be undone when the application server is warm started.

Operator Response: Notify your system programmer concerning the size of the log minidisks and the occurrence of this message. Possibly the size of the log minidisks should be increased. Also, it is possible that one logical unit of work has been started and has not been committed or rolled back; that is, it inadvertently remains active. This active logical unit of work prevents reuse of database log space.

You must always warm start the application server (initialization parameter STARTUP=W) to recover the logical units of work in progress when the system terminated. You can then, after a "clean" application server shutdown, redefine the minidisks and start the application server with the new log minidisk definitions using the SQLLOG EXEC to run with the larger logs.

System Programmer Response: You can minimize the space required in the log by running with LOGMODE=N or LOGMODE=Y (rather than LOGMODE=A). This is especially appropriate when loading or reloading the database (including DBS utility DATALOAD and RELOAD operations) and when running other applications that make mass updates to the database. If you are already doing this and are encountering this problem, consider redefining the logs with a larger size.

Switching log modes and redefining the logs are described in the *DB2 Server for VM System Administration* manual.

ARI0277E Archive input file *cuu* does not contain a DB2 Server for VM identifier.

Explanation: The volume mounted on tape unit *cuu* in response to message ARI0295A is not a DB2 Server for VM database archive tape.

Possible causes include:

- The mounted (and opened) volume for ddname ARIARCH does not contain a DB2 Server for VM database archive file.
- The mounted (and opened) volume for ddname ARIARCH is a DB2 Server for VM database archive volume, but is not the first volume of a multi-volume

archive file. This can occur only if you do not request CMS OS QSAM open processing to check the file's volume sequence number.

Note: No database updates have yet occurred.

System Action: The archive file is closed. Message ARI0295A is displayed again to allow the operator to mount the correct volume and continue the restore process. If the close fails, message ARI0298E is displayed and the application server terminates.

Operator Response: Determine why the correct volume was not mounted, mount the correct volume, and reply to message ARI0295A with the tape unit cuu.

If you are unable to determine the cause of the problem, contact your system programmer.

System Programmer Response: Ensure that archive file volumes are properly identified so that you know which volumes belong to a database archive and the order in which they must be mounted. Ensure that your DB2 Server for VM operator and VM system operator follow procedures that allow proper identification at archive creation time and proper volume mounting at archive restore time. You should use extensive standard volume and file label checking and creation procedures to assist in this identification and verification process.

ARI0278E Unexpected EOF on archive input file [*cuu*].

Explanation: During a restore from archive, one of the following occurred:

- The database manager encountered end-of-file on the database archive file before all data required to restore the database directory and DBEXTENTs was read.
- The database manager encountered end-of-file on a log archive before all data required to restore the archived DB2 Server for VM log was read.

cuu identifies the tape unit containing the database or log archive file. No *cuu* is specified for a log archive file being restored from disk. Possible causes include:

- A DB2 Server for VM or VM system error occurred that caused database or log archive file records to be lost.

System Action: The database or log archive file is closed, and the application server stops. Note that the database has been only partially restored and is not usable.

Operator Response: If the error occurred while restoring a database archive file and you can determine the cause of the error, restart the application server with the STARTUP=R initialization parameter to redo the database restore process.

If the error occurred while restoring a log archive file and you can determine the cause of the error, warm start the application server (STARTUP=W). It will resume the restore process with the current log archive file (the one which encountered this error).

Otherwise, contact your system programmer.

System Programmer Response: If the error occurred on a database archive file, you should retry the database (STARTUP=R) restore process. If the error persists, you will not be able to use this database archive file.

If the error occurred on a log archive file, you should try to resume the restore process with the current log archive file by warm starting the application server (STARTUP=W). If the error persists, you will not be able to use this log archive file.

Refer to the *DB2 Server for VM System Administration* manual for recovery procedures when a database or log archive file cannot be processed and you need to restart a database restore operation.

ARI0279E Archive input file is not current. No COLDLOG is issued.

Explanation: During a database restore from archive, the database manager determined from its log that:

- The archive copy was not the most recent archive copy of the database, and
- The COLDLOG function was not performed (COLDLOG is performed by executing the SQLLOG EXEC described in the *DB2 Server for VM System Administration* manual).

You cannot restore a database from an archive copy that is not the most recent archive copy of the database unless you immediately precede this task with the COLDLOG function. Refer to the *DB2 Server for VM System Administration* manual for information on restoring a database.

System Action: The application server terminates. The database has been updated to reflect the contents of the archive file, but it may not be in a "current and consistent state" because the updates in the DB2 Server for VM log have not been applied to the database.

Operator Response: Refer this message to your system programmer.

System Programmer Response: You have either:

- Accidentally attempted to restore a database from an archive copy that is not the most recent archive copy, or
- You have failed to execute the COLDLOG function before attempting to restore the database using a back-level archive copy.

You must either:

- Rerun the database restore process using the most recent archive copy of the database, or

- Run the COLDLOG function (SQLLOG EXEC) before running the database restore process using the back-level archive copy. If you intended to restore the database from a back level archive copy and forgot to perform the COLDLOG function, you can now perform the COLDLOG function and then restart the application server for normal processing (if you are sure that you have applied the correct back level archive copy).

ARI0280A Previously LOGMODE was A(ARCHIVE) and now LOGMODE=*&1*. To restart with LOGMODE=*&1*. perform shutdown with ARCHIVE unless an archive has just been taken, perform COLDLOG, and restart with LOGMODE=*&1*.

Explanation: This message is issued to protect against making a previous DB2 Server for VM (or any type) of user archive useless. To run with LOGMODE=N or LOGMODE=Y, after running with LOGMODE=A, follow the procedure given in the message.

Note: This message can occur if:

1. You last ran the database manager with LOGMODE=Y and you issued an SQLEND ARCHIVE or SQLEND UARCHIVE command. These commands internally change the current LOGMODE to A if the current LOGMODE is Y.

   Or

   You last ran the database manager with LOGMODE=Y, issued an SQLEND, then ran SQLADBEX, specifying ARCHIVE or UARCHIVE, which internally changes the current LOGMODE to A if the current LOGMODE is Y.
2. You then restarted the application server with LOGMODE set to Y or N. Note that LOGMODE=Y is the default.

In both of the above circumstances, performing a shutdown with archive is not required to continue with LOGMODE=Y, since an archive has just been performed.

System Action: The application server is ended.

Operator Response: If you have started the application server with an incorrect LOGMODE value, correct the LOGMODE value and restart the application server. Otherwise, restart the database manager with LOGMODE=N or LOGMODE=Y after performing the procedure outlined in the message. The archive step may be skipped if this message is caused by either of the two scenarios described in the above note. COLDLOG is performed by starting the database manager with the SQLLOG EXEC. See the *DB2 Server for VM System Administration* manual for more information on the SQLLOG EXEC.

ARI0281I LUW REDO is completed.

Explanation: The logical units of work in the current log which required redoing at startup time have been redone by the recovery procedure.

System Action: Initialization continues.

ARI0282I LUW UNDO is completed.

Explanation: The logical units of work in the current log which required undoing at startup time have been undone by the recovery procedure.

System Action: Initialization continues.

ARI0283I Log analysis is complete.

Explanation: The current log has been scanned successfully by the recovery procedure to determine those logical units of work which require redoing or undoing at startup time.

System Action: Initialization continues.

ARI0284I SHOW LOG command is rejected because LOGMODE = N.

Explanation: A SHOW LOG command was entered by the DB2 Server for VM virtual machine console operator or the ISQL terminal user while running with LOGMODE=N.

System Action: The command is ignored.

ARI0285I Checkpoint interval is reduced to *n*.

Explanation: The checkpoint interval is the number of log pages which will be written between automatic checkpoints. This interval is specified by the database initialization parameter CHKINTVL. At database initialization, an analysis of the checkpoint interval and the size of the log is made to determine if at least one checkpoint will be taken before end of log or log wrap. If CHKINTVL will not cause one checkpoint, it is reduced by the required amount to cause one checkpoint to be taken.

System Action: Processing continues. The value **n** (in pages) is used to cause a checkpoint to be taken.

Operator Response: Unless this message is expected, give it to your system programmer.

System Programmer Response: You may wish to change the CHKINTVL database initialization parameter to avoid this message.

ARI0286I RESTORE cancelled by operator request.

Explanation: The operator replied CANCEL to message ARI0295A while restoring the database. Message ARI0295A is displayed (to ensure that the tape

is mounted) before opening the database archive file or before opening a log archive file (restoring with LOGMODE=L) when the tape cuu to be used is unknown.

Unless STARTUP=R was specified and the restore process was canceled before restoring the database archive file, the database has been partially restored.

System Action: The application server ends.

Operator Response: Refer to message ARI0295A (where you replied CANCEL).

ARI0287I ARCHIVE command rejected because a {database | log} ARCHIVE is being performed.

Explanation: An ARCHIVE command was entered by the database operator while either a log archive or a database archive was being taken.

System Action: The log archive or the database archive that is being taken will be completed, and the ARCHIVE command is ignored.

ARI0288I ARCHIVE command rejected because archiving is not enabled.

Explanation: An ARCHIVE command was entered by the database operator, but the database manager is not running with LOGMODE=A or LOGMODE=L.

Note: It is valid to enter SQLEND ARCHIVE if archiving is not enabled (LOGMODE=Y).

Note: SHUTDOWN is a valid synonym for SQLEND.

The ARCHIVE command requires a LOGMODE of A or L.

System Action: The database manager ignores the ARCHIVE command.

Operator Response: If you were instructed to enter the ARCHIVE command, notify your system programmer. Otherwise, no action is required.

System Programmer Response: Tell the operator not to enter the ARCHIVE command if LOGMODE is not A or L and do not enter the SQLEND ARCHIVE command if the LOGMODE is not Y, A, or L.

ARI0289I Restoring directory disk.

Explanation: This message occurs when the application server has been started with STARTUP=R, the archive input volume has been readied, and the directory file is being restored.

System Action: The restore process continues.

ARI0290I Restoring data disk.

Explanation: The application server has been started with STARTUP=R , the archive input volume has been readied, the directory disk has been restored, and the restoring of the data disks (DBEXTENTs) is in process.

System Action: The database manager proceeds with the restore process.

ARI0291I System restore from directory and data disk(s) of database archive is completed.

Explanation: The application server has been started with STARTUP=R

System Action: The database manager continues with initialization. The archive volumes can be removed from the system and stored until needed.

ARI0292I Archive is completed.

Explanation: A database or log archive has completed. If the archive was to tape, the archive tape unit is now available for other purposes.

System Action: Processing continues.

Operator Response: If the archive was to tape, the archive volumes can be removed from the system and stored until required for the restore process (or until they are no longer required).

ARI0293I Archive is starting.

Explanation: A database or log archive is beginning.

System Action: The database manager continues the archive process.

ARI0294A Archive file [*cuu*] {read|write} failed. Reason Code = *nn*. Return Code = *nn*.

Explanation: While attempting to read or write a DB2 Server for VM database or log archive file, an access error occurred. *cuu* is the address of the tape device on which the error occurred; it is not displayed for an error on a disk device.

If the message text specifies WRITE, the database or log archive file has not been completed; it cannot be used. After the I/O error condition is corrected:

- If this error occurred while creating a log archive during the database restore process, restart the restore process with the same STARTUP parameter value. If you user-restored the database, you do not have to repeat that process but you must restart with STARTUP=U.
- If this error occurred while creating a log archive file with the LARCHIVE operator command, reenter the LARCHIVE operator command.
- Otherwise, restart the application server and if applicable, reinitiate the archive process.

If the message text specifies READ, a database or log archive file was being used to restore the database. (The application server had been started with the initialization parameter STARTUP=R or U. The database is only partially restored; it can only be recovered by a successful archive restoration (described below) or by rebuilding the database using the database generation process.

The Reason Code value is the Primary Error Code. The Return Code value is the Secondary Error Code. For further explanation, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Action: If this error occurred due to a log archive initiated using the LARCHIVE operator command, the log archive process ends, but the database manager continues normal operation. For all other cases, the application server ends.

Operator Response: For corrective action, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

If an archive file was being written, restart the application server or the LARCHIVE operator command according to the situation described under **Explanation**. This recovery action is described in the *DB2 Server for VM System Administration* manual.

If an error occurred when reading a database archive file and if you are able to use the archive file which previously failed, start the application server with the initialization parameter STARTUP=R to restore the database and continue normal processing.

If an error occurred while reading a log archive file, warm start the application server. It will resume the restore process with the log archive that failed.

If a database or log archive file cannot be read (for example, uncorrectable media error), refer to the *DB2 Server for VM System Administration* manual.

System Programmer Response: See Operator Response.

ARI0295A Ready archive input volume. Enter CANCEL or the cuu.

Explanation: Restoring a database from archive files has been begun by starting the application server with STARTUP=R or U.

The database manager is prompting the operator to ensure that the correct database or log archive tape is mounted and readied on tape unit **cuu**. The operator is being requested to respond by entering the **cuu** on which the tape has been mounted or CANCEL to terminate.

If the archive file could not be opened successfully, message ARI0296A was displayed to indicate the error condition. This message was redisplayed to give the operator the opportunity to correct the error and restart the archive restore process.

If the **cuu** value entered in response to this message was not valid, message ARI0297A was displayed to indicate the error condition. This message was redisplayed to give the operator the opportunity to enter a valid **cuu**.

If the database archive file did not contain a DB2 Server for VM database archive, message ARI0277E was displayed and the file was closed. This message was redisplayed to correct the error and restart the database restore process.

Note: If the request was for a database archive file, no database updates have yet occurred. Otherwise, the database has been partially restored.

System Action: If the operator responds with CANCEL, the restore is canceled and the application server is ended.

If the database operator replies with *cuu*, one of the following occurs:

- If the request is for a database archive file, the tape file on the virtual tape unit specified on the CMS FILEDEF command for ddname ARIARCH is opened for input, and the restore process proceeds.
- If the request is for a log archive file, the tape file on the virtual tape unit specified is opened for input, and the restore process proceeds.

For a database archive restore, the **cuu** entered is not actually used in opening the file. The operator should enter the correct **cuu**, however, because it may appear in later messages (ARI0296A, ARI0294A, and ARI0298A).

Operator Response: If this message was redisplayed after message ARI0277E, ARI0296A, or ARI0297A, take the corrective action suggested by that error message.

If the request is for a log archive file, you can use any available virtual tape unit.

Request (using the CP MSG OP command) that the CP operator select an available tape unit, mount and ready the first (or only) volume of the archive file, and attach the tape unit to your virtual machine with the required virtual tape unit address. (TAPn is unit 180-7, if n= 0 to 7 and TAPn is unit 288-F, if n= 8 to F.) When the CP operator notifies you that these actions have been taken, enter the **cuu** of the virtual tape unit.

If no virtual or real tape unit is available, or if the archive file is not available, or if the error condition from message ARI0277E or ARI0296A cannot be corrected, enter CANCEL to end the restore process and the application server.

If you reply CANCEL for a database archive file, you must restart with STARTUP= R. for database recovery.

If you reply CANCEL for a log archive file, you should warm start the application server (STARTUP=W) and continue the restore process with the current log archive file. The current log archive file is the one you were requested to mount.

ARI0296A Archive file [*cuu*] ASSIGN or OPEN failed. Reason Code = *nn*. Return Code = *nn*.

Explanation: A DB2 Server for VM database archive or restore from archive is in progress. An error occurred while attempting to open the archive output file or restore input file. *cuu* is the address of the tape device on which the error occurred; it will not be displayed for an error on a disk device.

The Reason Code value is the primary error code. The Return Code value is the secondary error code. For further explanation, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

Note: If the request was for a database archive file, no database updates have yet occurred. Otherwise, the database has been partially restored.

System Action:

For restore, message ARI0295A is redisplayed. For archive, if Reason Code = 8 and Return Code = 0, then message ARI0275D is displayed; otherwise:

- For a log archive initiated by the LARCHIVE operator command, the log archive ends and the database manager continues normal operation.
- For all other archive output files, the application server stops.

Operator Response: For corrective action, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

If the application server stops while creating a log archive during the restore process, restart the process after taking corrective action for the error.

If the application server stops while creating a normal database or log archive, warm start the application server (STARTUP=W) after taking corrective action for the error. Then, take the action you took previously to initiate the database or log archive. If the archive was initiated by the database manager, it will be initiated again.

If an LARCHIVE operator command request failed and the database manager continued operating, you can retry the LARCHIVE command after correcting the cause of this error message.

ARI0297A Response to archive prompt is not valid.

Explanation: An archive prompt message has been issued, but the operator's reply was not valid.

If ARI0263D has been issued, the response has a syntax error. Possible causes:

- TAPE or DISK was misspelled.
- If TAPE was specified, it was not followed by a 1-3 character cuu that is a valid address (each character must be 0-9 or A-F).
- If DISK was specified:
  - It was not followed by a valid CMS file identifier exactly three character strings of maximum lengths 8, 8, and 2, respectively, or the third character string (file mode) did not begin with an alphabetic character. If the second character string exists, it is not in the range `0` to 6.
  - An equal sign ("=") was used as the value for the file type or file mode, but the database manager had not established a default value for it.

System Action: The database manager issues the prompt message again and asks for valid input.

Operator Response: Ensure that the input being entered is valid, and enter it in response to the prompt message.

ARI0298E Archive file [*cuu*] CLOSE failed. Reason Code = *nn*. Return Code = *nn*.

Explanation: The database manager was creating or reading a database or log archive file. An error occurred when the database manager attempted to close the database or log archive file.

cuu is the address of the tape device on which the error occurred; the address will not be displayed for an error on a disk device.

System Action: If a log archive file was being created with the LARCHIVE operator command, the log archive ends and the database manager continues normal operation. If a database or log archive file was being created for any other reason, the application server ends. The database or log archive file cannot be used. The system log indicates that the archive was not successful.

If the database was being restored from a database archive and message ARI0277E precedes this message, the application server stops. The database has not been updated by the restore process.

If the entire database was being restored from a database archive and message ARI0277E was not displayed, normal processing continues.

If a log archive was being read during the restore process, normal processing continues.

Operator Response: For corrective action, look up the displayed error codes under either "Chapter 23.

Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

Then, if a log archive was being created during the restore process, restart the process.

Then, if this error occurred due to a log archive initiated by the LARCHIVE operator command, reenter the LARCHIVE operator command to re-initiate the log archive process.

Then, if a database or log archive was being created, warm start the application server, If desired, reschedule the archive creation by using the ARCHIVE, LARCHIVE, SQLEND ARCHIVE, SQLEND LARCHIVE, or SQLEND UARCHIVE operator command. This recovery action is described in the *DB2 Server for VM System Administration* manual. If the database manager implicitly initiated the database or log archive due to the ARCHPCT initialization parameter, it will do so again when you warm start the application server.

Note: SHUTDOWN is a valid synonym for SQLEND.

If the database was being restored from a database archive and message ARI0277E precedes this message, take the corrective action as implied by these two messages. Restart the database archive process.

If the database was being restored from a database archive and message ARI0277E does not precede this message, you can proceed with whatever activity you had planned. The database has been successfully restored from the archive copy.

No further action is required if the error occurred while closing a database or log archive input file during the restore process.

System Programmer Response: See Operator Response.

ARI0299A Ready archive output volume. Enter the cuu.

Explanation: A database or log archive is being created for one of the following reasons:

- You entered the ARCHIVE operator command.
- You entered the SQLEND ARCHIVE operator command.
- The database manager has implicitly initiated the creation of a database archive due to the ARCHPCT initialization parameter.

If the cuu entered in response to this message was not valid, message ARI0297A was displayed to indicate the error condition.

System Action: When the operator responds with cuu, tape unit cuu is dynamically assigned to the archive output file; it is opened (with rewind) for archive output and the archive process proceeds.

If a database archive is being created when you reply with cuu, the tape file on the virtual tape unit specified on the CMS FILEDEF command for ddname ARIARCH is opened for output and the archive process proceeds.

If a log archive is being created, when you reply with cuu, the tape file on the virtual tape unit specified on the CMS FILEDEF command for ddname ARILARC is opened for output and the archive process proceeds.

The cuu entered is not actually used in opening the file. The operator should enter the correct cuu, however, because it may appear in later messages (ARI0296A, ARI0294A, and ARI0298E).

Operator Response:

If the request is for a database archive file, you must use the virtual tape unit specified on the CMS FILEDEF command for ddname ARIARCH.

If the request is for a log archive file, you must use the virtual tape unit specified on the CMS FILEDEF command for ddname ARILARC.

Request (using the CP MSG OP command) that the CP operator select an available tape unit, mount and ready a scratch tape volume, and attach the tape unit to your virtual machine with the required virtual tape unit address. (If n=0 to 7, the TAPn is unit 180-7. If n=8 to F, the TAPn is unit 288-F) When the CP operator notifies you that these actions have been taken, enter the cuu of the virtual tape unit.

ARI2000E Index *index-id* in DBSPACE *dbspace-no* has been marked invalid.

Explanation: One of the following may have happened during ROLLBACK, UNDO, or REDO processing:

- The system cannot obtain a logical index page. This can result from the ROLLBACK or UNDO of either a DROP INDEX, a REORGANIZE INDEX, a deletion of a row, or an update of a row. It can also result from the REDO of either a CREATE INDEX, an insertion of a row, or an update of a row.
- The system cannot obtain a physical index page. This can result from the ROLLBACK or UNDO of a DROP INDEX, a REORGANIZE INDEX, a deletion of a row, or an update of a row. It can also result from the REDO of either a CREATE INDEX, an insertion of a row, or an update of a row.
- Not enough virtual storage or room in a temporary DBSPACE exists to recreate an index. The shortage can result from the ROLLBACK or UNDO of either a DROP INDEX or REORGANIZE INDEX, or from the REDO of a CREATE INDEX or REORGANIZE INDEX.

System Action: The database manager continues to process the ROLLBACK, UNDO, or REDO command

without updating the index and marks the index invalid.

Operator Response: Notify your system programmer.

System Programmer Response: Contact the owner of the invalid index. The owner must drop and recreate the index. The system has a limit of 16 invalid indexes. You can find the index name by using the following SELECT statement:

```
SELECT A.DBSPACENO, B.IID, B.INAME, B.ICREATOR,
      A.TNAME FROM SYSTEM.SYSCATALOG A,
      SYSTEM.SYSINDEXES B
      WHERE A.TNAME = B.TNAME AND
            A.CREATOR = B.CREATOR
            AND A.DBSPACENO = dbspace-no
            AND B.IID = index-id
      ORDER BY A.DBSPACENO, B.IID   ;
```

ARI2001I Warning: The following entities are invalid.

Explanation: This is a message heading for a list of invalid entities found during system start up. It is followed by the output of the SHOW INVALID command. For more information refer to the SHOW INVALID operator command.

System Action: Processing continues.

Operator Response: Notify and give your console output to your system programmer.

System Programmer Response: Notify the owners of the invalid indexes and have them drop and recreate the indexes.

ARI2002I Index pages are being reclaimed for DBSPACE *&1*.

Explanation: A free index page is required for a ROLLBACK, UNDO, or REDO work, but there are no free pages available. An attempt is being made to reclaim empty or partially empty pages from existing indexes.

System Action: If pages are reclaimed successfully processing will continue. If no pages are reclaimed the index will be marked invalid and processing will continue. Message ARI2003I will be issued when the page reclaim processing is complete.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Refer to message ARI2003I.

ARI2003I The index page reclaim processing is finished.

Explanation: A free index page was required for a ROLLBACK, UNDO, or REDO work, but there were no free pages available. An attempt to reclaim empty or partially empty index pages has completed.

System Action: If pages were reclaimed successfully processing will continue. If no pages were reclaimed the index will be marked invalid and processing will continue.

Operator Response: Refer this message to your system programmer.

System Programmer Response: This message indicates that the DBSPACE specified is running short of index pages. Index page reclaiming may adversely affect performance. At this time the SHOW DBSPACE command should be issued to monitor index page consumption. If there are still relatively few free index pages, then action should be taken to free some pages.

For more information, refer to the *DB2 Server for VSE & VM Database Administration* manual.

ARI2004I Warning: Error occurred while *action histype* log history area (HSDS). Reason Code = *nnn*. Return Code = *nn*.

Explanation: *action* indicates what action the database manager was doing on the log history area: archiving or restoring.

histype indicates which version of the log history area (file ARIHSDS PRECLDLG A or file ARIHSDS ARCHIVE A) the database manager was operating on: pre-COLDLOG or current version.

If you are processing a database or log archive, the database manager saves the log history area in file ARIHSDS ARCHIVE A.

If you are restoring a database, the database manager system tried to restore the log history area for one of the following reasons:

- The database manager cannot find a valid history area in the current log and is restoring the history area from file ARIHSDS ARCHIVE A.
- A database restore was requested immediately following a COLDLOG reconfiguration. The database manager is restoring the history area from file ARIHSDS PRECLDLG A.
- In the current log, the database manager cannot find the database to be restored. The database manager is restoring the history area from file ARIHSDS PRECLDLG A.

The Reason is determined by the function that the database manager attempted to perform, and the Code depends on the Reason. The possible values of Reason are:

100 An error occurred when the database manager issued a FILEDEF command on file ARILHSDS for the history area archive file. The Code value is the CMS return code from the FILEDEF.

101 An error occurred when the database manager

issued STATE for the history area archive file. The Code value is the CMS return code from the STATE.

102 The history area file being restored is corrupt.

103 An error occurred when the database manager issued access for the A-disk. The code value is the CMS return code from the access.

Other An error occurred when the system performed an operation on the history area archive file. The Reason value is the primary error code. The Code value is the secondary error code. For more information, see "Chapter 23. Sequential File Errors" on page 407.

System Action: If you issued a command to archive the history area, the database manager bypasses the remainder of the HSDS archive process, but the archive itself is unaffected. If you issued a command to restore the history area from ARIHSDS ARCHIVE A, the database manager issues this message before it terminates abnormally. If you issued a command to restore the history area from ARIHSDS PRECLDLG A, the database manager issues this message but continues with the restore process, using only the current log history.

Operator Response: Correct the problem by taking appropriate action based on the Reason and Code values specified. If you were doing a log archive, take a new log archive so that the HSDS copy on the A-disk reflects the previous log archive. If you were creating a log archive, and you have a critical need to be able to restore log archives at any time, ensure that the problem is fixed and a new log archive is created as soon as possible. If the user or the database manager was doing a database archive, no further action is required.

If you were doing a restore and the log history area was invalid, perform a COLDLOG reconfiguration and try the restore again. If the user was restoring an unknown database, the user must decide whether or not to restore the database without applying log archives (if applicable).

If you cannot solve the problem, refer it to your system programmer.

System Programmer Response: If you cannot identify the cause of the problem, contact the designated support group for your installation.

ARI2005I Log history area restored. Reason Code = *nnn*.

Explanation: The database manager restored the log history area (HSDS) on the log disks from ARIHSDS ARCHIVE A or ARIHSDS PRECLDLG A. The values of Reason Code are:

200 The log history area on the log disk is invalid.

201 The log history area on the log disk does not contain a record of the database archive that you want to restore. The history area saved prior to the COLDLOG has been restored. If a restore is requested immediately after a COLDLOG RECONFIGURATION, the pre-COLDLOG history area is restored automatically. If a restore is requested and the database archive is found only on the pre-COLDLOG history area, this history area is restored only upon request.

System Action: The current operation continues.

Operator Response: If you know that the database log file was destroyed and recreated with COLDLOG, ignore this message. Otherwise, inform your system programmer.

System Programmer Response: If you did not realize that a log was damaged or if you expect the database manager to restore the HSDS, check the previous console logs for the DB2 Server for VM database machine and ensure that no unexpected errors occurred that are related to the log.

Issue the SHOW LOGHIST command and verify that the recent entries agree with the database activity. If they do not, do a database archive.

ARI2006I Warning: Error occurred while copying log history area (HSDS). Copy File Return Code = *nn*.

Explanation: The database manager attempted to copy the current history area in disk file ARIHSDS ARCHIVE A to file ARIHSDS PRECLDLG A and the copy failed. The code value is the CMS return code from COPYFILE. For further explanation of the code value, look up the displayed error codes under "Chapter 23. Sequential File Errors" on page 407.

System Action: The COLDLOG process continues.

Operator Response: Correct the problem by taking appropriate action based on the code value specified. If the pre-COLDLOG information is necessary, copy file ARIHSDS ARCHIVE A to file ARIHSDS PRECLDLG A by using the CMS COPYFILE command.

System Programmer Response: If you cannot identify the cause of the problem, contact the designated support group for your installation.

ARI2007D The database archive currently mounted was taken before the last COLDLOG reconfiguration. If this archive is restored, the log history before the COLDLOG will replace the current log history. Enter either IGNORE to continue, or CANCEL to restart the restore process.

Explanation: The database manager cannot find the database archive to restore in the history area of the

current log. The database manager found the database archive in the history area of the disk file ARIHSDS PRECLDLG A, which was current before the COLDLOG reconfiguration. If the operator continues the restore process, the pre-COLDLOG history area will replace the history area in the current log. Because the history area containing the database archive is available, subsequent log archives are identified so that they can be applied as part of the restore process.

System Action: The restore process continues if requested.

Operator Response: Decide whether to continue the restore process or not. Enter IGNORE or CANCEL.

ARI2008I Archive is about to be started.

Explanation: The operator or database administrator requested a database or log archive with one of the following commands:

```
LARCHIVE
SQLEND LARCHIVE
ARCHIVE
SQLEND ARCHIVE
```

This message follows message ARI0065I. It informs the user that although the operator or database administrator requested an archive, the actual archive process has not started.

System Action: As soon as the system is ready, the database or log archive begins.

ARI2009E To continue with LOGMODE=A, you must first perform an initial database archive.

Explanation: You have been running with LOGMODE Y or N. If you want LOGMODE=A, you must first perform an initial database archive because the log indicates that no database archive exists.

System Action: The application server ends.

Operator Response: Start the application server with LOGMODE=Y and take a database archive. Next, restart the application server with LOGMODE=A.

ARI2010I WARNING: The database manager terminated abnormally. The log contents is required for recovery. Continuing with the COLDLOG will erase all log records and may cause the next startup to terminate during UNDO/REDO processing.

Explanation: The last time the application server started up, it terminated abnormally. The current log is required to recover the system to the point of failure, either by warmstarting the database or by restoring the last database archive.

System Action: Processing continues. The user will be given an opportunity to cancel the COLDLOG by message ARI0920D.

User Response: Cancel the coldlog in response to message ARI0920D. Warm start the database to recover up to the point of failure or restore the database from a previous archive.

ARI2012I The machine is not in XC-mode. Parameter *parmid* is ignored.

Explanation: The DB2 VM Data Spaces Support parameter is not valid with a 370 or XA-mode machine.

parmid identifies the DB2 VM Data Spaces Support parameter involved.

System Action: Initialization continues. DB2 VM Data Spaces Support will not be used.

Operator Response: Ensure that the **System Action** is what you intended.

ARI2014E The directory block size is invalid because the machine is not in XC-mode.

Explanation: Directory consists of 4KB blocks. This block size is not valid in a 370 or XA-mode machine.

System Action: Processing ends.

Operator Response: Issue the CP command SET MACHINE XC and re-IPL CMS to enable DB2 VM Data Spaces Support for directory, or reformat the directory into 512-byte blocks using the SQLCDBEX EXEC to run without DB2 VM Data Spaces Support.

ARI2015I The storage pool specification input file was not found. The database manager will use the default values.

Explanation: During startup, SQLSTART issued the following FILEDEF : 'FILEDEF ARISPOOL DISK *dbname* ARISPOOL * (NOCHANGE'. If the user issued a different FILEDEF for DDNAME ARISPOOL, SQLSTART will not override it. If this message is issued, then the file identified by DDNAME ARISPOOL was not found. The database manager will use system default values for all storage pools and continue processing.

System Action: Processing continues.

User Response: Ensure that the database manager is intended for starting without the storage pool specification input file, or create a file containing storage pool specification values. This file must have the name *dbname* ARISPOOL. Alternatively, a FILEDEF must be issued for a differently named file using DDNAME ARISPOOL before invoking SQLSTART.

ARI2017E **An error occurred when the database manager issued macro** *string1 string2***. Return Code =** *return-code***.**

Explanation: An error occurred when a DB2 VM Data Spaces Support macro was issued.

string1 string2 identify the DB2 VM Data Spaces Support macro.

return-code is the system return code. Please refer to the *VM/ESA: CP Programming Services* manual.

System Action: Processing ends.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the problem by taking appropriate action based on the Return Code specified and the DB2 VM Data Spaces Support macro involved. If you cannot identify the cause of the problem, contact the designated support group for your installation.

ARI2018E **Device number** *cuu* **is not supported by the VM/ESA system. Storage pool** *pool-num* **cannot be mapped to data spaces.**

Explanation: Data Spaces cannot be used, because the specified device is not supported by DB2 VM Data Spaces Support.

cuu identifies the device.

pool-num identifies the pool number.

System Action: Processing ends.

Operator Response: Turn DB2 VM Data Spaces Support off for the storage pool containing the unspported device.

ARI2019E **Device number** *cuu* **is not supported by the VM/ESA system. The directory cannot be mapped to data spaces.**

Explanation: Data Spaces cannot be used, because the specified device is not supported by DB2 VM Data Spaces Support.

cuu identifies the device.

System Action: Processing ends.

Operator Response: Do not use DB2 VM Data Spaces Support with the directory. Reblock the directory to 512 bytes, or copy the directory to a supported device before restarting the application server.

ARI2020I **The machine is not in XC-mode. Data spaces will not be used.**

Explanation: DB2 Server for VM storage pools cannot use DB2 VM Data Spaces Support if the database machine is not in XC-mode.

System Action: Processing continues. DB2 VM Data Spaces Support will not be used for storage pools.

ARI2021E **{OPEN |READ(GET)|CLOSE} error occurred during processing of the storage pool specification input file. Reason Code =** *primary-code* **Return Code =** *secondary-code***.**

Explanation: An error has occurred during database processing of the DDNAME 'ARISPOOL' used for the storage pool specification input file.

The Reason Code value is the Primary Error Code. The Return Code value is the Secondary Error Code. For further explanation, look up the displayed error codes in the "Include File Access Errors" section or "INCLUDE FILE ERROR" in the HELP text.

System Action: Processing is terminated.

Operator Response: Correct the problem as identified by the Reason Code and the Return Code.

ARI2022I **The database manager is using data spaces for the directory.**

Explanation: Directory is being accessed through the use of a data space since directory block size is 4K bytes.

System Action: Directory is mapped to data spaces. Processing continues.

ARI2023I **Storage pools cannot use data spaces when the application server is started with STARTUP=E.**

Explanation: The pool specification file indicates that some or all storage pools are to use DB2 VM Data Spaces Support. However, when the application server is being started for adding or deleting dbextents (STARTUP=E), DB2 Server for VM storage pools cannot use DB2 VM Data Spaces Support.

System Action: Initialization continues. Data spaces will not be used for storage pools.

ARI2024I **Storage pools cannot use data spaces when the application server is started with STARTUP=R.**

Explanation: The pool specification file indicates that some or all storage pools are to use DB2 VM Data Spaces Support. However, when the application server is being started for Archive Restore (STARTUP=R),

storage pools cannot use DB2 VM Data Spaces Support.

System Action: Archive Restore continues. DB2 VM Data Spaces Support will not be used for storage pools.

ARI2025E **Invalid or missing storage pool specification** *string1* **[in the storage pool specification input file. Line Number =** *number***].**

Explanation: The storage pool specification *string1* is not valid.

If the error occurred during storage pool specification input file processing, the message indicates the line number in the storage pool specification input file where the error occurred.

System Action: If processing an operator command, the command is ignored. If processing the storage pool specification input file, the storage pool specification will be ignored, the remainder of the storage pool specifications will be verified, and the application server will subsequently terminate processing.

Operator Response: Verify that proper and valid parameters are specified. It must be a valid storage pool value (nn), range (nnn-mmm), Working Storage residency priority ('DSn' where 'n' is between 1 and 5 inclusive), or specification (such as 'SEQ', 'STR', 'BLK', or 'DSn' where 'n' is between 1 and 5 inclusive).

Note: The specification 'BLK' is not valid for operator command line invocation.

ARI2026I **Some or all storage pools will use data spaces.**

Explanation: The database machine has determined that some or all storage pools will use DB2 VM Data Spaces Support. You can alter the storage pool specification input file before starting the application server to specify which pools will and which pools will not use DB2 VM Data Spaces Support. By default, all pools use DB2 VM Data Spaces Support. The SHOW POOL command gives information on data space use for all storage pools.

Note: The value of the startup parameter SEPINTDB as well as the specification for the storage pool to which internal dbspaces have been assigned determine whether internal dbspaces will use DB2 VM Data Spaces Support or not, and whether they will be mapped or unmapped.

System Action: Processing continues.

Operator Response: Ensure that this is intended.

ARI2027I **No storage pools will use data spaces.**

Explanation: The database machine has determined that no storage pools will use DB2 VM Data Spaces Support. You can alter the storage pool specification input file before starting the application server to specify which pools will and which pools will not use DB2 VM Data Spaces Support. By default, all pools use DB2 VM Data Spaces Support. The SHOW POOL command gives information on data space use for all storage pools. Some startup conditions will cause data spaces to not be used, thus overriding what was specified in the storage pool specification input file. A previous message will have informed you of such a condition.

Note: Internal dbspaces will still use separate unmapped data spaces if the startup parameter SEPINTDB has the value Y and startup conditions still allow the use of data spaces.

System Action: Processing continues.

Operator Response: Ensure that this is intended.

ARI2028E **Error detected using MVPG instruction. Using MVCL instruction instead.**

Explanation: The database machine detected an error when using a MVPG instruction to move a buffer page to a data space.

System Action: The database machine recovered by using the MVCL instruction to move the data.

Operator Response: Notify the system programmer.

System Programmer Response: Make a record of the error and contact the designated support group for your installation.

ARI2029E *SET-keyword* **= {***parameter* **|** *,***} is invalid. It must be in the range from** *lower-limit* **to** *upper-limit***.**

Explanation: **CMD** indicates that the parameter in error was specified in the operator command.

SET-keyword is the operator SET command keyword which has the invalid parameter value. It must be one of the following keywords: SAVEINTV, or TARGETWS. *parameter* is the invalid parameter value. If no value was specified, a comma is substituted for the missing value. If the parameter is longer than eight characters, only the first eight characters are displayed. The parameter value can be up to eight characters. Leading zeros are suppressed.

lower-limit is the lower limit (or minimum) for the correct parameter value.

upper-limit is the upper limit (or maximum) for the correct parameter value.

```
            lower   upper
yyyyyyyy    limit   limit     (default)
--------    -----  --------   ---------
SAVEINTV     1      999999       10
TARGETWS     1       99999       32
```

System Action: Processing continues.

ARI2030E **Device number** *device-number* **residing on FBA DASD does not start or does not end on a 4k block boundary. Storage pool** *storage-pool* **cannot be mapped to data spaces.**

Explanation: The *device-number* is a minidisk residing on a fixed block architecture (FBA) DASD device. The minidisk does not start or does not end on a 4K block boundary. (On VM/ESA, the minidisk's starting FBA block number and its ending FBA block number plus one must each be evenly divisible by eight) Please refer to the *VM/ESA: CP Programming Services* manual.

System Action: Processing ends.

Operator Response: You have two choices:
1. Turn off data spaces support for the *storage-pool*.
2. Refer this message to your system programmer.

System Programmer Response: You have two choices:
1. To continue using an FBA device, use DDR or use the SQLCDBEX EXEC to ensure that it is formatted on a 4k block boundary.
2. Move to a non-FBA device, using the SQLCDBEX EXEC.

For information about using the SQLCDBEX EXEC, see the *DB2 Server for VM System Administration* manual.

ARI2033E **The database manager has detected an I/O error. Error Type =** *type* **Error Code =** *code* **Data Space ASIT =** *asit* **[Address =** *address***]**

Explanation: The database manager has detected an I/O error that occurred while reading or writing via data spaces.

type specifies during what activity the error occurred and is one of 'MVC', 'MVCL', 'MVPG', 'MAPMDISK SAVE', 'CP paging', or 'N/A'.

code is the MAPMDISK SAVE Completion Code if *type* is 'MAPMDISK SAVE', otherwise it is the CMS Abend Code identifying the error. It is displayed in hexadecimal form.

asit is the address space identification token of the data space in which the error occurred. It is displayed in hexadecimal form.

address is the data space address where the error occurred. It is displayed in hexadecimal form.

If *type* is 'MAPMDISK SAVE' and *code* is 8, the data space specified by *asit* does not exist. In this case the "Address = ... " portion of the message will not be shown.

In all other cases, this error indicates that a DASD-related hardware failure has occurred. Again, the "Address = ..." portion will not be shown if the database manager did not receive the address from CP and therefore could not display it.

If *type* is 'MVC', 'MVCL' or 'MVPG', the error occurred when the database manager attempted to move data from or to a data space, causing a page fault which could not be completed. If *type* is 'MAPMDISK SAVE', the error occurred when CP attempted to save pages in a data space to disk as instructed by the database manager. If *type* is 'CP paging', the error occurred when CP attempted to save a page to disk as part of CP's regular paging activities. If *type* is 'N/A', the database manager was not able to determine the type.

In all cases except when *type* is 'CP Paging' or when *address* is not shown, message ARI2035I will follow this message, providing additional information about the error. Message ARI2034I may also appear.

System Action: If the *type* is 'CP paging', processing continues. Otherwise, additional messages as explained above may appear and the application server will terminate.

Operator Response: If *type* is 'CP paging', you may choose to terminate the application server yourself to allow the hardware failure to be found and fixed. In all cases, record the messages received and notify your System Programmer.

System Programmer Response: If *type* is MAPMDISK SAVE and *code* is 8, the data space referred to does not exist. Report this error to the designated support group for your installation. Otherwise, ensure that no one has detached any of the database machine's disks while the database is running (an error will occur even if the disk was later reattached). If this is not the case, the message indicates a DASD-related hardware failure and may be accompanied by other DB2 Server for VM and non-DB2 Server for VM messages. Determine the cause of the failure and take appropriate action.

ARI2034I **This ASIT and address correspond to dbspace** *dbspace***, page** *page***.**

Explanation: This message follows message ARI2033E and provides additional information that may be useful for diagnosing the reported error.

dbspace and *page* are the dbspace number and logical page number within the dbspace that correspond to the given ASIT and address reported by message ARI2033E. If these numbers are both -1, the database manager could not determine the dbspace and page numbers.

System Action: Processing ends.

ARI2035I This page resides on *location*.

Explanation: This message follows message ARI2033E and provides additional information that may be useful for diagnosing the reported error.

location is the location on disk corresponding to the ASIT and address provided in message ARI2033E. *location* will be one of the following phrases in the displayed message:

- **system paging DASD** - the error occurred somewhere on the system paging disks. The database manager has no additional information about CP's use of paging DASD.
- **the directory disk, block** *block* - the error occurred on the directory disk. *block* is the number of the block on which the error occurred.
- **dbextent** *dbextent*, **block** *block* - the error occurred on a DB2 Server for VM extent. *dbextent* and *block* indicate the extent number and block on the extent where the error occurred. If these numbers are both -1, the database manager could not determine the extent and block numbers.

System Action: Processing ends.

ARI2037D Restoring the database with LOGMODE=Y or N. Enter 0 to CANCEL and restart with logmode A or L, or enter 1 to CONTINUE.

Explanation: The restore process is started with LOGMODE=Y or N. You can enter `0` to cancel the restore process and restart with logmode A or L. If you enter 1 to continue, the restore process will continue to restore the Directory and the DBEXTENTs. The restore set for this database archive will also be displayed if there are log archives to be restored. Message ARI0280A will then be issued and the application server terminated. Also, the current log could be lost.

System Action: Processing continues.

Operator Response: Enter `0` to cancel the restore, and restart the restore process with logmode A or L, or enter 1 to continue.

Attention: The current log could be lost.

ARI2038E Archive being restored is from a different release of DB2 Server for VM. The restore has failed.

Explanation: The system has determined that the archive being restored was not taken with the current release of DB2 Server for VM.

System Action: The restore process ends. Note that the directory has been restored and the database is unusable.

Operator Response: Restore an archive taken with the current release of DB2 Server for VM.

ARI2039E CHKINTVL = *newval* is invalid. Maximum valid CHKINTVL value is *maxval*.

Explanation: The checkpoint interval is the number of log pages which will be written between automatic checkpoints. When this value is changed during run time, an analysis of the checkpoint interval and the size of the log is made to determine if at least one checkpoint will be taken before end of log or log wrap. This message is generated if the **CHKINTVL** specified will not cause at least one checkpoint to be taken.

System Action: The checkpoint interval is not changed and processing continues.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Correct the invalid **CHKINTVL** value. *maxval* represents the highest allowable checkpoint interval value based on the size of the log.

ARI2040E Pool Recovery is in progress, STARTUP=*startup* is not allowed.

Explanation: The Data Restore storage pool recovery process was initiated, but has not yet been completed by starting the database manager with STARTUP=U. Storage Pool recovery must be completed before the database can be started with STARTUP=*startup*.

System Action: The requested function is not performed.

Operator Response: Complete the storage pool recovery process by starting the database with STARTUP=U.

ARI2041I Pool Recovery processing continues.

Explanation: The Data Restore storage pool recovery process was initiated and completed successfully. The database is being started with STARTUP=U, which will complete the storage pool recovery process.

System Action: Storage pool recovery processing continues.

ARI2042E Pool Recovery cannot continue.

Explanation: The Data Restore storage pool recovery process was initiated, but did not complete successfully. The database cannot be started with STARTUP=U to complete the storage pool recovery process until the Data Restore pool recovery process is successfully completed. The database is currently in an inconsistent state. One of the following must be done before the database can be restarted.

- Use the RESTORE POOL command of the Data Restore feature to complete the initial phase of

storage pool recovery successfully. Then, start the database with STARTUP=U to complete storage pool recovery.

- Restore the entire database from a database archive. This can be either a database manager archive or a user archive.

System Action: Processing terminates.

ARI2043E During Pool Recovery, END RESTORE will not be accepted.

Explanation: The message displayed before ARI2043E indicates that END RESTORE is an acceptable reply to the message. However, during Pool Recovery, END RESTORE will be rejected as an invalid response.

System Action: The system is waiting for an operator response.

Operator Response: Enter a response to the previous message. If END RESTORE is entered, it will not be accepted as a valid response.

ARI2044D Agent identifier *n* in the FORCE command was using DRDA Two Phase Commit protocol. Forcing this agent may lead to heuristic damage. Do you want to force this agent? Enter 0(No) or 1(Yes).

Explanation: The in-doubt agent *n* was using the DRDA two phase commit protocol. Forcing this agent's work could cause data on other databases to be inconsistent with data at this database.

System Action: If you reply 1 for YES, the agent will be forced to commit or rollback its work as indicated on the FORCE command. If you reply 0 for NO, the in-doubt agent will not be forced and will remain in-doubt. If you do not reply 1 or 0, message ARI0052E will be displayed and message ARI2044D will be redisplayed.

Operator Response: Reply 1 if you are sure that you want to force this in-doubt agent. Reply 0 if you are unsure. If you need to force the in-doubt agent, the FORCE command can be reentered.

Database Administrator Action:

The database administrator must understand the application and the status of this unit of work on all of the involved application servers before forcing an in-doubt agent that was using DRDA two phase commit protocol. It is most desirable to let the systems resolve the in-doubt agents automatically during resynchronization recovery. The database administrator of the DRDA2 requester should be contacted to help determine the impact of forcing the in-doubt agent.

ARI2045E RESET INDOUBT command has insufficient number of parameters.

Explanation: The RESET INDOUBT operator command was entered without parameters.

System Action: The command processing is ended.

Operator Response: Reenter the RESET INDOUBT operator command with the appropriate operands.

ARI2046E Too many RESET INDOUBT command input parameters.

Explanation: The RESET INDOUBT * operator command was entered with extra parameters. When * is specified, no other parameters are allowed.

System Action: The command processing is ended.

Operator Response: Reenter the RESET INDOUBT operator command with the appropriate operands.

ARI2047E Transaction identifier *n* in RESET INDOUBT command is invalid.

Explanation: The transaction identifier *n* does not exist in the system.

System Action: The command processing is ended.

Operator Response: Enter the SHOW INDOUBT operator command to find out the valid, existing transaction identifiers. Then reenter the RESET INDOUBT operator command with the appropriate operands.

ARI2048D In-doubt *n* was forced. Are you sure you want to reset this in-doubt? Enter 0(No) or 1(Yes).

Explanation: The transaction identifier *n* was in-doubt and was resolved by using the FORCE operator command. The action taken by the FORCE command could be different than the action that resynchronization recovery would have done. This unit of work might still be in-doubt on the other participating application servers.

System Action: If you reply 1 for YES, the in-doubt entry will be removed from the system. If you reply 0 for NO, the in-doubt entry will not be removed from the system. If you do not reply 1 or 0, message ARI0052E will be displayed and message ARI2048D will be redisplayed.

Operator Response: Reply 1 if you are sure that you want to remove this in-doubt entry. Reply 0 if you are unsure. If you need to remove the in-doubt entry, the RESET INDOUBT command can be reentered.

Database Administrator Action:

The database administrator must understand the application and the status of this unit of work on all of

the involved application servers before removing an in-doubt entry that was forced. It is most desirable to let the systems resolve the in-doubt agents automatically during resynchronization recovery. The database administrator of the DRDA2 requester should be contacted to help determine the impact of forcing the in-doubt agent.

ARI2049I **FORCE RINIT may only be entered when Resychronization Initialization is active.**

Explanation: The operator entered the "FORCE RINIT" command to force the abnormal termination of *resynchronization initialization*. However, the database is not currently performing *resynchronization initialization*.

Note: The SHOW CONNECT operator command may have indicated that *resynchronization initialization* was active. However it ended before the FORCE command was issued.

System Action: Command processing is terminated.

Operator Response: Issue the SHOW CONNECT command to verify the status of *resynchronization initialization*.

ARI2050I **FORCE RREC may only be entered when Resychronization Recovery is active.**

Explanation: The operator entered the "FORCE RREC" command to force the abnormal termination of *resynchronization recovery*. However, the database is not currently performing *resynchronization recovery*.

Note: The SHOW CONNECT operator command may have indicated that *resynchronization recovery* was active. However it ended before the FORCE command was issued.

System Action: Command processing is terminated.

Operator Response: Issue the SHOW CONNECT command to verify the status of *resynchronization recovery*.

ARI2051I **FORCE {RINIT|RREC} is already scheduled.**

Explanation: If FORCE RINIT is displayed, then resynchronization initialization is being terminated by the database manager. If FORCE RREC is displayed, then resynchronization recovery is being terminated by the database manager.

System Action: Command processing is terminated.

ARI2052I **FORMAT 1**
A checkpoint was taken.
Scheduled: *sctime*
Started: *sttime*
Ended: *etime*
FORMAT 2
A checkpoint was taken.
Type = *type***.**
Scheduled: *sctime*
Started: *sttime*
Ended: *etime*
DIRREAD = *drd*
DIRWRITE = *dwr*
PAGEREAD = *prd*
PAGWRITE = *pwr*
DSREAD = *dsr*
DSWRITE = *dsw*
Time spent waiting to start: *wst* **seconds**
Time spent executing: *ext* **seconds**
Log pages filled since last checkpoint: *logused*

Explanation: Format 1 of the message is displayed if DSPSTATS=1. Format 2 of the message is displayed if DSPSTATS=2. The DSREAD and DSWRITE values are only displayed if the VM data spaces feature is installed and is being used.

type is the reason the checkpoint was performed. The type is only displayed if DSPSTATS=2. Valid types are:

COLDLOG
Log just formatted.

Warm Start
System just recovered (that is, warm start).

Periodic
Periodic system checkpoint taken. Periodic checkpoints can occur for various reasons, including, CHKINTVL was reached, SLOGCUSH was reached, SOSLEVEL was reached, and a DROP DBSPACE was issued. See the *DB2 Server for VSE & VM Performance Tuning Handbook* manual for more details.

Database Archive Started
Archive of the database started.

Database Archive Completed
Archive of database completed.

Log Archive Started
Archive of the log started.

Log Archive Completed
Archive of the log completed.

Shutdown
System being shutdown.

Shutdown with User archive
Shutdown with user archive.

User Archive Done
User archive done and verified.

Logmode=N
System checkpoint when LOGMODE=N.

sctime is the time of day the checkpoint was scheduled.

sttime is the time of day the checkpoint started execution.

etime is the time of day the checkpoint ended execution.

drd is the number of directory buffer reads that were done by the checkpoint processing.

dwr is the number of directory buffer writes that were done by the checkpoint processing.

prd is the number of page buffer reads that were done by the checkpoint processing.

pwr is the number of page buffer writes that were done by the checkpoint processing.

dsr is the number of data space reads that were done by the checkpoint processing. This is only displayed if the VM data spaces feature is being used.

dsw is the number of data space writes that were done by the checkpoint processing. This is only displayed if the VM data spaces feature is being used.

wst is the time that the checkpoint was waiting to begin processing. This is only displayed if DSPSTATS=2.

ext is the time that the checkpoint was processing. This is only displayed if DSPSTATS=2.

logused is the number of log pages filled since the last checkpoint was taken.

System Action: System processing continues.

User Response: Use the SET DSPSTATS command to disable these messages if they are not desired, or to change which format of the message appears.

ARI2053E The mandatory COLDLOG during migration was not done. A COLDLOG must be performed before continuing.

Explanation: The database startup has detected that a COLDLOG was not performed during migration. Failure to COLDLOG the database will prevent restoring database archives. The database cannot be started until a COLDLOG has been performed.

System Action: Database startup has terminated.

Operator Response: Refer this message to your system programmer.

System Programmer Response: You must perform a COLDLOG before the database manager will allow the database to finish startup.

ARI2054I Database manager has taken a FULL archive.

Explanation: The database manager has completed taken a FULL archive. Either an implicit archive was initiated by the database manager, or the ARCHIVE operator command was used with the FULL parameter.

System Action: Processing continues.

Chapter 4. ARI2900–ARI2916 Common Messages

ARI2900E **Language *&1*. cannot be accessed. Reason Code = *nn*.**

Explanation: Either the national language name has been misspelled, or the language specified on the SET LANGUAGE command cannot be accessed because of one of the following reasons:

01 Message file not found.

02 Storage problem.

03 FINDSYS/SEGMENT LOAD system error.

04 Language cannot be installed because:
- Language ID is not in table SYSLANGUAGE.
- Language ID is in SYSLANGUAGE but was inserted into SYSLANGUAGE during a DB2 Server for VM session.

&1. is the language identifier.

System Action: An attempt to set language &1. failed, because the language specified could not be found.

User Response: Make sure that the language identifier is correct and the requested language is supported on your system.

An INSERT may have been made into table SYSLANGUAGE during a DB2 Server for VM session. You must bring the DB2 Server for VM machine down and bring it up again in order for the inserted language IDs in SYSLANGUAGE to be recognized.

ARI2901I **Warning: The database manager cannot access the default language *&1*.. English will be used instead. Reason Code = *nn*.**

Explanation: The message file for the language currently set in VM could not be loaded and the attempt to set the DB2 Server for VM default language also failed. The English message repository is used instead. Reason codes have the following meaning:

01 Message file not found

02 Storage problem

03 FINDSYS/SEGMENT LOAD system error.

04 Language not installed.

ARI2902E **There is no message file that the database manager can access.**

Explanation: This message appears when all of the following conditions have been met:

- The message repository identified by the current CMS language setting could not be accessed or loaded.
- The repository corresponding to the DB2 Server for VM default language, defined as the first entry in the ARISNLSC MACRO Q file, could not be accessed or loaded.
- The repository corresponding to American English (LANGID: AMENG, LANKEY: S001) could not be accessed or loaded.
- Message file not found
- Storage problem
- FINDSYS/SEGMENT LOAD system error
- Language not installed

This message is always issued in English.

System Action: Subcomponent initialization is cancelled.

User Response: Notify your system programmer.

System Programmer Response: Ensure that the repository corresponding to the current CMS language is available, or that the first entry of the ARISNLSC MACRO Q file has a corresponding available repository, or that the repository for AMENG is available. Finally, verify that you have enough storage to load the message repository. Refer to the *DB2 Server for VM Program Directory* for more information on installing a language on your system.

ARI2903E **Error loading Resource Adapter. SQLCODE =** *&1A*

Explanation: Resource adapter could not be loaded. This message is always issued in English.

System Action: Subcomponent initialization is cancelled.

User Response: Notify your System Programmer.

Programmer Response: Correct the problem indicated by the SQLCODE value. For more information, see "Chapter 19. SQLCODEs" on page 277.

ARI2905E **The *command* command is not supported when using the DRDA protocol.**

Explanation: You are accessing a non-DB2 Server for VM application server or you are using the DRDA protocol, and have attempted to use the specified command against this application server. The command is available only against a DB2 Server for VM application server using a non-DRDA protocol.

System Action: Command processing failed.

© Copyright IBM Corp. 1987, 2000

User Response: Enter another command.

ARI2906I **The only valid isolation level is CS when the DRDA protocol is used. Isolation level of CS is now in effect.**

Explanation: If the DRDA protocol is used, the isolation level defaults to cursor stability (CS). The SET ISOLATION command cannot be used to change the isolation level.

System Action: The SET ISOLATION command is ignored.

User Response: You may continue with another command.

ARI2909I **Warning: An error in the conversation has been detected. Please wait for the data capture completion message to appear before taking further action. LUWID = *&1*.**

Explanation: An error has been detected in either the communication partner's datastream or internally using DRDA protocol processing.

System Action: System information is dumped to a console file. The logical unit of work identifier may be used to correlate this message to the console file.

User Response: Wait until completion message ARI2910I is displayed, then notify your system programmer. DO NOT erase any console files that may exist.

System Programmer Response: This message indicates that an error has occurred during DRDA protocol processing, and diagnostic error information has been dumped to a console file. Save the console file containing the dump identified by the logical unit of work identifier in the message and contact the designated support group for your installation.

ARI2910I **Data capture process completed successfully. Please notify your system programmer.**

Explanation: The data capture process referred to in message ARI2909I is completed.

ARI2911E **An internal error has occurred in the data capture process. Notify your system programmer. LUWID = *&1*.**

Explanation: An error has occurred during the data capture process. This indicates a serious system problem.

System Action: Process of data capture is interrupted. A console file containing partial data may or may not exist.

User Response: Notify your system programmer. Do NOT erase any console files that may exist.

System Programmer Response: This message indicated that a DB2 Server for VM internal error has occurred during data capture. There may or may not be a console file containing a partial dump identified by the logical unit of work identifier in the message. Save this file if it exists. Contact the designated support group for your installation.

ARI2912E **Error reading NETID from SNA NETID file.**

Explanation: You have specified PROTOCOL=AUTO which requires a NETID value. An error occurred when the SQLSTART EXEC tried to retrieve the NETID value from the SNA NETID file on the production disk.

System Action: Database startup is terminated because the NETID is a required parameter for DRDA protocol.

User Response: Check the existence of the SNA NETID file on the production disk. If it is not found or corrupted, then recreate the file with the correct NETID value and rerun the EXEC.

ARI2913I **Character translation table for CHARNAME = *charname* cannot be found. The character translation table for INTERNATIONAL will be used.**

Explanation: The row with the CHARNAME value cannot be found in the CMS file ARISSCR MACRO on the production minidisk, or the file ARISSCR MACRO does not exist. If the CHARNAME value is 'unknown', an error has occurred while retrieving the GLOBAL variable CHARNAME from the LASTING GLOBALV file.

System Action: Processing continues. The character translation table for INTERNATIONAL will be used.

User Response: If the CHARNAME value is 'unknown', re-run SQLINIT with a correct CHARNAME. Contact your system programmer if the specified CHARNAME cannot be found in ARISSCR MACRO on the system production minidisk, or if the file ARISSCR MACRO does not exist.

System Programmer Response: Check that the row corresponding to the CHARNAME in the user's LASTING GLOBALV file exists in the flat file ARISSCR MACRO on the production minidisk. If the row does not exist, or if the flat file does not exist, re-create the flat file ARISSCR MACRO. Refer to the *DB2 Server for VM System Administration* manual.

ARI2914E **An error occurred with the CMS MACRO *name* while accessing the file *filename filetype*. Return code = *rc***

Explanation: An error occurred with the CMS MACRO while accessing the file *filename filetype* that is

required for character translation support or for CCSID support. *filename filetype* may have one of the following values: ARISCCS MACRO, ARISSTR MACRO or ARISSCR MACRO. The return code value is the CMS MACRO return code.

System Action: Processing terminates.

User Response: Contact your system programmer.

System Programmer Response: Refer to the *VM/ESA: CMS Application Development Reference* for an explanation of the return code and correct the problem. If the problem cannot be corrected, contact the designated support group for your installation.

ARI2915E An error occurred when the application attempted to {get | free} dynamic storage.

Explanation: The application encountered an error in either getting or freeing storage needed for work areas.

System Action: Processing terminates.

User Response: Contact your system programmer.

System Programmer Response: If this is a get failure, increase the size of the user machine. Otherwise, refer to *VM/ESA: CMS Command Reference*

ARI2916E An error occurred with the CMS command *name*. Return code = *rc*.

Explanation: An error occurred with the CMS command *name*. The return code *rc* is the CMS command return code.

System Action: Processing terminates.

User Response: Contact your system programmer.

System Programmer Response: Refer to the appropriate *VM/ESA: CMS Command Reference* manual for an explanation of the return code, and correct the problem. If the problem cannot be corrected, contact the designated support group for your installation.

Chapter 5. ARI0300–ARI0399 CMS EXEC Messages

ARI0300A **How many nonrecoverable storage pools do you want?**

Explanation: The SQLDBGEN EXEC or the SQLADBEX EXEC is asking for the number of storage pools to be defined as nonrecoverable.

System Action: Processing continues after you reply with a valid number, 0 to 998.

User Response: Decide how many nonrecoverable storage pools you want and enter that number.

ARI0301A **Please enter a storage pool number to be specified as a nonrecoverable storage pool.**

Explanation: The SQLDBGEN EXEC or the SQLADBEX EXEC is asking for a specific storage pool number that you wish to have defined as a nonrecoverable storage pool.

Note: Message ARI0301A will be displayed n times, where n is the reply you entered for message ARI0300A, allowing you to enter the values for all nonrecoverable pools you wish to define.

System Action: Processing continues after you reply with a valid number, 2 to 999.

User Response: Decide which storage pool you want to designate as a nonrecoverable storage pool and enter its number. Storage pool number 1 cannot be used as a nonrecoverable storage pool.

ARI0302E **The number you entered is out of range. Nonrecoverable storage pools must be in the range of 2 to 999.**

Explanation: The SQLDBGEN and SQLADBEX EXECs display this message after your response to message ARI0301A. The numeric value you entered was less than 2 or greater than 999. The numeric value must be greater than 1 and less than 1000.

System Action: Message ARI0301A is redisplayed, prompting you for a nonrecoverable storage pool number.

User Response: In response to message ARI0301A, enter a number between 2 and 999.

ARI0303E &1 **is not correct** &2 &3 &4 &5 &6 &7 &8 &9 &10**.**

Explanation: &1 is the incorrect value detected by the SQLDBGEN or SQLADBEX EXECs. &2 through &10 is explanatory text. For example:

ARI0303E 999 IS NOT CORRECT FOR THE NUMBER OF POOLS YOU WANT.
ARI0303E 2 IS NOT CORRECT FOR THE NUMBER OF END STATEMENTS IN SQLDBA SQLDBGEN A.

System Action: The EXEC prompts you to retry the function in which the error was detected.

User Response: Read and respond to the prompt that follows message ARI0303E. If there is no informational message that helps you determine the correct value, consult the *DB2 Server for VM System Administration* manual for the database generation process or the ADD DBEXTENT process.

ARI0304I **Storage pool** &1 **will be a nonrecoverable storage pool.**

Explanation: ARI0304I is displayed by the SQLDBGEN and SQLADBEX EXECs. &1 is the storage pool number that will be defined to the database as a nonrecoverable storage pool.

System Action: Processing continues.

ARI0305E **The number of DBEXTENTS specified in the SQLFDEF file of** &1 **does not match the CUREXTNT keyword statement value of** &2 **in the** &3 **SQLDBGEN A file.**

Explanation: The SQLDBGEN EXEC displays this message when there is a mismatch between the dbname SQLFDEF file and the dbname SQLDBGEN A file regarding the number of current DBEXTENTs defined. &1 is the current number of DBEXTENTs defined in the dbname SQLFDEF file (residing on the production disk). &2 is the current number of DBEXTENTs specified by the CUREXTNT value in the dbname SQLDBGEN A file. &3 is the name of the database.

System Action: Message ARI0307I and message ARI0637D are displayed. If you do not re-edit the dbname SQLDBGEN A file, the current processing of database generation will fail.

User Response: If the number of DBEXTENTs defined by the dbname SQLFDEF file is correct, then re-edit the dbname SQLDBGEN A file and change the CUREXTNT value to the &1 value in this message.

If the number of DBEXTENTs defined by the dbname SQLFDEF file is not desired, then:
1. Enter N in response to message ARI0637D.
2. Enter CANCEL in response to message ARI0919D.
3. Rerun the SQLDBGEN EXEC.

© Copyright IBM Corp. 1987, 2000

4. Enter REPLACE in response to message ARI0632D.
5. Define the number of current DBEXTENTs that you want.

ARI0306E **The CUREXTNT value in** &1 **SQLDBGEN A is** &2**.** &3 **DBEXTENT statements are in** &1 **SQLDBGEN A. Please define a total of** &4 **DBEXTENTS in** &1 **SQLDBGEN A.**

Explanation: The SQLDBGEN EXEC displays this message when there is a mismatch between the CUREXTNT value and the number of DBEXTENT control statements in the dbname SQLDBGEN A file.

&1 is the name of the database.

&2 is the current number of DBEXTENTs specified by the CUREXTNT value in the dbname SQLDBGEN A file.

&3 is the number of DBEXTENT control statements in the dbname SQLDBGEN A file after the first END STATEMENT.

&4 is the CUREXTNT value in the dbname SQLDBGEN A file.

System Action: Message ARI0307I and message ARI0637D are displayed. If you do not re-edit the dbname SQLDBGEN A file, the current processing of database generation will fail.

User Response: Re-edit the dbname SQLDBGEN A file. You must make the number of DBEXTENT control statements equal the CUREXTNT value.

ARI0307I **You must modify the** &1 **SQLDBGEN A file for the database generation processing to be successful.**

Explanation: The SQLDBGEN EXEC has found an inconsistency in the dbname SQLDBGEN A file. A previous error message has explained the inconsistency.

&1 is the name of the database.

System Action: Message ARI0637D is displayed. If you do not re-edit the dbname SQLDBGEN A file, the current processing of database generation will fail.

User Response: Re-edit the dbname SQLDBGEN A file. Correct the file using the message displayed prior to ARI0307I as a guide for correcting the problem.

ARI0308E **Missing** &1 &2 &3 &4 &5 &6 &7 &8 &9 &10

Explanation: This message is displayed if an error is found during validation checking of the CUREXTNT keyword control statement in the dbname SQLDBGEN A file. &1 through &10 is explanatory text. SQLDBGEN displays the following examples:

ARI0308E MISSING CUREXTNT KEYWORD CONTROL STATEMENT

ARI0308E MISSING EQUAL SIGN AND VALUE IN CUREXTNT KEYWORD CONTROL STATEMENT

ARI0308E MISSING CUREXTNT VALUE IN CUREXTNT KEYWORD CONTROL STATEMENT

ARI0308E MISSING EQUAL SIGN IN CUREXTNT KEYWORD CONTROL STATEMENT

System Action: Message ARI0307I and message ARI0637D are displayed. The current processing of database generation will fail unless you correct the error.

User Response: Re-edit the dbname SQLDBGEN A file. Correct the file using the text of this message as a guide for correcting the problem.

ARI0309E **You entered a nonnumeric data type.**

Explanation: This message is displayed if an error is found during validation checking of message replies. &1 is the value you entered in response to a prompt from the SQLDBGEN or SQLADBEX EXECs.

System Action: The EXEC redisplays the prompting message issued prior to ARI0309E.

User Response: Respond again to the prompting message using a numeric data type.

ARI0310E **The** &1 &2 &3 **file contains the following unknown keywords:** &4 &5 &6 &7 &8 &9 &10**.**

Explanation: The SQLDBGEN EXEC displays this message during validation checking of the POOL and DBEXTENT control statements in the dbname SQLDBGEN A file. The error occurs if the statement is not a POOL control statement, a DBEXTENT control statement, or an END delimiter control statement.

&1 is the CMS file name.

&2 is the CMS file type of the file.

&3 is the CMS file mode.

&4 through &10 is the string of unknown keywords that were found.

System Action: Message ARI0313E follows message ARI0310E. Then messages ARI0307I and ARI0637D are displayed. The current processing of database generation will fail unless you correct the error.

User Response: Re-edit the SQLDBGEN A file using messages ARI0310E and ARI0313E as guides for correcting the problem. You have either mistyped a POOL or DBEXTENT control statement, or you failed to place the END delimiter control statement directly

after the last DBEXTENT control statement.

ARI0311I **You may need to modify the** &1 **SQLDBGEN A file. This EXEC updates nonrecoverable storage pool control statements only.**

Explanation: The SQLDBGEN EXEC displays this message if you decided to replace the dbname SQLFDEF file or the dbname SQLDBGEN file. If you redefined the DBEXTENTs in the database, for example, the changes will not be made automatically to the dbname SQLDBGEN A file, causing the database generation processing to fail. You will need to make the SQLDBGEN A file compatible with the redefinition of the database.

&1 is the dbname.

System Action: Processing continues. Message ARI0638D is displayed.

User Response: Edit the dbname SQLDBGEN A file. Ensure that the CUREXTNT value and DBEXTENT control statements match, and ensure that the number of current DBEXTENT definitions in the database match the CUREXTNT keyword control statement value in the dbname SQLDBGEN A file.

ARI0312E **The** statement-types **are out of sequence.**

Explanation: The SQLDBGEN EXEC displays this message during validation checking of the DBEXTENT control statements in the dbname SQLDBGEN A file. The extent numbers in the DBEXTENT control statements must start with number 1 and must be in consecutive, ascending order.

System Action: Message ARI0307I and message ARI0637D are displayed. The current processing of database generation will fail unless you correct the error.

User Response: Re-edit the dbname SQLDBGEN A file. Put the DBEXTENT control statements in sequence.

ARI0313E **The error occurred in the** &1 &2 &3 &4 &5 &6 &7**.**

Explanation: The SQLDBGEN EXEC displays this message after displaying message ARI0310E.

&1 through &7 tells where the error occurred in the SQLDBGEN A file during validation checking.

System Action: Message ARI0307I and message ARI0637D are displayed. The current processing of database generation will fail unless you correct the error where this message indicates.

User Response: Re-edit the dbname SQLDBGEN A file using this message to locate the error.

ARI0314E **No bootstrap modules could be found for the saved segment ID specified** &1 **on the SQLSTART EXEC.**

Explanation: The SQLSTART EXEC did not find any bootstrap module identified by &1 on the database production minidisk. The error could be caused by a saved segment ID that was not spelled correctly. The error could also occur if the SQLGENLD EXEC has not yet been run for that saved segment ID.

System Action: Processing ends.

User Response: Check the spelling of the saved segment ID parameter.

- If it is not correct, correct the spelling and rerun the EXEC.
- If it is correct, contact your system programmer to verify that the SQLGENLD EXEC has been run. After the SQLGENLD EXEC has been run, access the production minidisk again and then rerun the EXEC. If the error still occurs, follow your installation's problem determination procedure.

ARI0315E **No bootstrap modules could be found for the saved segment ID named** &1 **in the** &2 **SQLDBN file.**

Explanation: The SQLSTART EXEC did not find any bootstrap module identified by &1 on the database production minidisk. The &2 SQLDBN file indicates that an &1 bootstrap module did exist at one time, but no longer can be found.

System Action: Processing ends.

User Response: Contact your system programmer to obtain a valid saved segment ID name and rerun specifying that name in the saved segment ID parameter of the SQLSTART EXEC. Alternatively, your system programmer may require that the SQLGENLD EXEC be rerun to regenerate the appropriate bootstrap modules. If neither of the above two conditions apply, you may rerun the SQLSTART EXEC specifying SQLDBA as the saved segment ID name.

Note: This will require that the storage size for the database machine be defined large enough to include the database code. Likewise, ensure that the user machines have their storage size large enough to include the Resource Adapter code and ISQL code (if ISQL is being used).

ARI0316E **An error occurred in** &1 **while the database manager was attempting to query** &2**. Return Code =** &3**.**

Explanation: An error occurred while attempting to issue the CMS NUCEXT command to determine if the database ARIRTL1 or ARISRRM modules are CMS nucleus extensions. The return code (&3) from the CMS NUCEXT command is given. The meaning of the return

code is given in the *VM/ESA: CMS Command Reference* manual for your system.

System Action: Processing ends.

User Response: Follow your installation's problem determination procedure.

ARI0317E **SQLINIT did not establish application server access. Rerun the SQLINIT EXEC: SQLINIT DB (**server-name**).**

Explanation: VM machine access to a DB2 Server for VM application server was not established. Access to an application server had not been previously established by the user issuing SQLINIT DB (*server-name*).

System Action: The command is ended.

User Response: Rerun the SQLINIT EXEC, specifying the name of the application server to be accessed.

ARI0318D **Do you want** xxxxxxxx **to be the default saved segment ID for the user machines? Enter 0(No) or 1(Yes).**

Explanation: The SQLDCSID file is generated on the DB2 Server for VM Production minidisk. This file contains the default saved segment ID used by user machines that do not have a dbname SQLDBN file available for the database being accessed.

System Action: If you reply 1, processing continues and the SQLDCSID DEFAULT file is created. If you reply 0, the SQLDCSID DEFAULT file is not created and processing continues. If you do not reply, this message is displayed again.

User Response: Reply either 0(NO) or 1(YES).

ARI0319D **SQLDCSID default already exists. Do you want to replace it? Enter 0(No) or 1(Yes).**

Explanation: The SQLDCSID DEFAULT file that contains the default saved segment ID for user machines has previously been generated. To generate a new SQLDCSID file, reply 1. To keep the previously generated file, reply 0.

System Action: The SQLGENLD EXEC issues this message. If you reply 1, processing continues and the SQLDCSID DEFAULT file is replaced. If you reply 0, the SQLDCSID DEFAULT file is not replaced and processing continues. If your reply is neither 1 nor 0, this message is displayed again.

User Response: Reply either 0(NO) or 1(YES).

ARI0320I **The default server-name is** &1**.**

Explanation: The ARISRMBT MODULE (Resource Adapter bootstrap) on the user's A-disk is set up to access the named application server.

System Action: SQLDBID issues this message. If you have invoked SQLDBID directly, processing ends. If SQLDBID is invoked by another EXEC, control returns to that EXEC.

ARI0321E **Internal error in the ARISXMSG message EXEC. Return Code=**nn**.**

Explanation: The EXEC ARISXMSG failed with the following return codes:

12 Message not found.

14 Virtual storage capacity exceeded.

24 Invalid parameters selected.

System Action: Processing is terminated.

User Response: This is an internal error. Contact the designated support group for the installation.

ARI0322E **The ARISNLSC MACRO update has failed with a return code of** &1**. Processing terminated.**

Explanation: The EXEC was not able to update the ARISNLSC MACRO with an entry for the language being added.

System Action: The function is terminated.

System Programmer Response: Investigate the return code and take the required action. Ensure that the file layout for the ARISNLSC MACRO is valid. Refer to the *DB2 Server for VM System Administration* manual for the record layout expected in this file, and edit the file accordingly.

ARI0323I **It is recommended that a saved segment be created for this language's messages.**

Explanation: This is the expected response from the database manager.

System Programmer Response: Refer to the *DB2 Server for VM System Administration* for further information on how to create saved segments.

ARI0324E **An error occurred while the database manager was reading the** &1 **MACRO file. Return Code =** &2**.**

Explanation: An error occurred when the EXEC in process used EXECIO to extract the language and language identifier from the MACRO file specified in the error message.

System Action: The function is terminated.

System Programmer Response: Contact the designated support group for your installation.

ARI0325E **An error occurred when the database manager was attempting to copy the message repositories. Return Code =** &1**.**

Explanation: An error occurred when you tried to copy the message repository files. The return code from the CMS COPYFILE command is given in the message. Refer to the *Communications Reference* or the *VM/ESA: CMS Command Reference* manuals for more information.

System Action: The function is terminated.

User Response: Follow your installation's problem determination procedure.

ARI0326I **Now processing the CMS HELP text for** &1**.**

Explanation: This is the expected response from the database manager.

User Response: The following messages ask for the required information needed to complete the installation of CMS HELP for the language identified by language key &1.

ARI0327I **Only one language is installed and it is the default. This language cannot be deleted. Processing terminates.**

Explanation: Only the default language is currently installed on the database manager. Because at least one language must be available to generate system messages, this EXEC does not allow the last language to be deleted. Processing has been discontinued.

System Action: The function is terminated.

ARI0328I **Unable to activate DB2 Server for VM message repository. SET LANGUAGE command return code =** *&1***.**

Explanation: The EXEC has issued a CMS 'SET LANGUAGE LANGID (ADD ARI USER)' command and failed. Either the language identified by LANGID cannot be set or the ARI message repository cannot be activated. The return code from the SET LANGUAGE command is given in *&1*.

System Action: Processing continues.

System Programmer Response: Check the return code to determine the problem. If the ARI message repository cannot be activated, ensure that the message repository file ARIUME TEXT or ARIUME TXTLANGID can be accessed during run time of the EXEC. If necessary, link and access the ARI message repository file, issue the SET LANGUAGE command on the VM command line to activate the repository before running the EXEC.

ARI0329I **Cannot reset CMS language to** *&1***. SET LANGUAGE command return code =** *&2***.**

Explanation: The EXEC has issued a CMS 'SET LANGUAGE *&1*.' command and failed. The language identified by *&1*. cannot be set. The return code from the SET LANGUAGE command is given in *&2*.

System Action: Processing continues. The language set for the CMS session remains the one previously set.

System Programmer Response: Check the return code to determine the problem. If necessary, issue the SET LANGUAGE command on the VM command line to activate the required language for subsequent sessions.

ARI0330E **An error occurred while the database manager was attempting to copy** *filename-1 filetype-1 filemode-1* **to** *filename-1 filetype-1 filemode-2***. CMS COPYFILE command Return Code =** *return-code***.**

Explanation: An error occurred when copying the file with *filename-1* and *filetype-1* from *filemode-1* to *filemode-2*. The return code from the COPYFILE command was *return-code*.

System Action: Processing ends.

ARI0331E **Errors occurred during a DB2 Server for VM link edit. Refer to previous console messages or virtual printer messages for a description of the errors encountered.**

Explanation: One or more errors occurred during a database link edit.

System Action: The processing is terminated.

User Response: Check the messages displayed on the console and on the virtual printer to determine where the error occurred, and take the appropriate action.

ARI0332E **A problem occurred while performing the LISTFILE command on the product identifier file. The return code is** &1 **from the CMS LISTFILE command.**

Explanation: The CMS LISTFILE command failed during execution.

&1 is the return code from the LISTFILE command.

System Action: The processing is terminated.

System Programmer Response: Investigate the return code in the *VM/ESA: CMS Command Reference* manual and take the required action.

ARI0333I **DB2 Server for VM *&1*. service applied against language *&2*. Update all saved segments for the *&1*. component for this language according to instructions in the *DB2 Server for VM System Administration* manual.**

Explanation: This is the expected response from the database manager. Service has been applied against a component belonging to the language specified.

&1 is the component identifier that required the service.

&2 is the language identifier for the language serviced.

System Action: Processing continues.

System Programmer Response: Update the saved segments (saved segments) for the component of the language to which service was applied. Refer to the *DB2 Server for VM System Administration* manual for further instructions.

ARI0334I **No other DB2 Server for VM service needs to be applied.**

Explanation: This is the expected response from the database manager. Corrective service application has finished. All corrective service has been applied.

System Action: The processing continues.

ARI0337D **Do you want the input parameters to be a permanent change to SQLDBA bootstrap package? Enter 0(No) or 1(Yes).**

Explanation: You are given an opportunity to specify if a new saved segment is to become a permanent change to the SQLDBA bootstrap package.

System Action: The system waits for you to enter your response.

Programmer Response: Answer NO if you do not want the SQLDBA bootstrap package to be regenerated with this saved segment information. Answer YES if you want the SQLDBA bootstrap package to be permanently changed. Review the section on saved segments in the *DB2 Server for VM System Administration* manual for more information.

ARI0338I **Warning: The ARISSEGC MACRO *Q* file is missing, unavailable, or incorrectly formatted. The default SQLDBA bootstrap package will be generated to run the components in a free storage area.**

Explanation: The EXEC in process has not been able to use the default Saved Segment control file ARISSEGC MACRO *Q*.

Q is the filemode of the production disk or directory, usually accessed as filemode Q.

System Action: Processing continues.

Programmer Response: Determine the state of the Saved Segment control file. If necessary, recreate it by copying ARISSEGC MACRO V to the production (Q) disk and making any required changes. Once the file has been recreated, execute SQLBOOTS to regenerate the SQLDBA bootstrap package. Refer to the section on saved segments in the *DB2 Server for VM System Administration* manual for more information.

ARI0340D **The service process assumes that this procedure is being invoked from the MAINT userid, or a like virtual machine. To successfully complete service, write links to the database machine (normally SQLDBA) production and service minidisks will be required. If prompted, you should know the write link passwords to the production and service minidisks. If you are prepared to continue, depress the ENTER key, any other response will cause an exit.**

Explanation: This EXEC requires write access to the production and service minidisks of the database machine.

System Action: The system waits for your response.

User Response: If you know the write link passwords for the production and service minidisks, or if you know that the required write links exist, press ENTER. If not, press any other key to exit from processing. Rerun the service EXEC when you have the necessary information.

ARI0341I **Current &1-disk has been released.**

Explanation: The disk accessed as mode &1 has been properly released.

System Action: Processing continues.

ARI0342A **Enter the MACHID (user ID) of the virtual machine that owns the DB2 Server for VM service minidisk. The default is *&1*.**

Explanation: You are prompted to enter the id of the owner of the DB2 Server for VM service minidisk. The default machine id is &1.

System Action: The system waits for your response.

User Response: Enter the machine id for the owner of the DB2 Server for VM service minidisk.

ARI0343A **Enter the virtual address of the** &1 **minidisk owned by** &2**: The default is** &3**.**

Explanation: You are prompted to enter the address of the &1 (service or production) minidisk owned by the &2 id, usually SQLDBA. The default virtual address is given as &3.

System Action: The system waits for your response.

User Response: Enter the virtual address of the specified minidisk.

ARI0344D **DB2 Server for VM user id:** &1 **service MDISK:** &2 **production MDISK:** &3**.**

Are the above values correct? (yes | no | quit) null = yes.

Explanation: You are asked to confirm the values of the DB2 Server for VM owner user id as &1, service and production minidisks as &2 and &3 respectively.

System Action: The system waits for your response.

User Response: Enter YES if you wish to use these values and continue. Enter NO if these values are not correct, you will be asked to re-enter new values. Enter QUIT if you wish to abort processing. Processing will then stop.

ARI0345I **Linking to** &1 &2 **as virtual address** &3 **and accessing it as the** &4 **minidisk. If prompted, enter the write password for this minidisk.**

Explanation: The EXEC is about to link in the &2 (service or production) minidisk for the owner specified (&1) as virtual address &3. It will be accessed with file mode &4.

System Action: Processing continues.

User Response: Be prepared to enter the write password for the &2 minidisk.

ARI0348D **In order to perform a link edit the A disk must have 3200 1K blocks available. The current A-disk: LABEL CUU M STAT CYL TYPE BLKSIZE FILES BLKS USED-(%) BLKS LEFT BLK TOTAL** &1 &2 &3 &4 &5 &6 &7 &8 &9 &10 &11 &12 **Enter GO - to continue using this A disk CUU - to access CUU as the A disk. Note: CUU cannot be the CUU currently accessed as the C-disk. QUIT - to end.**

Explanation: The EXEC is asking the user to confirm that the required space is available on the A disk to do the link edit. The user has a choice of continuing with the present A disk, accessing a new disk as the A disk, or quitting the service process.

System Action: Waits for user's response.

User Response: If the current A disk has sufficient space, enter GO to continue with this A disk. If there is insufficient space on the current A disk, enter the cuu to access a different disk as the A disk. If you want to stop processing at this time, enter QUIT to exit.

ARI0349E exec-name **EXEC must be running from your C disk and your C disk must be accessed in write mode. Refer to the PTF cover letter and the 5648072 memo to users.**

Explanation: The *exec-name* EXEC is not being executed from the C disk, or the C disk is not in write mode.

System Action: Processing stops.

User Response: Rerun this EXEC from a C disk accessed in write mode.

ARI0350E **No tape drive is attached at virtual address 181. Mount the correct service tape volume and attach drive before re-invoking the service process.**

Explanation: The database distribution tape should be mounted at virtual address 181.

System Action: Processing stops.

User Response: Mount the database distribution tape and attach the drive as virtual address 181. Now re-invoke the service process.

ARI0351E **Error loading to** &1 &2**.**

Explanation: There has been an error while loading a file on the &1 &2 minidisk.

System Action: Processing stops.

User Response: Follow your installation's problem determination procedure.

ARI0352I **System link edit complete. DB2 Server for VM service complete.**

Explanation: This is the expected response from the database manager. The database service has finished.

ARI0353D **There is no A-disk currently accessed. Your choices are only 'CUU' and 'QUIT'.**

Explanation: There is no A-disk accessed and one is required.

System Action: Waits for user response.

User Response: Enter a valid cuu to access an A-disk or 'QUIT' to exit from processing.

ARI0354I **Original** &1**-disk re-accessed.**

Explanation: This is the expected response from the database manager. The &1 disk has been re-accessed successfully.

ARI0355I **Appropriate files are being copied from the service minidisk to the production minidisk.**

Explanation: This is the expected response from the database manager. This message is displayed prior to building the production minidisk from the files on the service minidisk.

ARI0357D **A-disk is now** &1**. LABEL CUU M STAT CYL TYPE BLKSIZE FILES BLKS USED-(%) BLKS LEFT BLK TOTAL**

&2 &3 &4 &5 &6 &7 &8 &9 &10 &11 &12 &13.
Enter GO, CHANGE or QUIT.

Explanation: The EXEC is asking the user to confirm the use of the current A disk (&1). &2 to &13 are the values that appear under the headings LABEL, CUU, M, and so on to BLK TOTAL.

System Action: Waits for user response.

User Response: Enter GO to use the current A-disk. Enter CHANGE to choose another A-disk. Enter QUIT to exit from processing.

ARI0359I **Loading of files to** &1 &2 **and** &3 **complete.**

Explanation: This is the expected response from the database manager. &1 is the A disk CUU, such as 191. &2 is the service disk CUU, such as 193. &3 is the production disk CUU, such as 195.

ARI0360W **Unable to access** &1**. Return Code =** &2**.**

Explanation: The chosen minidisk address to be accessed as the A-disk could not be accessed. This message is followed by message ARI0348D that asks you to enter a new minidisk address to be accessed.

System Action: The system displays message ARI0348D.

ARI0364E **Unable to acquire read access mode RR. User=**&1 **Device=**&2 **Return Code=**&3**.**

Explanation: The ARISPROD EXEC was unable to get READ access to the minidisk specified by &1 and &2.

System Action: The function is terminated.

Programmer Response: Verify that this user ID has write access to the production minidisk. The error could be caused by another user having gained WRITE access before this user ID. If this has occurred, detach the production minidisk from the other user ID and re-execute the function.

ARI0366I **Warning: The ARISAVES EXEC was unable to create the saved segment for the** &1 **component with ORIGIN** &2 **and SYSNAME** &3**. The database manager will generate this component in free storage. After installation successfully finishes, execute ARISAVES to create the default saved segment** &3**.**

Explanation: The ARISAVES EXEC is unable to complete successfully for the &1 component. The component is generated in free storage.

System Action: Processing continues.

Programmer Response: Investigate the return code from the ARISAVES EXEC after the installation has completed successfully and determine the cause of the problem. If you still want to have this component generated in a default saved segment, execute ARISAVES after installation is completed. Refer to the *DB2 Server for VM Program Directory* and the *DB2 Server for VM System Administration* manuals for more information.

ARI0370D **Was DB2 Server for VM installed in SFS? Enter** 0**(No), 1(Yes), or 111(Quit).**

Explanation: This message is issued by a service EXEC. The database manager production and service files have already been installed before running the service exec. The system prompts you to find out whether the database manager files were installed in CMS SFS directories or CMS minidisks.

System Action: The system waits for your response.

User Response: Type 1 (yes) if the database manager files were installed in CMS SFS directories. Type 0 (no) if the database manager files were installed in CMS minidisks. Type 111 (quit) if you want to exit from the exec program.

ARI0371D **Do you wish to install DB2 Server for VM system in SFS? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: The system prompts you to find out whether or not you want to install the DB2 Server for VM product on to SFS directories.

System Action: The system waits for your reply. If you type 1, the files are installed on to SFS directories. If you type 0, the files are installed on to minidisks.

User Response: Type 1 to install the files on to SFS directories. Type 0 to install the files on to minidisks.

ARI0372D **The following are the DB2 Server for VM system directory definitions:**

&5 **directory filepool = ***&1*
&5 **1st level directory = ***&2*
&5 **2nd level directory = ***&3*
&5 **directory = ***&4*

That is, your *&5* directory will be: *&1&2&3&4*
Do you want to use these values? Enter 0(No), 1(Yes), or 111(Quit).

Explanation: The system prompts you to verify that the directory full name *&1&2&3&4* is correct or not with: *&5* = service or production, *&1* = filepool name, *&2* = 1st level directory name, *&3* = 2nd level directory name, *&4* = service or production directory name

System Action: The system waits for your response. If you type 1 or Yes, the directory *&1&2&3&4* will be taken for *&5* directory. If you type `0` or No, you will be prompted to input the directory name again. If you type 111 or Quit, the program will exit at your request.

User Response: Enter `0` (No), 1 (YES), or 111 (Quit) to the prompt.

ARI0373A **Enter filepool (or a null for the existing value).**

Explanation: The system prompts you to enter the name of the Shared File System filepool. If you enter a null, the system will assign a default value.

System Action: The system waits for you to enter the name of the filepool or a null.

User Response: Enter the name of the filepool of the Shared File System directory you intend to use. Entering null will mean that the system will use a default filepool.

ARI0374A **Enter first directory (or a null for existing value).**

Explanation: The system prompts you to enter the name of the Shared File System first level directory. If you enter a null, the system will assign a default value.

System Action: The system waits for you to enter the name of the directory or a null.

User Response: Enter the name of the first level directory of the Shared File System you intend to use. Entering null will mean that the system will use a default value.

ARI0375A **Enter second directory (or a null for existing value).**

Explanation: The system prompts you to enter the name of the Shared File System second level directory. If you enter a null, the system will assign a default value.

System Action: The system waits for you to enter the name of the directory or a null.

User Response: Enter the name of the second level directory of the Shared File System you intend to use. Entering null will mean that the system will use a default value.

ARI0376I **The DB2 Server for VM CMS Shared File System directories are being created.**

Explanation: This message informs you that the Shared File System directories you identified are about to be loaded with database data.

System Action: The EXEC continues processing and starts loading the directories.

ARI0377E **An error occurred while accessing filepool** &1**.**

Explanation: An error occurred when the specified SFS filepool was accessed.

System Action: The installation EXEC terminates.

User Response: Check that the SFS filepool can be accessed in write mode. Rerun the database manager installation EXEC.

ARI0378E **An error occurred while accessing** &1 **directory as file-mode** &2**.**

Explanation: An error occurred when the specified SFS directory was accessed with a production or service filemode.

System Action: The installation EXEC terminates.

User Response: Check that the specified SFS directory can be accessed with the filemode displayed. Rerun the installation EXEC.

ARI0379I **Begin loading the DB2 Server for VM service directory.**

Explanation: The database manager service files are loading from the tape to the service directory that you specified.

System Action: Processing continues.

ARI0380I **The DB2 Server for VM service directory was loaded successfully.**

Explanation: This message informs you that the database Shared File System service directory was loaded successfully.

System Action: Processing continues.

ARI0381E **The DB2 Server for VM service directory load failed.**

Explanation: This message informs you that the Shared File System service directory was not loaded properly.

System Action: The EXEC terminates.

User Response: Analyze the CMS message that will accompany this error message. It will help you to determine whether data can be written to the identified Shared File System service directory. If data can not be written to this directory, contact your system programmer for assistance. Otherwise, you may re-run the EXEC to attempt the load again.

ARI0382I **Note: The DB2 Server for VM service directory file** &1 **MACRO has been updated to identify the** &2 **used for the DB2 Server for VM system installation.**

Explanation: The ARISPIDC EXEC identifies the names of the directories that are used in the database manager installation process.

System Action: This message is displayed when the MACRO has been updated.

ARI0383E **An error has occurred while copying the** &1 **MACRO file from the service directory to the production directory.**

Explanation: The ARISPIDC macro has been successfully copied from the service directory to the production directory.

System Action: The installation EXEC terminates.

User Response: Check that the service and production directories have the access modes that are necessary for the copy to proceed correctly and rerun the EXEC.

ARI0384I **The DB2 Server for VM system directories are loaded successfully. Continue with the installation process as described in the DB2 Server for VM Program Directory.**

Explanation: The specified DB2 Server for VM directories have been loaded correctly with the database code.

System Action: The downloading of the files on to the directories is complete.

User Response: Continue with the installation process as described in the *DB2 Server for VM Program Directory*

ARI0385I **The DB2 Server for VM system directories are not loaded successfully. Do not continue with the DB2 Server for VM installation process. Please correct the error identified by the previous messages and/or review the installation procedure. Then re-issue the command** &1**.**

Explanation: The specified DB2 Server for VM directories have not been loaded successfully.

System Action: The installation EXEC terminates.

User Response: Correct any errors identified by previous messages. Review the installation procedure. Rerun the installation EXEC from the beginning.

ARI0386I **Unable to access {***address* | *directory-name***} with file-mode** *file-mode*

Explanation: The specified minidisk with address *address* or the specified directory named *directory-name* cannot be accessed as file mode *file-mode*.

ARI0387I **Unable to access {***address* | *directory-name***} with file-mode** *file-mode*

Explanation: The specified minidisk with address *address* or the specified directory named *directory-name* cannot be accessed as file mode *file-mode*.

ARI0388A **Enter the CMS SFS directory in the format filepool:userid.dir.dir...**

Explanation: This requests you to enter the name of the SFS directory.

System Action: The system waits for your response.

User Response: Specify the full name of the SFS directory, in the form filepool:userid.dir.dir..., and so on.

ARI0389I **The directory specified is not valid.**

Explanation: You entered an incorrect directory name.

System Action: Processing ends.

User Response: Rerun the EXEC and specify the correct directory name.

ARI0390I **You cannot specify** *directory* **as filemode** *mode* **when SFS is not enabled.**

Explanation: You cannot specify a directory name and access it as filemode because your system does not have CMS shared file support (SFS) support.

ARI0392E **An error occurred while creating directory** *directory-name***.**

Explanation: The *directory-name* directory cannot be created.

System Action: Processing ends.

ARI0393E **An error occurred while accessing the root directory.**

Explanation: The root directory cannot be accessed by the program.

System Action: Processing ends.

ARI0394E **An error occurred while initializing filepool** *filepool-name***.**

Explanation: The file pool *filepool-name* either does not exist or is not available.

System Action: Processing ends.

ARI0395E **An error occurred while accessing directory** *directory-name***.**

Explanation: The *directory-name* directory cannot be accessed by the program.

System Action: Processing ends.

ARI0396D **Do you want to install the DB2 Server for VM HELP TEXT on a CMS Shared File System directory? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: You can specify that you want to put the DB2 Server for VM HELP TEXT in an SFS directory or in a minidisk, or that you want to exit from the program.

System Action: The system waits for your response. If you type 1 or yes, the DB2 Server for VM HELP TEXT will be installed in an SFS directory. If you type `0` or no, the DB2 Server for VM HELP TEXT will be installed in minidisks. If you type 111 or quit, the program will end.

User Response: Enter `0` (no), 1 (yes), or 111 (quit).

ARI0397D **Do you require a NETID parameter? Enter 0(No), 1(Yes), or 111(Quit). Note: This is a mandatory parameter if you plan to use distributed server support.**

Explanation: You are given an opportunity to specify the NETID of the system on which the application server resides. You need to specify the NETID if the application server will support the DRDA* protocol requests.

System Action: The system waits for user response. If the response is 1(Yes), then the system will display message ARI6134A next.

User Response: Reply either 0(No) or 1(Yes).

ARI0398E **NETID parameter not in correct format. NETID must be a string from 1 to 8 characters long. The string can contain uppercase letters (A-Z), numerics (0-9), and special characters $, @, # and the first character cannot be numeric.**

Explanation: You have entered an invalid value for NETID. A valid NETID must be a 1 -to 8-character string comprised of uppercase letters (A-Z), numerics (0-9), and special characters ($,#,@). The first character must not be numeric.

System Action: The system prompts for another user response.

User Response: Specify a valid NETID value.

ARI0399I **Errors/warnings occurred during** *exec-name* **EXEC processing.**

Explanation: There are errors or warnings from the system when running the *exec-name* EXEC.

Chapter 6. ARI0428–ARI0499 Resource Adapter Messages

ARI0428E **DB2 Server for VM services were requested but DB2 Server for VM is not available in Virtual Machine** &1**..**

Explanation: An application program requested a DB2 Server for VM service, but the database manager is not available in the virtual machine identified by &1.

System Action: The Resource Adapter returns to the application with an SQLCODE of -940 and the SQLWARN0 and SQLWARN6 flags set to S, indicating a serious error. If another SQL call is issued, the application will abend, and message ARI0439E will be issued.

Operator Response: Retry the application program after logging on the DB2 Server for VM virtual machine and running the SQLSTART EXEC.

ARI0429I **SQL CANCEL was requested while COMMIT or ROLLBACK was in process. CANCEL of COMMIT or ROLLBACK is not allowed. The database manager is ignoring the CANCEL request.**

Explanation: The terminal operator tried to cancel an SQL request while an SQL COMMIT or ROLLBACK command was processing. Cancellation of COMMIT or ROLLBACK processing is not allowed.

System Action: The SQL CANCEL request is ignored.

ARI0439E **The application is abending because of a serious system error that occurred on a call to the application server. SQLCODE =** &1A**..**

Explanation: The application tried to issue an SQL command after a serious error. (SQLWARN0 and SQLWARN6 were set to S.) The resource adapter prevents further tries to communicate with the database manager by terminating the application.

System Action: The resource adapter abends the application and rolls back the logical-unit-of-work.

Programmer Response: Correct the problem indicated by the SQLCODE value. (See the list of SQLCODEs starting at “Chapter 19. SQLCODEs” on page 277 for an explanation.) Notify the operator when the application can be tried again.

Operator Response: The cause of the problem may be due to your database environment. That is, the database manager is not available, or the database manager is accessing a different database than the one expected. If this is the case, correct the environment and try the application again. If you cannot determine the cause of the problem, gather pertinent information and notify the system programmer.

ARI0440E **The application is abending. An error or warning is encountered and the WHENEVER condition specifies STOP. SQLCODE = nnn.**

Explanation: The application had specified WHENEVER ERROR STOP or WHENEVER WARNING STOP. The SQLCODE satisfied this WHENEVER condition and the resource adapter stopped the application with an operating system abend.

System Action: The transaction ends.

Operator Response: The cause of the problem may be an environmental one. That is, the database manager is not available, or it is accessing a different database than the one expected. If this is the case, correct the environment and try the application again. If you cannot determine the cause of the problem, gather pertinent information and notify the application programmer.

Programmer Response: Correct the problem indicated by the SQLCODE. (See the list of SQLCODEs starting at “Chapter 19. SQLCODEs” on page 277 for an explanation.) Notify the operator when the application can be tried again.

ARI0442E **A non-DB2 Server for VM message has been received. The Resource Adapter has severed the communication path.**

Explanation: The resource adapter received a non-DB2 Server for VM message.

System Action: The resource adapter terminated the communication path.

Operator Response: If you are unable to determine the cause of the problem, retry the application program. If the problem repeats, collect pertinent information and notify the system programmer.

Programmer Response: Make a record of the error and contact the designated support group for your installation.

ARI0451I **Status of application server conversations on** timestamp**.**

Explanation: This is the normal DB2 Server for VM response to SQLQRY.

© Copyright IBM Corp. 1987, 2000

ARI0452I **EXTNAME =** external name.
RDBMS = dbname server version
STATUS = status
TIME = time
LU = luname
TPN = tpn
LUWID= luwid**.**

Explanation: This is the normal DB2 Server for VM response to SQLQRY.

ARI0470E **Trace file OPEN failed. Reason Code =** nn**. Return Code =** nn**.**

Explanation: The Resource Adapter trace is being initialized before application execution. An error occurred when the database manager attempted to open the trace output file on tape unit or CMS minidisk. The Reason Code value is the Primary Error Code. The Return Code value is the Secondary Error Code. For further explanation, look up the displayed error codes under either “Chapter 23. Sequential File Errors” on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Action: The trace facility remains inactive and the application continues to execute.

Operator Response: For corrective action, look up the displayed error codes under either “Chapter 23. Sequential File Errors” on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Programmer Response: See the Operator Response.

ARI0471I **Trace is ON.**

Explanation: The Resource Adapter trace parameters have been successfully processed. The trace is now active with trace output going to a tape unit or a CMS minidisk.

System Action:

- Trace is active. Resource Adapter trace is active as specified by the TRACERM, TRACECONV, TRACEDRRM initialization parameters to the SQLINIT EXEC.
- Trace output is directed to tape unit or CMS minidisk.

ARI0472E **Trace file CLOSE failed. Reason Code =** nn**. Return Code =** nn**.**

Explanation: The Resource Adapter Trace is being turned off during application termination. When the system attempted to close the trace output file on a tape unit or a CMS minidisk, a termination error occurred. The Reason Code value is the Primary Error Code. The Return Code value is the Secondary Error Code. For further explanation, look up the displayed error codes under either “Chapter 23. Sequential File Errors” on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Action: Trace shutdown is completed. DB2 Server for VM Resource Adapter trace is in the OFF state. Assume that the trace tape or minidisk file was not closed and is not fully processable (see below).

Operator Response: Report the problem (message) to your system programmer. Include any CMS minidisk or tape unit error messages displayed by CMS.

You can run the DB2 Server for VM Trace Formatter Utility Program (ARIMTRA). It will format and output all trace data up to the point of failure. At that point, it may end because of a lack of tape/minidisk CLOSE recording or an unreadable record on the tape/minidisk.

System Programmer Response: For corrective action, look up the displayed error codes under either “Chapter 23. Sequential File Errors” on page 407, or "SEQ FILE ERRORS" in the HELP text.

ARI0473I **Trace is OFF.**

Explanation: The Resource Adapter Trace is inactive and the trace output file is closed on a tape unit or a CMS minidisk.

System Action: The trace output file is closed.

Operator Response: None required.

ARI0474I **Warning: The status shared segment could not be updated.**

Explanation: The Resource Adapter was unable to update the status shared segment. The segment is either no longer available or does not have sufficient space for another entry.

System Action: The Resource Adapter continues without updating the status in the shared segment.

User Response: Make a record of the warning and contact the designated support group for your installation.

System Programmer Response: Check that the status shared segment is still loaded. If this message occurs intermittently, the status shared segment is not large enough to support your installation. Increase the space available for the segment.

ARI0499I **Warning: The status shared segment is not loaded. Return Code =** retcode**.**

Explanation: The Resource Adapter was unable to load the status shared segment.

System Action: The Resource Adapter continues without updating the status in the shared segment.

User Response: Make a record of the warning with the return code *retcode* and contact the designated

support group for your installation.

Operator Response: Check that the shared segment exists and can be loaded.

Chapter 7. ARI4021–ARI4175 DB2 Server for VM Control Messages

ARI4021E **The length{s} contained in a data stream sent by CRR recovery server at TPN** *tpn* **does not match the length of the data received. Length in Exchange Log Names:** *length1* **Length in Compare States:** *length2* **Length of data received:** *length3*

Explanation: One of the following has occurred:

- During *Resynchronization initialization* processing, the database manager received an Exchange Log Name's reply from the CRR recovery server. This reply contained a length that did not equal the length of the actual data received.
- During *Resynchronization recovery* processing, the database manager received an Exchange Log Names request and a Compare States request from the CRR recovery server. The sum of the lengths of these requests did not equal the length of the actual data received.

tpn indicates the transaction program name for the CRR recovery server with which the database manager is communicating.

length1 indicates the length, in decimal, found in the Exchange Log Names reply or request sent by the CRR recovery server.

length2 is only present in *Resynchronization recovery* processing and indicates the length found in the Compare States request sent by the CRR recovery server.

length3 indicates the length of data, in decimal, that was actually received.

System Action: The SYNCPNT parameter is reset from "Y" to "N" and the database manager is unable to use VM/CRR for handling DRDA sync point operations. As a result, this database may not be a participant in DRDA logical units of work that update multiple remote databases. However, the database manager is **still** able to process SQLDS protocol requests and DRDA requests that do not involve a sync point manager (for example, DRDA logical units of work that only update a single remote database).

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

ARI4022E **A data stream sent by CRR recovery server at TPN** *tpn* **has an invalid GDSID. GDSID expected:** *gdsid1* **GDSID received:** *gdsid2*

Explanation: One of the following has occurred:

- During *Resynchronization initialization* processing, the database manager received an Exchange Log Name's Reply from the CRR recovery server. This reply contained an invalid GSD identifier.
- During *Resynchronization recovery* processing, the database manager received an Exchange Log Name's Request and Compare States Request from the CRR recovery server. One of these requests contained an invalid GDS identifier.

gdsid1 indicates the GDS identifier that was expected by the database manager. This is a four digit hexadecimal value.

gdsid2 indicates the GDS identifier that was sent by the CRR recovery server. This is a four digit hexadecimal value.

System Action: The SYNCPNT parameter is reset from "Y" to "N" and the database manager is unable to use VM/CRR for handling DRDA sync point operations. As a result, this database may not be a participant in DRDA logical units of work that update multiple remote databases. However, the database manager is **still** able to process SQLDS protocol requests and DRDA requests that do not involve a sync point manager (for example, DRDA logical units of work that only update a single remote database).

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

ARI4023I **The CRR recovery server has completed resynchronizing work with the database manager. Date:** *date* **Time:** *time*

Explanation: Message ARI4023I is a signal to the database manager operator that the CRR recovery server has completed *resynchronization recovery* with the database manager at the date and time indicated. This message is followed by message ARI0182I which identifies the *transaction id* and *LUW id* for the unit of work being recovered.

System Action: Database manager processing continues.

Database Administrator Action: If errors occurred during *resynchronization recovery*, then an error message

© Copyright IBM Corp. 1987, 2000

will have been displayed that indicated the error. See the explanation for that message and resolve the error.

ARI4100I **TCP/IP service initialized from** *{ETC SERVICES | TCPPORT}***. Service name is** *service-name***. TCP/IP Port is** *port-number***.**

Explanation: The TCP/IP service was successfully initialized by the application server.

TCP/IP service can be initialized from the ETC SERVICES file or TCPPORT. If it is ETC SERVICES, then the service name and port number were determined by doing a look up in the ETC SERVICES file. If it is TCPPORT, then the port number was taken from the TCPPORT parameter specified on the SQLSTART command.

service-name is the service name that was used to do the look up in the ETC SERVICES file. If *service-name* is N/A, then there is no service name. This will only happen if TCP/IP support was initialized by specifying the TCPPORT parameter.

port-number is the port number that the application server will listen to for connect requests. It is determined by the value in the ETC SERVICES file or from the TCPPORT parameter.

System Action: The application server listens for TCP/IP connect requests on the port identified in *port-number*. The application server is able to handle both SNA connections and TCP/IP connections.

ARI4101I **Unable to find** *service-name* **in ETC SERVICES.**

Explanation: The application server attempted to search the ETC SERVICES file for the value identified by *service-name*.

System Action: The system continues. If *service-name* is the same value as the RESID, the application server will attempt to use the default service name of ddm-rdb. If *service-name* is ddm-rdb, the application server will discontinue TCP/IP initialization. The application server will only be able to accept SNA connections.

Operator Response: Report this message to your system programmer.

System Programmer Response: Determine if an entry for *service-name* should be added to the ETC SERVICES file. Add the entry if it is required. If TCP/IP services are not required, ignore this message or start the application server with the TCPPORT=0 parameter to bypass the TCP/IP initialization.

ARI4102I **Unable to initialize TCP/IP service.**

Explanation: The application was unable to initialize TCP/IP services because access to TCP/IP for VM was not available.

System Action: The application server continues, but it can only accept SNA connections.

Operator Response: Report this error to your system programmer.

System Programmer Response: If TCP/IP services are required, make sure that TCP/IP for VM is available and running, and the ETC SERVICES file or the TCPPORT parameter is correct. Shut down and restart the application server when TCP/IP for VM is available. If TCP/IP services are not required, ignore this message or start the application server with the TCPPORT=0 parameter to bypass the TCP/IP initialization.

ARI4103I **TCP/IP service** *service-name* **failed. Return code =** *n***.**

Explanation: A TCP/IP function was performed but an error occurred during the execution of the function. *service-name* identifies the function and *n* is the return code that was received.

System Action: The system continues.

Operator Response: Report this error to your system programmer.

System Programmer Response: Take corrective action, if necessary, based on the function and return code received.

The description of the function reported by this message can be found in the *IBM Transmission Control Protocol/Internet Protocol for VM: Programmer's Reference Version 2 Release 4* manual.

The explanation for this return code can be found in the *IBM Transmission Control Protocol/Internet Protocol for VM: Programmer's Reference Version 2 Release 4* manual.

ARI4106I **The connection with the TCP/IP machine has been severed.**

TCP/IP service = *servicename***.**
Sever Reason = *reasoncode***.**

Explanation: The IUCV connection between the application requester and the TCP/IP VM machine has been severed.

System Action: The logical unit of work is rolled back.

Operator Response: Report this error to your system programmer.

System Programmer Response: Report this error to the designated support group for your installation.

ARI4107I **Error while communicating with the TCP/IP machine.**

TCP/IP service = *servicename*.
IPRCODE = *reasoncode*.

Explanation: A communications error occurred between the application requester and the TCP/IP VM machine.

System Action: The communications path is severed and the logical unit of work is rolled back.

Operator Response: Report this error to your system programmer.

System Programmer Response: Report this error to the designated support group for your installation.

ARI4108E **Unable to initialize the external security manager. SECTYPE has been changed to DB2.**

Explanation: The application server issued the external security manager initialization command "RPIUCMS INIT" and it failed. The command was issued because the initialization parameter SECTYPE=ESM was specifed.

System Action: SECTYPE is switched to DB2 and processing continues.

System Programmer Response: Correct the problem with the external security manager.

ARI4109E **While trying to process an external security manager request through the RACROUTE macro, an error occurred.**

Function = *function*
SAF Return Code = X'*return code***'**
RACF Return Code = X'*return code***'**
RACF Reason Code = X'*reason code***'**

Explanation: The application server issued the RACROUTE macro to authenticate a userid. The RACROUTE macro responded with the information described in the message.

System Action: Authentication of the userid failed. SECTYPE remains at ESM and processing continues.

System Programmer Response: Correct the problem with the external security manager. For more information about the return codes and reason codes, see *External Security Interface (RACROUTE) Macro Reference for MVS and VM* manual.

ARI4110E **ESM function** *ESM function* **failed. Return code =** *return code*

Explanation: The application server was performing function *ESM function* and encountered an error. *return code* indicates the error that occurred.

System Action: Processing continues.

System Programmer Response: Report this error to the designated support group for your installation.

ARI4112I **TCPPORT** *nnnnn* **is invalid. It must be in the range from** *n1* **to** *n2*.

Explanation: An invalid port number was specified in the START TCPIP TCPPORT operator command, *nnnnn* is the port number specified.

n1 is the lower limit (or minimum) for the correct parameter value.

n2 is the upper limit (or maximum) for the correct parameter value.

System Action: The system continues, the application server ignores the restart TCP/IP request. The application server will only be able to accept SNA connections.

Operator Response: Retry the command by specifying a valid port number.

ARI4113I **TCP/IP support for the database manager was never enabled. Start request is ignored.**

Explanation: The START TCPIP operator command was issued while the TCP/IP support initialization process was never invoked during application server startup. The restart process is relying on the TCP/IP agent structure previously created during application server initialization. Without the agent structure, restart process cannot be performed. This is the case where either TCPPORT = 0 was specified or RMTUSERS = 0 was specified.

System Action: The system continues, the application server ignores the restart TCP/IP request. The application server will only be able to accept SNA connections.

Operator Response: Report this error to your system programmer.

System Programmer Response: Restart the application server by specifying a valid TCPPORT parameter if TCP/IP support is desired.

ARI4114I **TCP/IP support for the database manager is still active. Start request is ignored.**

Explanation: The START TCPIP operator command was issued, but the TCP/IP support for the application server is still active.

System Action: The system continues, the application server ignores the restart TCP/IP request.

Operator Response: Issue the START TCPIP command only after TCP/IP support for the application server went down abnormally.

ARI4115I TCP/IP support for the database server is disabled.

Explanation: TCP/IP support for the application server is disabled. This could be the result of a severe error detected during the execution of a TCP/IP function. The listener socket of the application server is closed and no new TCP/IP connect request can be accepted. The application server will only be able to accept SNA connections.

For the error situation encountered, there should be a preceding message that describes the error. Refer to the corresponding message and take action accordingly.

System Action: The system continues, but no TCP/IP support is available.

Operator Response: Report this error to your system programmer.

System Programmer Response: Take correction action, if necessary, based on the function and return code received in the preceding error message. Once the error condition is resolved, issue the START TCPIP operator command to restart TCP/IP support if TCP/IP communication for the application server is desired.

ARI4151I FORMAT 1
START PROC successful for *proc-name.*
FORMAT 2
START PROC successful for *proc-name* **for AUTHID** *authid.*

Explanation: The stored procedure referenced has been started successfully.

System Action: Processing of the operator command has completed.

ARI4152E FORMAT 1
START PROC failed for *proc-name***.**
Reason Code *reason-code.*
FORMAT 2
START PROC failed for *proc-name* **for AUTHID** *authid***.**
Reason Code *reason-code.*

Explanation: Processing of the command failed. Possible reason codes are:

1 The procedure is not found.

2 A wildcard is used improperly in the procedure name.

3 A wildcard is found in the *authid*.

4 The procedure name specified is a keyword.

5 The *authid* specified is a keyword.

System Action: The operator command is not processed.

Operator Response: Correct the error and issue the command again.

ARI4153I FORMAT 1
Stored procedure *proc-name* **is already started.**
FORMAT 2
Stored procedure *proc-name* **for AUTHID** *authid* **is already started.**

Explanation: The stored procedure referenced was started by a previous START PROC command.

System Action: Processing of the operator command has ended.

ARI4154I FORMAT 1
Stored procedure *proc-name* **is not found.**
FORMAT 2
Stored procedure *proc-name* **for AUTHID** *authid* **is not found.**
FORMAT 3
Wildcard is used improperly in stored procedure name *proc-name***.**
FORMAT 4
Wildcard is used improperly in AUTHID *authid* **for stored procedure** *proc-name***.**
FORMAT 5
Stored procedure name *proc-name* **is a keyword.**
FORMAT 6
AUTHID *authid* **for stored procedure name** *proc-name* **is a keyword.**
FORMAT 7
ACTION *action* **specified is invalid. Only QUEUE and REJECT are valid.**

Explanation: This message provides information about an error that occurred when an operator command was issued. It indicates that either

1. The procedure specified was not found in the cached information from SYSTEM.SYSROUTINES
2. Invalid information was entered.

System Action: Processing of the operator command has ended.

Operator Response: Correct the source of the error and issue the command again.

ARI4155I **FORMAT 1**
STOP PROC successful for *proc-name.*
FORMAT 2
STOP PROC successful for *proc-name* **for AUTHID** *authid***.**

Explanation: The stored procedure referenced has been stopped successfully.

System Action: Processing of the operator command has completed.

ARI4156I **FORMAT 1**
New calls to stored procedure *proc-name* **will be** *action-code.*
FORMAT 2
New calls to stored procedure *proc-name* **with AUTHID** *authid* **will be** *action-code.*

Explanation: This message is displayed when a stored procedure is stopped. It indicates how subsequent CALLs to that procedure will be handled.

proc-name
The name of the stored procedure

authid The authorization ID associated with the stored procedure

action-code
Either QUEUED or REJECTED

System Action: The STOP PROC command has completed successfully for the procedure indicated.

ARI4157E **FORMAT 1**
STOP PROC failed for *proc-name.*
Reason code *reason-code.*
FORMAT 2
STOP PROC failed for *proc-name* **for AUTHID** *authid.*
Reason code *reason-code.*

Explanation: Processing of the command failed. Possible reason codes are:

1 The procedure is not found.

2 A wildcard is used improperly in the procedure name.

3 A wildcard is found in *authid*.

4 The procedure name specified is a keyword.

5 The *authid* specified is a keyword.

6 *Action* specified is invalid. Only QUEUE and REJECT are valid.

System Action: Processing of the operator command has ended.

Operator Response: Correct the error and issue the command again.

ARI4158E **Error in operator command** *oper-cmd***. Reason code** *reason-code.*

Explanation: Processing of the command failed. Possible reason codes are:

1 The stored procedure is not found.

2 A wildcard is used improperly in the procedure name.

3 A wildcard is found in *authid*.

4 The procedure name specified is a keyword.

5 The *authid* specified is a keyword.

6 Stored procedure server is not defined.

7 Stored procedure server group is not defined.

8 A wildcard is used improperly in the stored procedure server name.

9 A wildcard is used improperly in the stored procedure server group name.

10 The stored procedure server name specified is a keyword.

11 The stored procedure server group name specified is a keyword.

oper-cmd The operator command issued.

System Action: Processing of the operator command has ended.

Operator Response: Correct the error and issue the command again.

ARI4159I **FORMAT 1**
Stored procedure *proc-name* **is already stopped.**
FORMAT 2
Stored procedure *proc-name* **for AUTHID** *authid*
is already stopped.

Explanation: The stored procedure referenced was stopped by a previous STOP PROC command.

System Action: Processing of the operator command continues.

ARI4160E **Operator command** *oper-cmd* **has insufficient number of input parameters.**

Explanation: The command entered did not use the correct syntax.

oper-cmd The operator command issued.

Operator Response: Refer to the *DB2 Server for VSE &*

VM Operation manual for the syntax diagram for the command.

ARI4161E **Operator command** *oper-cmd* **has too many input parameters.**

Explanation: The command entered did not use the correct syntax.

oper-cmd The operator command issued.

System Action: Processing of the operator command has terminated.

Operator Response: Refer to the *DB2 Server for VSE & VM Operation* manual for the syntax diagram for the command.

ARI4162I **Stored procedure server** *serv-name* **has been started.**

Explanation: The stored procedure server *serv-name* is able to accept SQL CALL statements to run stored procedures associated with this server.

System Action: Operator command complete successfully. Processing continues.

ARI4163I **FORMAT 1**
Stored procedure server
serv-name **is not defined.**
FORMAT 2
Stored procedure server group
serv-group-name **is not defined.**
FORMAT 3
Wildcard is used improperly in stored procedure server name *serv-name***.**
FORMAT 4
Wildcard is used improperly in stored procedure server group name
*serv-group-name***.**
FORMAT 5
Stored procedure server name
serv-name **is a keyword.**
FORMAT 6
Stored procedure server group name
serv-group-name **is a keyword.**

Explanation: This message provides information about an error that occurred when an operator command was issued. It indicates that either

1. The stored procedure server specified was not found in the cached information from SYSTEM.SYSPSERVERS
2. Invalid information was entered.

System Action: Processing of the operator command terminates.

Operator Response: Correct the error and issue the command again.

ARI4164I **Stored procedure server** *serv-name* **is already started.**

Explanation: The stored procedure server *serv-name* was started by a previous START PSERVER command.

System Action: Processing of the operator command has ended.

ARI4165I **START PSERVER cannot be processed for** *server-name* **because it has a status of** *status***.**

Explanation: Only a stored procedure server that is currently stopped can be started. Possible values for *status* are:

- STARTING
- STOPPING

System Action: Processing of the operator command terminates.

Operator Response:
- If the current status is STARTING, no action is necessary.
- If the current status is STOPPING, the stored procedure server is waiting for a stored procedure to complete. When the stored procedure finishes, the stored procedure server will be terminate, and then the START PSERVER command can be issued for that stored procedure server.

  If you need to stop and start the stored procedure immediately, you can use the STOP PSERVER command with the QUICK option, and then issue the START PSERVER command.

ARI4166I **FORMAT 1**
Unable to acquire storage for a pseudoagent for server *server-name***.**
FORMAT 2
Unable to acquire storage for an XPCC control block for server
*server-name***.**

Explanation: In VM, when START PSERVER is issued, the database manager allocates the pseudoagent structure to be used for the connection between the database manager and the stored procedure server. In VSE, when START PSERVER is issued, the database manager allocates an XPCC control block to be used for the connection between the database manager and the stored procedure server. In either case, if storage for the structure is not available, the START PSERVER command cannot complete.

System Action: Processing of the operator command terminates.

Operator Response: Try the command again later. Since storage utilization on the database manager varies depending on the number of active agents and

their activities, storage may become available, enabling the command to complete.

Database Administrator Action:

Consider whether the amount of storage available to the database manager needs to be increased.

ARI4167E **FORMAT 1**
START PSERVER for *serv-name*
failed.
Reason code *reason-code.*
FORMAT 2
START PSERVER GROUP for
serv-group-name **failed.**
Reason code *reason-code.*

Explanation: Processing of the command failed. Possible reason codes are:

1 Stored procedure server *serv-name* is not defined.

2 Stored procedure server group *serv-group-name* is not defined.

3 The database manager is not able to acquire a pseudoagent for the stored procedure server.

4 The database manager is not able to allocate an XPCC control block for the stored procedure server.

5 START PSERVER cannot be issued for a stored procedure server that has a status of STARTING.

6 START PSERVER cannot be issued for a stored procedure server that has a status of STOPPING.

7 Wildcard is used improperly in stored procedure server name *serv-name*.

8 Wildcard is used improperly in stored procedure server group name *serv-group-name*.

9 Stored procedure server name *serv-name* specified is a keyword.

10 Stored procedure server group name *serv-group-name* specified is a keyword.

System Action: Processing of the operator command has ended.

Operator Response: This message is preceded by an informational message that indicates the reason for the error. Follow the instructions given in the Operator Response for the informational message.

ARI4168I **START PSERVER operator command timed out.**

Explanation: The START PSERVER command did not complete in the amount of time specified by the initialization parameter PTIMEOUT. Either:

- the PTIMEOUT value is too short
- the database manager was not able to connect to the stored procedure server

System Action: Processing of the operator command terminates.

System Programmer Response: If the connection failed, use the diagnostic information displayed to determine the source of the problem and correct the error.

Database Administrator Action:

If no error occurred during the attempt to connect to the stored procedure server, consider whether the PTIMEOUT value should be increased.

ARI4169E **FORMAT 1**
Stored Procedure *proc-name* **has exceeded the maximum number of abends allowed. It has been stopped successfully. New calls will be rejected.**
FORMAT 2
Stored Procedure *proc-name* **for AUTHID** *authid* **has exceeded the maximum number of abends allowed. It has been stopped successfully. New calls will be rejected.**

Explanation: A stored procedure has exceeded the number of allowed abnormal terminations as defined by PROCMXAB init parameter.

System Action: Processing of the SQL statement has ended.

Operator Response: Issue START PROC command to restart the procedure.

Programmer Response: Determine the reason for stored procedure abend.

ARI4170E **FORMAT 1**
Cannot *action* **stored procedure** *proc-name* **The cache structure is currently being updated by another agent.**
FORMAT 2
Cannot *action* **stored procedure** *proc-name* **for AUTHID** *authid*
The cache structure is currently being updated by another agent.
FORMAT 3
Cannot *action* **stored procedure server** *serv-name*
The cache structure is currently being updated by another agent.

Explanation: The cache structure is currently being updated by another agent.

System Action: Processing of the operator command has ended.

Operator Response: Issue the command again at a later time

ARI4171I **FORMAT 1**
Stored procedure server *serv-name*
has been stopped.
It can be implicitly started.
FORMAT 2
Stored procedure server *serv-name*
has been stopped. It cannot be implicitly started.

Explanation: Stored procedure server *serv-name* has been deactivated as requested by a STOP PSERVER or SQLEND command.

System Action: Processing of the operator command is complete.

ARI4172I **Stored procedure server** *serv-name* **is already stopped.**

Explanation: The stored procedure server *serv-name* was stopped by a previous STOP PSERVER command.

System Action: Processing of the operator command terminates.

ARI4173I **FORMAT 1**
Stored procedure server *serv-name* **is waiting for stored procedure** *proc-name* **to finish.**
FORMAT 2
Stored procedure server *serv-name* **is waiting for stored procedure** *proc-name* **with AUTHID** *authid* **to finish.**

Explanation: The STOP PSERVER command was issued for a stored procedure server that is currently running a stored procedure.

System Action: When the stored procedure finishes, the database manager will stop the stored procedure server.

Operator Response: If you must stop the stored procedure server immediately, issue the STOP PSERVER command again, specifying the QUICK option.

ARI4174E **FORMAT 1**
STOP PSERVER for *serv-name* **failed.**
Reason code *reason-code.*
FORMAT 2
STOP PSERVER GROUP for *serv-group-name* **failed.**
Reason code *reason-code.*

Explanation: Processing of the command failed. Possible reason codes are:

1 The definition of the stored procedure server *serv-name* is not found.

2 The definition of the stored procedure server group *serv-group-name* is not found.

3 A wildcard is used improperly in stored procedure server name *serv-name*.

4 A wildcard is used improperly in stored procedure server group name *serv-group-name*.

5 The stored procedure server name *serv-name* specified is a keyword.

6 The stored procedure server group name *serv-group-name* specified is a keyword.

System Action: Processing of the operator command has ended.

Operator Response: Correct the error and issue the command again.

ARI4175I **The** *cach-name* **cache is loaded at** *load-address***.**

Explanation: *cach-name* is the name of the cache loaded by the database manager.

load-address is the storage address (expressed in hexadecimal) where the cache was loaded.

System Action: Processing continues.

System Programmer Response: You may want to note the address for use in any potential problem determination.

Chapter 8. ARI0500–ARI0513 SQL Messages

The SQL messages are associated with the SQLCODEs. These messages provide general information; they do not give detailed descriptions of the cause of the error. The SQLCODE appears in message ARI0505I. The text following messages ARI0501 and ARI0503 is the SQLCODE descriptive text. To find the explanation and related action, refer to "Chapter 19. SQLCODEs" on page 277 of this manual.

ARI0500I SQL processing was successful.

Explanation: The processing of the SQL statement was successful. This message occurs only when SQLCODE = 0 and no SQL warning conditions occurred during SQL statement processing. This message is followed by message ARI0505I, which indicates the SQLCODE and ROWCOUNT information.

System Action: Normal completion.

ARI0501I An SQL warning has occurred.
SQLCODE descriptive text

Explanation: This message occurs when either:

- The SQLCODE is greater than zero, or
- The SQLCODE is zero and SQL warning conditions occurred during SQL statement processing, or
- The SQLCODE is negative, but the preprocessor is treating it as a warning at preprocessing time.

This message is followed by message ARI0505I, which provides the SQLCODE and ROWCOUNT information, and in the first two conditions, the ARI0502I message that indicates the specific warning conditions.

The SQLCODE descriptive text describes the SQLCODE error condition.

System Action: SQL statement processing completed successfully.

User Response: For an SQLCODE greater than zero, take action based upon the SQLCODE descriptive text. For the SQLCODE of zero, take action defined for message ARI0502I. For an SQL code of less than zero, correct the problem described in the SQLCODE descriptive text before the application program is executed.

ARI0502I Following SQL warning conditions encountered: {truncation | NULLIGNORED | mismatch | NULLWHERE | performance | implicit rollback | adjustment | ineligible | no storage | can't fit 2}.

Explanation: Each of the SQL processing warning condition keywords is directly associated with an SQLCA SQLWARN indicator. The applicable SQLWARN indicator is identified in the explanation of each keyword.

TRUNCATION (SQLWARN1) - one or more of the requested data items was truncated due to insufficient space in the declared program variable. By examining the null indicator variables of the data items returned, you can find which data items were truncated. A positive value in the null indicator denotes actual length before truncation.

Set to W, if the value of a string column was truncated when it was assigned to a host variable.

Set to Z, if mixed character data was truncated and the resulting data does not follow the proper rules regarding mixed data.

Set to Z, if both W and Z truncations occur.

NULLIGNORED (SQLWARN2) - set to W, if null values were ignored in the computation of a column function such as AVG, SUM, MAX, MIN, or COUNT.

MISMATCH (SQLWARN3) - set to W, if the number of target variables in the INTO clause is less than the number of items in the SELECT list.

If the number of target variables in the INTO clause is greater than the number of items in the SELECT list, then the warning flag will not be set.

NULLWHERE (SQLWARN4) - set to W, if an UPDATE or DELETE command has been used without a WHERE clause. The user should verify that the update or deletion was intended unconditionally on the entire table.

PERFORMANCE (SQLWARN5) - set to W, if a WHERE clause, represented internally by one or more search arguments, associated with a SELECT statement has exceeded a DB2 Server for VM internal limitation. This means that a performance degradation will result because the database manager will not internally convert eligible predicates to search arguments. The database manager may still choose to use indexes for eligible predicates, but if not, degradation may further be increased since a DBSPACE scan would be used to retrieve query data. Decreasing the number of predicates, such as by removing unnecessary conditions

© Copyright IBM Corp. 1987, 2000

which may exist in the WHERE clause of the SELECT statement, may alleviate this condition.

IMPLICIT ROLLBACK (SQLWARN6) - the logical unit of work is no longer active. It has been rolled back due to a system failure.

Set to W, when the database manager is forced to terminate a logical unit of work.

Set to S, when the database manager issues a severe SQLCODE.

ADJUSTMENT (SQLWARN7) - set to W, when an adjustment has been made to a derived date or timestamp value for the last day of the month.

Set to Z, when the divisor or dividend of a DECIMAL divide operation requires readjustment in order to complete the divide operation. This may affect the accuracy of the quotient (result).

INELIGIBLE (SQLWARN8) - set to W, if the statement is disqualified for blocking for reasons other than storage, such as use of long fields.

Set to Z, if a character that could not be converted was replaced with a substitute character.

NO STORAGE (SQLWARN9) - set to W, if there was insufficient storage for blocking and therefore blocking is cancelled for the subject cursor.

CAN'T FIT 2 (SQLWARNA) - set to W, if blocking factor of at least two rows cannot be maintained and therefore blocking is cancelled for the subject cursor.

System Action: Normal completion; SQL statement processed successfully.

User Response: Take appropriate action based upon warning condition.

ARI0503E **An SQL error has occurred.** *SQLCODE descriptive text*

Explanation: Processing of the SQL statement was not successful. This message is followed by message ARI0505I, which provides the SQLCODE and ROWCOUNT information, and message ARI0504I, which indicates the condition that caused the error message.

SQLCODE descriptive text is derived from SQLCODE descriptive text modules. Multiple lines of SQLCODE descriptive text may appear.

System Action: SQL statement was not processed.

User Response: Take action based on the SQLCODE descriptive text. For additional information on the specific problem, refer to the appropriate SQLCODE in the section "Chapter 19. SQLCODEs" on page 277, or use the SQLCODE number as the topic for the ISQL HELP command (for example: enter HELP -105).

ARI0504I **SQLERRP:** *pppppppp* **SQLERRD1:** *nnn* **SQLERRD2:** *nnn*.

Explanation:

SQLERRP RDS module name, preprocessor module name, Resource Adapter module name, or RUN-TIME that set either the warning flag or the SQLCODE.

RUN-TIME means that an error occurred while a DB2 Server for VM data manipulation statement was being executed.

If an error occurs while the statement is being preprocessed, the RDS module name is displayed. If the error occurs in the run-time access generator (when the statement is being executed), the message text may contain either the RDS module name or the literal string RUN-TIME. The RUN-TIME message is issued when the error is one that cannot be trapped by the access generator run-time modules and is handled by the DB2 Server for VM abend handler. For example, division by zero or an arithmetic overflow.

SQLERRD1 Secondary return code value set by module *pppppppp*. This value, along with the module name, can be used by persons responsible for locating problems within the database manager itself. This information is not usually of interest to others. However, if the SQLCODE is -314, the SQLERRD1 value denotes the reason why a host variable was incorrectly declared. Refer to SQLCODE -314 for more information.

SQLERRD2 Return code value set by DBSS or the Resource Adapter. The definition of the Resource Adapter return codes can be found under "Chapter 21. DB2 Server for VM Reason Codes" on page 399. See "Chapter 18. DBSS Return Codes" on page 271 for definitions of the DBSS return codes.

This message follows messages ARI0503E (or ARI0533E) and ARI0505I.

System Action: See System Action for the associated ARI0503E (or ARI0533E) message.

User Response: See User Response for the associated ARI0503E (or ARI0533E) message.

ARI0505I **SQLCODE = *nnn* [SQLSTATE = *nnnnn*] ROWCOUNT = *nnn***

Explanation: This message follows messages ARI0500I, ARI0501I, and ARI0503E (or ARI0533E). In the case of DELETE statements, it only appears when the number of dependent rows affected is greater than zero.

SQLCODE= nnn:

nnn < 0 SQL statement processing has not been completed because an error occurred.

nnn = 0 SQL statement processing completed. Warning may or may not indicate an error.

nnn > 0 This indicates a warning. SQL statement processing completed but one or more unusual conditions occurred.

The text following message ARI0501I or ARI0503E is the SQLCODE descriptive text. To find the explanation and related action, either refer to the section "Chapter 19. SQLCODEs" on page 277 in this manual or enter a HELP command for the SQLCODE indicated.

SQLSTATE=nnnnn, where nnnnn is a 5-digit number associated with an SQLCODE. See "Chapter 20. SQLSTATEs" on page 377 for a list of SQLSTATE values.

Note: The application server does not issue an SQLSTATE value if the error occurs during DB2 Server for VM preprocessing.

ROWCOUNT = nnn, where nnn is a value that indicates:

UPDATE the number of rows updated.

INSERT the number of rows inserted.

DELETE the number of rows deleted.

For any other SQL statement or catalog migration, the value of ROWCOUNT is not applicable.

System Action: See System Action for the associated SQLCODE.

User Response: See User Response for the associated SQLCODE.

ARI0506I **Dependent rows affected = *n.***

Explanation: This message appears only for DELETE statements when the number of dependent rows affected is greater than zero. The command caused changes to dependent tables because of referential constraints with a rule of SET NULL or CASCADE. The number (n) of dependent rows affected is reported.

System Action: Processing of the command continues.

ARI0511E **SQL message formatter error. Reason Code = *nn*.**

Explanation: The SQL Message Formatter Routine has encountered incorrect parameter values being passed to it by some system internal calling routine.

REASON CODE = nn:

nn = 10
The MSGID SQLTEXT flag is set ON but the SQLMSGNO field does not contain either of the valid values, 501 or 503.

nn = 20
The MSGID SQLTEXT flag is set OFF with one or more of the MSGID fields nonzero and either the MSGNO field is not equal to the SQLMSGNO field, or the SEQNO field is not greater than 1.

System Action: Processing of the current SQL statement has ended. A return code value of 20 is returned to the calling routine in register 15.

User Response: Notify your system programmer.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

ARI0512E **An error has occurred.**

SQLCODE *sqlcode* is received from the application server
PRODUCT = *product*
Level = *release*
SQLERRP = *routine-name*
SQLSTATE = *state*
SQLERRD(1) = *nnn*
SQLERRD(2) = *nnn*
SQLERRD(3) = *nnn*
SQLERRD(4) = *nnn*
SQLERRD(5) = *nnn*
SQLERRD(6) = *nnn*
SQLWARN0 = *"c"*
SQLWARN1 = *"c"*
SQLWARN2 = *"c"*
SQLWARN3 = *"c"*
SQLWARN4 = *"c"*
SQLWARN5 = *"c"*
SQLWARN6 = *"c"*
SQLWARN7 = *"c"*
SQLWARN8 = *"c"*
SQLWARN9 = *"c"*
SQLWARNA = *"c"*
SQLERRM = *tokens*

Explanation: Preprocessing or execution of an SQL statement against a non-DB2 Server for VM application server has resulted in the message indicated by *SQLCODE*. The variables (*tokens, nnn*, and *c*) shown are

the values that the non-DB2 Server for VM product would have substituted into the message.

The product that issued the message is shown in the IBM product *PRODUCT*.

The release level of this product is shown in *Level*, which has the form

VvvRrrMm

where vv is the version number, rr is the release number, and m is the modification number. For example, V03R01M0 would be Version 3 Release 1 Modification 0.

System Action: The non-DB2 Server for VM product's action depends on the specific message that was issued from it. If the database manager was affected by the event causing the message, one or more DB2 Server for VM messages should have been issued to describe this effect.

User Response: Look up the reported message in the appropriate manual of the indicated product, and act accordingly.

ARI0513I A warning has occurred.

SQLCODE *sqlcode* **is received from the application server**
PRODUCT = *product*
Level = *release*
SQLERRP = *routine-name*
SQLSTATE = *state*
SQLERRD(1) = *nnn*
SQLERRD(2) = *nnn*
SQLERRD(3) = *nnn*
SQLERRD(4) = *nnn*
SQLERRD(5) = *nnn*
SQLERRD(6) = *nnn*
SQLWARN0 = *"c"*
SQLWARN1 = *"c"*
SQLWARN2 = *"c"*
SQLWARN3 = *"c"*
SQLWARN4 = *"c"*
SQLWARN5 = *"c"*
SQLWARN6 = *"c"*
SQLWARN7 = *"c"*
SQLWARN8 = *"c"*
SQLWARN9 = *"c"*
SQLWARNA = *"c"*
SQLERRM = *tokens*

Explanation: Preprocessing or execution of an SQL statement against a non-DB2 Server for VM application server has resulted in the message indicated by *SQLCODE*. The variables (*tokens*, *nnn*, and *c*) shown are the values that the non-DB2 Server for VM product would have substituted into the message.

The product that issued the message is shown in the IBM product *product*.

The release level of this product is shown in *level*, which has the form

VvvRrrMm

where vv is the version number, rr is the release number, and m is the modification number. For example, V03R01M0 would be Version 3 Release 1 Modification 0.

System Action: The non-DB2 Server for VM product's action depends on the specific message that was issued from it. If the database manager was affected by the event causing the message, one or more DB2 Server for VM messages should have been issued to describe this effect.

User Response: Look up the reported message in the appropriate manual of the indicated product, and act accordingly.

Chapter 9. ARI0520–ARI0535, ARI5806 Catalog Generation/Migration/Index Reorganization Messages

All messages written during catalog generation, catalog migration, and catalog index reorganization are written to the SYSPRINT file. These include any error messages as well as status messages. If an error occurs while trying to write to SYSPRINT, message ARI0527A is written to the virtual machine console.

ARI0520E **Storage allocation failure. Reason Code =** nn**.**

Explanation: In allocating virtual storage during catalog generation, a CMS free storage macro failure occurred. The cause of the error is described by the REASON value nn, which is the return code from the free storage macro:

1 Not enough storage space available.

any other value
DB2 Server for VM system error.

System Action: Database generation ends.

System Programmer Response: Take action based on the REASON value:

1 The DB2 Server for VM virtual machine does not have enough virtual storage. Either modify DB2 Server for VM initialization parameters so that less storage is required, or reconfigure the system so you have a larger database manager virtual machine. Restart the database generation.

Any other reason -
Make a record of what went wrong and contact the designated support group for your installation. When the problem is corrected, restart the database generation.

ARI0521E **ACCESS error** &1C. **MACRO** &2.**, input file. Reason Code =** nn**. Return Code =** nn**.**

Explanation: An error occurred while the system was trying to access the ARISCAT file. This file is a VM card image CMS file. The Reason Code is the Primary Error Code. The Return Code value is the Secondary Error Code. For further explanation, look up the displayed error codes under either "Chapter 24. Include File Access Errors" on page 411, or "INCLUDE FILE ERRORS" in the HELP text.

System Action: If *&1C.* is ARISCAT, database generation ends unsuccessfully.

Programmer Response: If *&1C.* is ARISCAT, take action based on the Reason Code and Return Code values.

ARI0522E **Catalog generation control record error** catalog generation control record**.**

Explanation: An error occurred while trying to process a catalog generation control record from the ARISCAT file. This file is a VM card image CMS file. One of the following conditions was detected:

- A missing or invalid keyword
- A missing or invalid parameter
- An invalid numeric field
- An invalid record format.

The invalid record is printed as the second line of the message.

System Action: Database generation ends.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation. Restart the database generation after the problem is corrected.

ARI0523E **DBSS error occurred while processing the system catalog. Reason Code =** nn**.**

Explanation: In an attempt to update the database during the catalog generation stage of database generation, or the catalog update phase of ADD DBSPACE, or while catalog migration was processing the catalog, a DBSS error occurred. The cause is described by the Reason Code=nn value in the message text. The Reason Code values are DBSS return codes. Refer to the description of the DBSS return codes under "Chapter 18. DBSS Return Codes" on page 271.

System Action: Processing is terminated.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Refer to the description of the DBSS return codes under "Chapter 18. DBSS Return Codes" on page 271. Make a record of what went wrong and contact the designated support group for your installation.

If the problem occurred during database generation, restart database generation when the problem is corrected.

If the problem occurred during ADD DBSPACE (SQLADBSP) execute the SQLSTART EXEC with the following parameters:

© Copyright IBM Corp. 1987, 2000

```
STARTUP=W
SYSMODE=S
PROGNAME=ARISEGB
LOGMODE=current value
```

This is necessary because the information about the DBSPACEs is already recorded in the database directory minidisk. By specifying the above parameters only the catalog update phase of ADD DBSPACE will be executed.

ARI0524E **Required** objecttype objectname **not found in** catalogname**.**

Explanation: In an attempt to locate the DB2 Server for VM *objecttype objectname*, an error occurred. *objecttype* can be index, link, column, catalog or package. If *objecttype* is an index, link, or column, the associated *catalogname* is displayed in the message text.

System Action: The DB2 Server for VM database maintenance is terminated.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation. Restart the database maintenance function when the problem is corrected.

ARI0525E **Number of columns is incorrect in catalog** catalogname**.**

Explanation: This is a DB2 Server for VM programming error. An inconsistency was found in the required number of columns for *catalogname* and the actual number of columns which the database manager defined for this catalog during database generation. The catalog generation stage determined that internal data areas were incorrect.

System Action: Database generation is terminated.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation. Restart database generation when the problem is corrected.

ARI0526A **Error on user ID record.** user ID record

Explanation: A user-supplied user ID record was being processed during the catalog generation stage of database generation and a syntax error or missing/invalid keyword/parameter was detected. The incorrect record is displayed as the second line of the message.

System Action: The user ID record is ignored and database generation continues.

System Programmer Response: The database has been successfully created without the user ID on this user ID record defined in the catalog table SYSTEM.SYSUSERAUTH. You may either:

- Enter a GRANT command for the user specifying the appropriate authority and password. See the *DB2 Server for VSE & VM Interactive SQL Guide and Reference* manual.
- Compare the invalid record with the description of the NAME column in the SYSTEM.SYSUSERAUTH catalog table as discussed in the *DB2 Server for VM System Administration* manual.

ARI0527E access **error** filename**. Reason Code =** nn**. Return Code =** cc **[**nnnn**].**

Explanation: An error occurred in an attempt to access SYSPRINT or SYSIN.

If the *nnnn* value is included in the message, it reflects an attempt to display message **ARI***nnnn*.

The Reason Code value is the Primary Error Code. The Return Code value is the Secondary Error Code. For further explanation, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Action: If **access** is GET or CLOSE for file name SYSIN, the system was trying to read an INSERT record during the catalog generation. The record and any additional records will be ignored by the system; the database generation continues.

If *access* is OPEN for file name SYSPRINT, database generation or catalog index reorganization processing ends. Catalog migration processing continues.

If *access* is PUT or CLOSE for file name SYSPRINT, processing continues.

Operator Response: For corrective action, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Programmer Response: See Operator Response.

Refer to message ARInnnn (where nnnn is the value identified in the message text) to determine what system condition caused the message to be displayed.

ARI0528I **User** userid **defined with password** password auth**.**

Explanation: User *userid* has been defined to the database with password *password* and authorization *auth* (where auth may be any combination of DBA, SCHED, and RES) and inserted into the catalog table SYSTEM.SYSUSERAUTH.

System Action: Database generation continues.

ARI0529I &1**. is completed.**

Explanation: This is the expected response from the database manager. The function is successfully completed. &1. can be:
- CATALOG GENERATION
- INDEX REORGANIZATION
- FORWARD MIGRATION

System Action: The program continues processing until the end.

ARI0530E **An error occurred while attempting to free storage. Reason Code =** nn**.**

Explanation: In freeing virtual storage during catalog generation or catalog migration, a CMSSTOR macro failure occurred. The Reason Code value *nn* is the return code from the CMSSTOR macro. This condition is a VM system error.

System Action: Processing continues.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

ARI0531E **DBSS error. Return Code =** &1A**.. Reason Code =** &2A., &3**.=**&4**..**

Explanation: An error occurred during catalog index reorganization when the database manager attempted to perform a DBSS action.

The DBSS returns an error code shown in the Return Code value. (See "Chapter 18. DBSS Return Codes" on page 271 for an explanation of DBSS return codes.)

The Reason Code value describes the type of DBSS operation that was in progress when the error occurred.

10 Opening a scan

20 Closing a scan

30 Deleting (either a row or an Index Control Record)

40 Next

50 Fetch (either a row or an Index Control Record)

60 Update

70 Insert (applies only to the Index Control Record)

80 Acquiring an internal DBSPACE

90 Releasing an internal DBSPACE

&3. is either INDEX or CATALOG.

&4. is either index name or catalog name.

System Action: Utility execution stops. All changes to the database are rolled back.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

ARI0532I **Package** &1**. has been marked invalid.**

Explanation: Catalog index reorganization processing has marked a package invalid.

&1. is owner.packagename.

User Response: You can use this message to determine which packages catalog index reorganization has marked invalid.

Note: If an error occurs that causes catalog index reorganization to fail, these messages no longer apply.

ARI0533E **Catalog migration error processing** object**.**

Explanation: An error occurred during forward migration while modifying the SYSTEM CATALOG. The object can be a SYSTEM CATALOG table, column, or index. This message is followed by message ARI0504I, which indicates the condition that caused the error and message ARI0505I. Message ARI0505I provides the SQLCODE information.

Note: The SQLCODE information in the accompanying ARI0505I message should be less than `0` (zero).

System Action: The catalog migration is terminated.

System Programmer Response: If this error occurs during forward migration processing (ARISMEX EXEC), make a record of what went wrong and contact the designated support group for your installation.

ARI0534E **A column named** &1**. already exists in** &2**..**

Explanation: A column name *&1.* was found in *&2.* that was not created by the database manager. The forward migration process cannot continue.

System Action: The program is terminated.

System Programmer Response: If you are migrating from SQL/DS™ 2.2.0, you must change the column name from *&1.* to another name.

The recommended procedure is:

1. Drop any indexes that have been created on the named column and any programs that reference or use it.
2. Drop the *&1.* column by deleting the row in SYSCOLUMNS where *&1.* = CNAME. Recreate the column in *&2.* with a different name.
3. Recode and repreprocess the programs that were dropped in step 1.

4. Rerun forward catalog migration. (This step can be performed before step 3.)

If you are migrating from SQL/DS 3.1.0 or later, contact the support group for your installation.

ARI0535E This catalog is not a release n **catalog.**

Explanation: If you are performing forward migration, this error occurred because the catalog information indicates that the catalog is not at a release level supported by migration.

System Action: The program terminates with no catalog updates performed.

System Programmer Response: If you are certain that the catalog is at the proper release level, contact the designated support group for your installation.

ARI5806E There is insufficient space in the server directory for migration.

Explanation: The invalid entities table cannot be migrated to the new format because there is not enough space in the database directory. There must be at least four DBSPACE blocks remaining in the database directory for this migration step.

System Action: Database migration ends.

System Programmer Response: Increase the size of the directory and rerun this migration step. Refer to the *DB2 Server for VM System Administration* manual for more information on increasing the size of the directory. file

Chapter 10. ARI0536–ARI0599, ARI5361–ARI5406 Preprocessor Messages

Messages from a preprocessor are written to the SYSPRINT and SYSPUNCH files. In addition, an expanded SQLCA structure will be placed in the modified source for all SQL errors found by the preprocessors or the database manager.

Preprocessing can result in messages from the preprocessor and other components of the database manager.

ARI0536I **Internal error** n1**. Records remaining** n2**.**

Explanation: An error occurred during the merging of system internal files or closing system files. The *n1* value indicates the type of error; the *n2* value gives additional data, as follows:

n1	**Meaning and** *n2* **Value**
10	Unexpected end of file on SYS002 workfile; *n2* = 0.
20	Unexpected end of file on SYS001 workfile; *n2* = 0.
30	Records not all merged for SYS001 workfile; *n2* = number of records remaining.
40	Records not all merged for SYS002 workfile; *n2* = number of records remaining.
50	Error while closing SYSIPT; *n2* = 0.
60	Error while closing SYS001; *n2* = 0.
70	Error while closing SYS002; *n2* = 0.
80	Error while closing SYSPCH; *n2* = 0.
90	Error while closing SYSLST; *n2* = 0.

System Action: Processing continues; however, if the value *n1* is 10, 20, 30, 40, or 80, the results may not be what you expected.

System Programmer Response: This condition is a DB2 Server for VM error. If problems persist, contact the designated support group for your installation.

ARI0537E operation **failure on secondary input source** textname**. Return Code =** n1**. Reason Code =** n2**.**

Explanation: During processing of an INCLUDE statement, the specified operation (FIND or GET) for the secondary source identified by *textname* failed. *textname* identifies an 80-byte fixed-length record CMS file.

The Return Code value *n1* is the Primary Error Code. The Reason Code value *n2* is the Secondary Error Code. For further explanation, look up the displayed error codes under either "Chapter 24. Include File Access Errors" on page 411, or "INCLUDE FILE ERRORS" in the HELP text.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Take action based on the Return Code and Reason Code values.

ARI0538E **The following is incorrect or missing: [**value**] was found for** *keyword***.**

Explanation: The value supplied for the indicated preprocessor option *keyword* was either invalid or missing.

Probable causes of the error are:

- You did not specify a value, or you specified an incorrect value for the ISOLation keyword. Allowed values are CS, RR, UR, and USER.
- You did not specify a value or you specified an incorrect value for the DATE or TIME keyword. Allowed values are ISO, JIS, USA, EUR, LOCAL.
- You did not specify a value or you specified an incorrect value for the STDSQL keyword. Allowed values are NO, 86, and 89.
- You did not specify a value or you specified an incorrect value for the CHARSUB keyword. Allowed values are SBCS, MIXED, and BIT.
- You did not specify a value or you specified an incorrect value for the CTOKEN keyword. Allowed values are YES and NO.
- You did not specify a value or you specified an incorrect value for the LABEL keyword. The label-text can be a maximum of 30 characters in length.
- You did not specify a value or you specified an incorrect value for the CCSIDSBCS, CCSIDMIXED, or CCSIDGRAPHIC keywords. Allowed values fall in the range from 1 to 65534 inclusively, and must be allowed in the database system.

  Note: Only the first eight characters appear in the message text.
- You did not specify a value, or you specified an incorrect value for the DBList keyword. Server names can be a maximum of 18 characters in length.

© Copyright IBM Corp. 1987, 2000

- There are misplaced commas in the server name list specified for the DBList keyword.
- There are duplicate server names in the server name list specified for the DBList keyword.
- You did not specify a value, or you specified an incorrect value for the QUALifier keyword. Qualifier names can be a maximum of 8 characters in length.
- You did not specify a value, or you specified an incorrect value for the OWner keyword. Owner names can be a maximum of 8 characters in length.
- You did not specify a value, or you specified an incorrect value for the MSGQueue keyword.

System Action: Preprocessing or CBND ends. A new package is not produced. If the package previously existed, it is restored.

User Response: Supply the valid preprocessor or CBND option and rerun the preprocessor.

ARI0539I description **was expected, but** string **was found for** option**.**

Explanation: During preprocessing, the database manager discovered that a DB2 Server for VM Preprocessor parameter contained an unidentifiable string.

description is a phrase indicating what was expected.

string is a string that could not be identified.

option is a preprocessor parameter containing the string in question.

For example: For the LINECOUNT option, a number is required. If the character string **6P** was somehow entered as the LINECOUNT value, this message would read:

```
NUMERIC VALUE was expected
but (6P) was found for LINECOUNT.
```

System Action: Preprocessing continues. For LINECOUNT, the value is set to 60.

User Response: If the default value of LINECOUNT is not acceptable, correct the string and rerun the preprocessor.

ARI0540E **The secondary input file named in an INCLUDE statement was not found.**

Explanation: A preprocessor SQL INCLUDE statement specified a textname that could not be found.

System Action: Processing continues, but the INCLUDE statement that is in error is not processed further. A new package will not be produced. If the package existed previously, it will be restored.

User Response: Check the spelling of the specified textname for accuracy. Ensure that the textname specified exists and resides in a CMS file on a disk accessed by the user's virtual machine. Also ensure that the file type of the secondary input source is consistent with the preprocessor being invoked. See message ARI0542E for the required file types.

ARI0541E **You have incorrectly specified the TEXTNAME portion of an SQL INCLUDE statement.**

Explanation: A preprocessor SQL INCLUDE statement was specified with an incorrect string. One of the following may be the cause of the error:

- Text-file-name is longer than 8 characters.
- Text-file-name contents do not follow the rules for SQL identifiers:
  - First character of identifier must be a letter (A to Z), @, #, or $.
  - Identifier must contain only letters (A to Z), @, #, $, numbers (0 to 9), or underscores (_).
- Text-file-name was omitted.
- Additional statements may not appear on the same line after an INCLUDE (except in COBOL programs).
- The INCLUDE statement for text inclusion must be specified entirely on one line in PL/I and Assembler programs.

System Action: Processing continues, but the INCLUDE statement in error is not processed further. A new package will not be produced. If the package existed previously, it will be restored.

User Response: Correct the text-file-name in the INCLUDE statement. Rerun the preprocessor.

ARI0542E **An INCLUDE statement was specified within the extended include source** *text-file-name***.**

Explanation: Secondary input containing an SQL INCLUDE statement other than an INCLUDE SQLCA or INCLUDE SQLDA was found in the *text-file-name* specified. This is not allowed.

text-file-name is the file name of a CMS file, with the file type depending on the host language:

```
ASMCOPY  - Assembler
CCOPY    - C
COBCOPY  - COBOL
FORTCOPY - FORTRAN
PLICOPY  - PL/I
```

System Action: Processing continues, but the INCLUDE statement in error is not processed further. A new package will not be produced. If the package existed previously, it will be restored.

User Response: Delete all invalid INCLUDE statements from the secondary input specified by *text-file-name*. Rerun the preprocessor.

ARI0543I INCLUDE SQLCA is not permitted when you request NOSQLCA support.

Explanation: When preprocessing an application program with the NOSQLCA support, you cannot include the SQLCA structure in the source program.

System Action: The database manager ignores the INCLUDE SQLCA statement. Preprocessing continues.

User Response: Remove the INCLUDE SQLCA statement.

ARI0544E A graphic string is either too long, incorrectly terminated, or contains bytes that are not correctly paired for a graphic string.

Explanation: A graphic string cannot cross input record boundaries. It must also contain an even number of bytes between the opening SO (X'0E') and the closing SI (X'0F').

System Action: Preprocessing continues, but a new package will not be produced. If the package existed previously, it will be restored.

User Response:

1. Provide the correct pairing of bytes in the graphic string or decrease the size of the graphic string.
2. Repreprocess the program.

ARI0545E You have not declared cursor cursorname**.**

Explanation: During an OPEN, FETCH, PUT, or CLOSE cursor statement, the database manager discovered that the cursor being referenced had not been declared.

System Action: Processing of the SQL statement has ended.

User Response: Define the cursor by means of a DECLARE cursor statement.

ARI0546I The preprocessor did not process the package.

Explanation: The FORTRAN preprocessor performs initialization processing before any package changes are made. If one of the initialization processing items fails, the preprocessor ends with a nonzero return code.

Preprocessor message ARI0598I is incomplete because no processing has been done against a package. If a package existed prior to the preprocessor run that gave this message, the preprocessor performed no operations against the package and it is in the same state it was before the preprocessing. Similarly, if a preprocessor run receives this message and no package existed previously, a package will not exist after the preprocessing is complete.

User Response: Take appropriate action for the error messages that precede this message.

ARI0547I You did not declare any host variables in progname.

Explanation: The preprocessor detects valid host variable declarations. At the end of each program, or FORTRAN program unit, the preprocessor writes messages ARI0560I and ARI0561I, which list the valid host variable declarations that it found. If the preprocessor found no host variable declarations, message ARI0547I is issued.

progname is the name of the program or program unit (as specified in a FORTRAN SUBROUTINE, FUNCTION, PROGRAM, or BLOCK DATA statement).

User Response: If you did not have any host variable declarations in the specified program, then take no action. If you meant to have host variable declarations, be sure that you did not incorrectly specify any of them, and check that you placed all host variable declarations in an SQL DECLARE section.

ARI0548E The cursor cursorname **has already been declared.**

Explanation: During an SQL DECLARE statement, the database manager discovered that the cursor being referenced had already been declared.

System Action: Processing of the SQL statement has ended.

User Response: Ensure that the cursor being declared has a unique name.

ARI0549E You have declared more than 512 cursors in this program.

Explanation: During preprocessing, the database manager discovered that more than 512 cursors were declared for the current program unit. This exceeds a DB2 Server for VM preprocessor limit.

System Action: Processing of the SQL statement has ended.

User Response: Rewrite the program to contain fewer DECLARE cursor statements.

ARI0550E Host variable *variable* **is declared with** *attribute* **that is incorrect for its use in the SQL statement.**

Explanation: During preprocessing, the database manager discovered that *variable* was declared using an incorrect attribute.

The contents of *variable* can be any of:

- The name of the variable with the declaration error

- **user-id**, indicating that the user ID host variable has a declaration error
- **password**, indicating that the password host variable has a declaration error
- **database**, indicating that the application server-name host variable has a declaration error.

The contents of *attribute* can be any of:

- **data type**, indicating that the data type attribute of the variable referenced in *variable* is incorrect
- **length**, indicating that the length attribute of the variable referenced in *variable* is incorrect
- **an attribute**, indicating that the structure or array attribute of the variable referenced in *variable* is incorrect.

Some examples of incorrect attributes are:

- The user ID or the password host variable is not fixed character length 8.
- The server-name host variable is not a character string with a length between 1 and 18.
- The host variable that is used to contain an SQL statement is not a fixed length character string in FORTRAN.
- The indicator variable is not a small integer variable.
- The structure is not a valid host structure.
- Use of a host structure is not permitted.
- An indicator array used in place of a host variable or main variable.

System Action: Processing of the SQL statement has ended.

User Response: Correct the declaration of the attribute for the host variable.

ARI0551I LINECOUNT specified incorrectly. The database manager set it to 60.

Explanation: During the checking of the preprocessor options, an error was found in the specification of the LINECOUNT option. The error is noted here and the default value of 60 for the LINECOUNT option is set.

System Action: The LINECOUNT value is set to 60. Preprocessing continues.

User Response: If you want a LINECOUNT value other than 60, correct the error in the LINECOUNT option specification and rerun the preprocessor.

ARI0552I LINECOUNT value > 32767 or < 10. The database manager set it to 60.

Explanation: The value specified for LINECOUNT is outside the range of valid values (from 10 to 32767).

System Action: The LINECOUNT value is set to 60. Preprocessing continues.

User Response: If you want a LINECOUNT value other than 60, correct the error in the LINECOUNT option specification and rerun the preprocessor.

ARI0553E identifier **name either missing, or greater than** nn **bytes.**

Explanation: During preprocessing, the database manager discovered an identifier greater than the number of bytes **nn**, the maximum allowable for that type of identifier.

- *identifier* is the host variable name, user ID, password, program name, file name, statement name, statement, label, cursor name, or database name.
- *nn* is the maximum length of the *identifier* name.

System Action: Preprocessing of the SQL statement ends.

User Response: Correct the length of the identifier name.

For more information about the correct lengths of identifier names, refer to the identifiers description in the *DB2 Server for VSE & VM SQL Reference* manual.

ARI0554E SQL statement contains more than 256 host variables.

Explanation: During preprocessing, the database manager discovered more than 256 host variables referenced in a single SQL statement. This exceeds a DB2 Server for VM limit.

System Action: Processing of the SQL statement has ended.

User Response: Change the SQL statement to contain fewer than 256 host variables.

ARI0555E You did not follow the colon marker with a host identifier.

Explanation: During preprocessing, the database manager identified a colon marker that was not followed by a host identifier. The host identifier must be the name of a host variable or indicator variable. In those languages that support host structures, the host identifier can be a host structure or host structure field. The indicator variable can be an indicator array.

System Action: Processing of the SQL statement has ended.

User Response: Change the SQL statement so that a host identifier follows the colon marker. Rerun the preprocessor.

ARI0556E ident-type **contains incorrect characters.**

Explanation: During preprocessing, the database manager discovered that *ident-type* contained characters that are not valid.

System Action: Processing of the SQL statement has ended.

User Response: Change the SQL statement so that all characters are valid.

ARI0558I **A statement label is not allowed on this SQL statement. The label has been removed:** &1..

Explanation: During preprocessing, the database manager discovered a label on a BEGIN DECLARE SECTION, END DECLARE SECTION, WHENEVER, or INCLUDE statement.

System Action: The identified label is blanked out.

ARI0559E **The** *language* **preprocessor does not allow** *function***.**

Explanation: The specified language preprocessor does not permit the function identified in the message.

System Action: Processing of the SQL statement has ended.

User Response: Change the SQL statement to eliminate the disallowed function.

ARI0560I **The DB2 Server for VM** *language* **preprocessor recognized the following host variable declarations: (* before the level number indicates a structure which may be used as a host structure)**

LVL NAME SQL TYPE LENGTH

Explanation: This message always appears together with message ARI0561I. The explanation for message ARI0561I covers both messages.

ARI0561I *lvl varname vartype varlength*

Explanation: During preprocessing, the preprocessor identifies and keeps track of declarations of valid host and indicator variables, host structures, and indicator arrays. Messages ARI0560I and ARI0561I list all of these declarations. These messages are issued at the end of each program unit that contains these declarations.

language identifies the preprocessor (assembler, C, COBOL, FORTRAN, or PL/I).

lvl is the logical level of a field within a structure. The logical level of a field is always one unit deeper than that of its immediate containing structure. All fields of a structure are reported together and listed sequentially in this message.

Scalar variables have a logical level of one (1).

If *lvl* begins with an asterisk (**lvl*), *varname* is the name of a host structure. If *lvl* is a structure, but it does not conform to the rules for a host structure, then *lvl* does not begin with an '*'.

varname is the declared name of a valid host variable, structure, structure field, indicator variable, or indicator array.

If *varname* is "OCCURRENCES *nn*", *varname* on the line immediately above it is the name of an indicator array with *nn* occurrences.

vartype is the SQL data type of the variable identified in *varname*.

If *vartype* is empty, *varname* is a structure.

varlength is the number of bytes of storage required to store the host variable, structure field, or indicator array element.

If *varlength* is empty, *varname* is a structure.

User Response: You can use this message and error and warning messages to find errors in the declarations of host variables, structures, and indicator arrays. Review the variables, structures, and arrays listed in these messages to ensure that the specifications are being recognized.

ARI0562I **The following SQL error and/or warning conditions occurred for the** language **preprocessor initiated SQL** statement type **statement.**

Explanation: During preprocessing, the execution of an SQL statement by the named preprocessor resulted in a nonzero SQLCODE or a warning flag in the SQLCA.

language identifies the ASSEMBLER, COBOL, C, PL/I, or FORTRAN preprocessor.

statement type identifies the type of SQL statement tried.

Following this message will be SQL message lines that describe the SQL conditions encountered.

System Action: If the SQL condition encountered was an error (negative SQLCODE for example), then processing of the statement specified in the message ends. The ASSEMBLER, COBOL, C, PL/I, or FORTRAN preprocessor issues a ROLLBACK WORK statement. Preprocessing ends.

If the SQL condition encountered was a warning (positive SQLCODE for example), preprocessing continues.

User Response: Analyze the SQL problem described in the messages that follow this message.

If an error has occurred, correct the error and rerun the preprocessor.

If this is a warning, check that the warning is acceptable. If it is not, attempt to correct the warning condition, and rerun the preprocessor.

ARI0563I jjj **records read from source input file.** kkk **records written to source output file.**

Explanation: This message tells the number of records that the preprocessor read from the input file and the number of records written to the output file. The first number (*jjj*) is the count of all records read from the input file, including records read from any secondary input file specified in an SQL INCLUDE statement. The number of records written (*kkk*) is a count of the records written to the modified source output file. This number is not the number of records written to the preprocessor listing file.

ARI0564E **Source input file is empty.**

Explanation: The input file which the preprocessor tried to read is empty.

System Action: A new package will not be produced. If the package existed previously, it will be restored.

User Response: Ensure that the source program to be preprocessed is in the input file.

ARI0565I **The input source contains either no SQL statements, or no SQL statements that will create a package.**

Explanation: One of the following occurred:

- The EXEC SQL prefix was not found on any statements in the input source program.
- The EXEC SQL prefix was found only on statements that do not create a package (for example: CONNECT, COMMIT, or ROLLBACK). COMMIT and ROLLBACK apply only to the FORTRAN preprocessor.

This condition may cause an empty package to be stored in the database for this program.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Check the preprocessor input file to determine if errors were made in the SQL statements embedded in the source program. Correct all errors. Rerun the preprocessor.

ARI0566E **You did not declare host variable** *varname***.**

Explanation: The host variable *varname* is used in an SQL statement, but it was not recognized as a valid host variable declare in the current program or subprogram unit. For a FORTRAN program, a host variable used in an SQL statement must be declared in an SQL DECLARE section in the same program or subprogram unit in which the SQL statement occurs.

System Action: Processing of the SQL statement ends.

User Response: Ensure that the *varname* specified in the message is correctly declared. Messages ARI0560I and ARI0561I of the preprocessor listing output may be helpful in resolving the problem.

ARI0567E **You have already declared host identifier** *ident-name***.**

Explanation: The host identifier *ident-name* is declared more than once in an SQL DECLARE section, in this program or subprogram unit. The host identifier is the name of a host variable or indicator variable. In those languages that support host structures, the host identifier can be a host structure or host structure field. The indicator variable can be an indicator array.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Remove the duplicate declaration. Rerun the preprocessor.

ARI0568E **The SQL statement or** *IF* **statement exceeds the limit of 125 lines.**

Explanation: The SQL statement or FORTRAN IF statement is longer than 125 lines (1 plus 124 continuation lines). This number of lines exceeds a FORTRAN preprocessor limitation.

This message may also result from having a large number of comment lines or blank lines in or immediately following the statement mentioned above.

System Action: The statement that caused the message is treated as a comment in the modified source output and is not used. Preprocessing continues. A new access module is not produced. If the access module existed previously, it is restored.

User Response: Compress the statement to fit on not more than 125 lines (embedded comment lines and blank lines included). If this is not possible, break the statement into two or more separate statements. Rerun the preprocessor. If the error is caused by comment lines or blank lines immediately following the statement, precede the comment lines or blank lines with a FORTRAN CONTINUE statement and rerun the preprocessor.

ARI0569E string1 **was expected, but** string2 **was found.**

Explanation: During processing of an SQL statement, the database manager expected *string1* but discovered *string2*. For example, if the preprocessor encountered only a fragment of a command, such as EXEC SQL FETCH, this message would be displayed as:

```
CURSOR NAME was expected
but END OF STATEMENT was found.
```

System Action: Processing of the SQL statement has ended.

User Response: Correct the syntax of the SQL statement.

ARI0570I **User** username **not authorized to perform** action **on object** object.

Explanation: During preprocessing, an SQL statement was found which, when executed, would perform action *action* on object *object*. The user *username* who is preprocessing this program is not authorized to perform this action on this object. However, the preprocessor will treat this as a warning and assumes that the user will be properly authorized by the time the program is executed.

User Response: Ensure that the user is authorized before executing the program.

Note: This condition will cause additional system overhead when the program is executed. If it is a frequently executed SQL statement and if performance is important, the user should consider repreprocessing the program once the user has been properly authorized.

ARI0571I **User** username **not authorized to** action.

Explanation: During preprocessing, an SQL statement was found which, when executed, would perform the action *action*. The user *username* who is preprocessing this program is not authorized to perform this action. However, the preprocessor will treat this as a warning and assumes that the user will be properly authorized by the time the program is executed.

User Response: Ensure that the user is authorized before executing the program.

Note: This condition will cause additional system overhead when the program is executed. If it is a frequently executed SQL statement and if performance is important, the user should preprocess the program again after being properly authorized.

ARI0572E **Cursor** cursorname **is not valid.**

Explanation: This error occurs when preprocessing an OPEN, FETCH, PUT or CLOSE cursor statement. It means that an error was discovered while processing the DECLARE CURSOR or associated PREPARE statement. The cursor has been marked invalid.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Correct the error in the DECLARE CURSOR or associated PREPARE statement. Rerun the preprocessor. If you specify ERROR as a preprocessing parameter, the cursor may not be marked invalid when the error is detected, and the OPEN, FETCH, PUT, or CLOSE CURSOR statement is processed as if the cursor is valid.

ARI0573E **An SQL statement was found in a program other than the first program or the main program.**

Explanation: This message is issued only by the COBOL preprocessor. A DB2 Server for VM statement was found in the source after the end of the main program of a nested program, or after the end of the first program in a batch compilation.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Move the DB2 Server for VM statements and host variable declarations to the main program or first program and rerun the preprocessor.

ARI0574E **Label is greater than** number **characters.**

Explanation: While preprocessing an application program, a label was found on an SQL statement or host variable which was greater than the number of characters allowed by the host language.

For Assembler, C, COBOL, COBOL II, and PL/I, the maximum number of characters allowed in an SQL statement is 256.

For FORTRAN, this message refers to the label used in an SQL WHENEVER statement with the GOTO action specified:

- If the first character of the specified label is numeric, it is assumed that the label is numeric and refers to a statement number in the FORTRAN program. These labels can have no more than 5 characters.
- If the first character of the specified label is alphabetic, it is assumed that the label is a variable name to be used in a FORTRAN assignment statement. These labels can have no more than 6 characters.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Correct the label. Rerun the preprocessor.

ARI0575E **Comma delimiter between host variables not found.**

Explanation: While processing an OPEN, FETCH, PUT, or EXECUTE SQL statement, the preprocessor found a *host-variable-list* that does not have comma delimiters between the *host-variable*s. A comma is required between *host-variable*s.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Add commas between host identifier pairs. Rerun the preprocessor.

ARI0576E **The preceding source program lines contain a string of nonblank characters that exceeds 256 characters. This exceeds a preprocessor limitation.**

Explanation: The preprocessor encountered a string of characters of length greater than 256. This will cause an overflow of preprocessor internal storage areas.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Break the statement into smaller pieces by separating the pieces by blanks.

ARI0577E **Unable to release storage. Reason Code =** nn.

Explanation: During preprocessing, the preprocessor was unable to release storage. The reason code is the return code from a CMS DMSFRET macro. This is a DB2 Server for VM preprocessor error.

System Action: This error may occur after the preprocessor has attempted to store the package in the database. The preprocessor tries to continue after the point of this error to complete termination processing.

User Response: This message should be followed by message ARI0598I which will tell you the state of your package. Final preprocessor results may be incorrect or in an inconsistent state. Notify your system programmer.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

ARI0578I **INCLUDE SQLCA ignored. It has already been found.**

Explanation: An SQL INCLUDE SQLCA statement was encountered, but an INCLUDE SQLCA statement was already found in the current program/subprogram unit.

System Action: The SQL statement that caused the error is commented out in the modified source output and ignored. Preprocessing will continue. The preprocessor warning count will be incremented. A nonzero return code will be returned by the preprocessor.

User Response: Remove the extra SQL INCLUDE statement from the input source.

ARI0579I **The preprocessor encountered the following SQL error or warning conditions on return from a call to store the package:**

Explanation: The database manager returned a nonzero SQLCODE or a warning flag in the SQLCA during an attempt to store the package in the database. The SQL message describing the SQL error or warning follows this message.

System Action: If an error occurred, preprocessing ends. Message ARI0598I (which follows the SQL error messages) indicates that the package was not produced or that it was restored if the package previously existed.

If a warning occurred, preprocessing continues and message ARI0598I indicates that the package was produced or replaced if it previously existed.

User Response: Check the SQL message lines that follow this message for a description of the SQL error or warning.

If this was a warning, check that the warning is acceptable. If it is not, attempt to correct the warning condition and rerun the preprocessor. Correct the problem and rerun the preprocessor.

ARI0580E **Required preprocessor option** name **not specified.**

Explanation: The identified option keyword is required in the parameters passed to the preprocessor, but it was not provided.

name is 'PREPNAME='.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Include the missing keyword in the preprocessor options and rerun the preprocessor.

ARI0581E **Incorrect or missing parameter on** name **keyword.**

Explanation: The information supplied on the indicated preprocessor option keyword was either not valid or missing.

Probable reasons for error are:

- No program name or incorrect program name specified on 'PREPNAME=' keyword. The program name must be a character string of up to 8 characters.
- No user ID or password, or incorrect user ID or password specified on 'USERID=' keyword. User ID and password must be a character string of up to 8 characters.
- No preprocessor file was specified or the format is incorrect. The correct format is 'PREPFILE=(*filename*)',

'PREPFILE=(*filename filetype*)' or 'PREPFILE=(*filename filetype filemode*)', as documented in the *DB2 Server for VSE & VM Application Programming* manual.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Supply valid parameter and retry the operation.

ARI0582E Terminating error occurred during preprocessing.

Explanation: An SQL return code caused termination of the preprocessor.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Check the previous error message to determine the cause of the terminating error. Correct the problem. Rerun the preprocessor.

ARI0583E Keywords *option* and *option* were both found. Specify only one.

Explanation: The keywords specified are mutually exclusive, yet both were specified as preprocessor options. Probable reasons for error:

- Two of KEEP, REVOKE, and NEW were specified.
- Both PRINT (PR) and NOPRINT (NOPR) were specified.
- A combination of CHECK, NOCHECK, and ERROR was specified.
- Both QUOTE (Q) and APOST were specified.
- Both GRAPHIC (GR) and NOGRAPHIC (NOGR) were specified.
- The ISOLATION preprocessor option was specified more than once with conflicting values.

  Note: The allowed values are CS, RR, RS, UR, and USER.
- The DATE keyword was specified with different format values, such as DATE (USA) and DATE (ISO).
- The TIME keyword was specified with different format values, such as TIME(USA) and TIME(EUR).

  Note: Allowed formats for each of DATE and TIME are: ISO, JIS, USA, EUR, LOCAL.
- Both STDSQL(89) and STDSQL(NO) were specified.

  Note: STDSQL(86) is a synonym for STDSQL(89).
- Conflicting options of EXPLAIN were specified.
- Both COMMA and PERIOD were specified.
- Both RELEASE (COMMIT) and RELEASE (DEALLOCATE) were specified.
- The CCSIDSBCS preprocessor option was specified more than once with different numeric values. Allowed values for these options must be in the range 1 through 65534 and be allowed in the database manager.
- The CCSIDMIXED preprocessor option was specified more than once with different numeric values. Allowed values for these options must be in the range 1 through 65534 and be allowed in the database manager.
- The CCSIDGRAPHIC preprocessor option was specified more than once with different numeric values. Allowed values for these options must be in the range 1 through 65534 and be allowed in the database manager.
- Here are some other conflicts that can cause this message:
  - REPLACE and NEW
  - PUNCH (PU) and NOPUNCH (NOPU)
  - SEQUENCE (SEQ) and NOSEQUENCE (NOSEQ)
  - BLOCK (BLK), NOBLOCK (NOBLK), and SBLOCK (SBLK)
  - EXIST and NOEXIST
  - SQLAPOST (SQLA) and SQLQUOTE (SQLQ).

  Check the descriptions in the *DB2 Server for VSE & VM Application Programming* manual to resolve conflicts in options or values for CHARSUB, SQLFLAG, CTOKEN, QUALIFIER, OWNER, or LABEL.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Specify only one of the keywords and rerun the preprocessor.

ARI0584E Unable to allocate storage of n1 **bytes. Return Code =** n2.

Explanation: During preprocessing, the preprocessor was unable to allocate storage of the specified size needed for processing. The Return Code value is the return code from a CMS free storage macro.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: For Return Code = 1, increase the virtual machine size. Rerun the preprocessor.

For all other Return Code values, notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0585E operation **failed on** dname**. Reason =** n2**. Return =** n1**.**

Explanation: During preprocessing, the indicated operation (OPEN, CLOSE, GET, or PUT) failed on the indicated file. The *dname* is specified in the ddname of the CMS FILEDEF command.

The Return value is the Primary Error Code. The Reason value is the Secondary Error Code. For further explanation, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: For corrective action, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

ARI0586I **Preprocessing ended with** n1 **errors and** n2 **warnings.**

Explanation: The preprocessor ended with the specified number of SQL errors or SQL warnings.

System Action: Processing ends.

User Response: If errors or warnings occurred, look through the SYSPRINT output for the specific error or warning message. Correct the problem, if necessary, and rerun the preprocessor.

ARI0587I **Object** name **owned by** owner **was not found.**

Explanation: During preprocessing, an SQL statement was found which referenced the named object (for example a table) owned by the specified owner. This object could not be found. However, the preprocessor treats this as a warning condition; it assumes that the object will have been created by the time the program is executed.

name is the name of the object.

owner is the owner of the object.

System Action: Preprocessing continues.

User Response: Ensure that the owner name and object name are correct. If they are correct, ensure that the object is created before executing the program.

Note: This condition will cause additional system overhead when the program is executed. If it is a frequently executed SQL statement and if performance is important, consider repreprocessing the program after the object has been created.

ARI0588E **Incorrect string** string **in preprocessor options.**

Explanation: The preprocessor found the specified 1- to 26-character string in the preprocessor options, but it has no meaning to the preprocessor. Using a preprocessor option file and specifying two preprocessor options on the same input record, or including a comment not beginning with an asterisk (*) may have caused the error.

Note: The preprocessor option USERID=userid/password is not valid when using the DRDA protocol.

Note: The option DYNALC is not valid for the FORTRAN preprocessor.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Correct the preprocessor options. Rerun the preprocessor.

ARI0589E **Error in message formatter, R.C. > 16.**

Explanation: The preprocessor message handler was called to format a message and received a return code greater than 16. The DB2 Server for VM message may not be completely generated.

System Action: Preprocessing continues.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0590E **No valid END DECLARE SECTION found.**

Explanation: A BEGIN DECLARE SECTION was found in the user's source program, but no corresponding END DECLARE SECTION was found before the end of the source program was reached.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Check the program and ensure that a valid END DECLARE SECTION is present after the last valid BEGIN DECLARE SECTION.

ARI0591E **No valid BEGIN DECLARE SECTION found.**

Explanation: An END DECLARE SECTION was found in the source program, but no corresponding BEGIN DECLARE SECTION was found.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Check the program and ensure that a valid BEGIN DECLARE SECTION is present.

ARI0592E **Incorrect continuation during** language **preprocessing.**

Explanation: For an SQL statement embedded in an Assembler application program, columns 1 to 15 of a continuation of an SQL statement or a host variable declaration contained nonblanks.

language identifies that the application program is being preprocessed by the Assembler preprocessor.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: For an Assembler program, ensure that columns 1 to 15 of continuation lines are blank.

ARI0593E **FROM clause of PREPARE or EXECUTE IMMEDIATE statement does not contain the name of a variable.**

Explanation: A quoted string is not allowed for the PREPARE or EXECUTE statement.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Specify only a variable name in the FROM clause of a PREPARE or EXECUTE IMMEDIATE statement.

ARI0594E **Incorrect scan mode in COBOL preprocessing.**

Explanation: An incorrect SCAN Mode was found in COBOL preprocessing.

System Action: This is a DB2 Server for VM error. Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Give the output listing to your system programmer.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

ARI0595E **Incorrect** condition **of a literal.**

Explanation: An alphanumeric literal in COBOL has been improperly ended.

condition is either LENGTH (>120 in COBOL or >160 in COBOL II) or CONTINUATION.

Either:

- Continuation line does not begin with a quote mark, or a beginning quote mark was detected, but an ending quote mark was not found on the same or subsequent line.
- Length of the literal is greater than 120 characters (the maximum allowed by COBOL) or is greater than 160 characters (the maximum allowed by COBOL II).

System Action: The COBOL preprocessor ends processing of the program where the error occurred.

User Response: Correct the literal. Rerun the preprocessor.

ARI0596E **End of source file found without finding either a procedure division, a working storage section, or an END EXEC for the last SQL statement.**

Explanation: During preprocessing of a COBOL program, one of the following conditions has been detected:

- The end of the source program was found without finding a WORKING-STORAGE SECTION.
- The end of the source program was found without finding a PROCEDURE DIVISION.
- The end of the source program was found without finding the END-EXEC terminator to end the last EXEC SQL indicator encountered.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Add the missing item to the source program. Rerun the preprocessor.

ARI0597E **SQL statements other than the INCLUDE statement are not allowed in the declare section.**

Explanation: The SQL statement identifier EXEC SQL was found in a DECLARE SECTION (for example, after a BEGIN DECLARE SECTION and before a valid END DECLARE SECTION). Only the extended INCLUDE statement is allowed in a DECLARE SECTION. No other SQL statement is allowed in a DECLARE SECTION. A possible explanation for this error is that the END DECLARE SECTION statement was inadvertently omitted, or was incorrectly declared.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Ensure that the DECLARE SECTION contains no SQL statements other than an extended INCLUDE statement. Rerun the preprocessor.

ARI0598I **The application server [***server***] *status* the package for *program name*.**

[DATE: *yyyy-mm-dd*
TIME: *hh:mm:ss***]**

Explanation: Preprocessing of *program name* is completed and the *status* of the package is one of the following:

replaced, or produced if it did not already exist
The program was successfully preprocessed, and the package replaced the package for this program from a previous preprocess.

produced
This program was preprocessed successfully for the first time. The preprocessor produced a package.

restored or did not produce
Either:

- The program failed during preprocessing and the previous package for this program is restored.
- A failure did not occur, but no SQL statement encountered in the program required the creation of a package. The previous package for this program is restored.

did not produce
Either:

- The program failed in the preprocessing step and no package exists for it.
- A failure did not occur, but no SQL statement encountered required the creation of a package.

The following SQL statements do not require a package:

```
CONNECT
COMMIT WORK [RELEASE]
ROLLBACK WORK [RELEASE]
DROP PACKAGE/PROGRAM using host variables
Extended PREPARE
Extended DECLARE CURSOR
Extended OPEN, FETCH, PUT, CLOSE
Extended DESCRIBE
Extended EXECUTE
DROP STATEMENT
CREATE PROGRAM
```

System Action: The status is as specified in the explanation.

User Response: Refer to other ARI05xx messages produced for this program.

ARI0599E **The preprocessor termination processing encountered a serious error. The results of the preprocessing are unreliable.**

Explanation: An error occurred during normal or error termination processing of the preprocessor. This error may seriously affect the results of the preprocessor. A specific error message precedes this message. This and other error messages may be issued as write-to-operator messages if both SYSPRINT and SYSPUNCH are disabled.

System Action: The preprocessor tries to complete termination processing. A new package may or may not be produced (or an existing one may or may not be replaced or restored), depending on when the system error occurred.

User Response: The actual problem is usually found in the error messages preceding this one. Follow the USER RESPONSE for those messages.

ARI5361E **Statement name** statementname **has not been prepared.**

Explanation: During preprocessing of a DECLARE or EXECUTE statement, the database manager discovered that the statement name being referenced has not been prepared.

System Action: Processing of the SQL statement has ended.

User Response: Define the statement name by means of the PREPARE statement.

ARI5362E **Statement name** statementname **is not valid because of an error in the PREPARE statement.**

Explanation: The DECLARE or EXECUTE statement in the application cannot be preprocessed because an error was discovered during the processing of the PREPARE statement. The EXECUTE or DECLARE statement references the PREPARE statement by the statement name.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Correct the error in the PREPARE statement. Rerun the preprocessor.

ARI5363E **Cursor name** cursorname **is not valid because the prepared statement, named** statementname**, was a character constant which contained a command other than an INSERT or SELECT.**

Explanation: During processing of a DECLARE statement, an error was encountered involving the statement identified by the *statementname*. That SQL statement was a string spec and PREPAREd as a result

of the previous execution of a PREPARE statement. The only SQL statements that can be used in a string spec for subsequent cursor operations are INSERT and SELECT. The cursor has been marked as invalid.

System Action: Processing of the SQL command has ended.

User Response: Correct the PREPARE statement to process either a SELECT or INSERT statement.

ARI5365E **More than 512 unique statement names have been used in this program.**

Explanation: During preprocessing, the database manager discovered that more than 512 unique statement names were specified in PREPARE statements within the current program unit. This number exceeds a DB2 Server for VM preprocessor limit.

System Action: Processing of the SQL statement has ended.

User Response: Rewrite the program to contain fewer PREPARE statements.

ARI5366E **Statement name** statementname **has already been used in a previous PREPARE statement.**

Explanation: During the preprocessing of an SQL PREPARE statement, the database manager discovered that the statement name *statementname* has already been used in a previously preprocessed PREPARE statement.

System Action: Processing of the SQL statement has ended.

User Response: Ensure that the statement name being PREPAREd has a unique name.

ARI5367E **You cannot use the SQL EXECUTE statement to perform a COMMIT or ROLLBACK.**

Explanation: The commands that control logical units of work cannot be the object of dynamic SQL commands: PREPARE, EXECUTE, or EXECUTE IMMEDIATE.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

ARI5368E **A package name consisting of more than two parts is not supported.**

Explanation: In a DROP PACKAGE statement, a package name of more than two parts was used. This is not supported.

System Action: Processing of the SQL statement ends.

User Response: Correct the statement. Try again.

ARI5370E **The following host identifier reference is ambiguous:** *varname*

Explanation: The host variable *varname* is declared as a field of more than one structure. The preprocessor cannot determine the occurrence of the variable that is referenced in this SQL statement.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Fully qualify the host variable reference. Rerun the preprocessor.

ARI5375E **More than one cursor is declared for statement name** *statement-name*

Explanation: You can only declare one cursor for statement name *statement-name*.

System Action: Preprocessing of the command ends.

User Response: Correct your application program such that there exists only one cursor per statement name.

Note: This message is issued in all environments of the database manager system.

ARI5376E **Preprocessor parameter file** filename **not found.**

Explanation: The preprocessor cannot find the preprocessor option input file.

System Action: Preprocessing ends. The preprocessor does not create a new package nor change the existing package.

User Response: Check the spelling of the specified file name for accuracy. Ensure that the preprocessor parameter file specified exists and resides in a CMS file on a disk accessed by your virtual machine.

ARI5377E **Invalid specification of preprocessor parameters and PREPFILE parameter.**

Explanation: The preprocess request contained both preprocessor parameters (other than the PREPNAME and USERID parameters) directly in the preprocess request and a PREPFILE parameter.

System Action: Preprocessing ends. The preprocessor does not create a new package nor change the existing package.

User Response: If you use the PREPFILE parameter, you cannot specify any prepparm parameters after PREPNAME and USERID in the SQLPREP command. Specify all prepparm parameters except the PREPNAME and USERID parameters in the file

identified by PREPFILE, or specify all preprocessor parameters directly through the PREPPARM parameter and do not refer to a PREPFILE.

ARI5382E **Incorrect option specified in preprocessor option file.**

Explanation: The preprocessor option file contains either the PREPNAME or USERID parameters. Specifying these parameters through the preprocessor option file is invalid.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Remove the invalid parameters from the preprocessor option file. Specify these parameters in the preprocess request directly.

ARI5383I **Warning:** *statement-type* **is not supported.**

Explanation: The database manager system does not support the SQL command shown (*statement-type*). The *statement-type* can be DECLARE TABLE, DECLARE STATEMENT, or DISCONNECT. In other IBM SQL products, these statements cause code expansions to occur. Because the database manager system does not support these statements, the preprocessors did not perform these code expansions. Your program is preprocessed successfully, however, unless other errors occurred and you did not specify the ERROR option as a preprocessing parameter.

System Action: Preprocessing continues.

User Response: If your program depends on the code expansion, you should change your application so that it does not depend on the above statements.

Note: All environments of the database manager receive this message.

ARI5384E **Preprocessor parameter** parameter **duplicated.**

Explanation: The preprocessor option specified occurred more than once in the list of PREPPARM options.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Remove the duplicates of the parameter and rerun the preprocessor.

ARI5385E **Missing delimiter after preprocessor parameter** parameter**.**

Explanation: The preprocessor found a comma missing after the parameter specified. A character other than a comma was found.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Supply the missing comma and rerun the preprocessor.

ARI5386E clause **clause permitted with dynamic cursors only.**

Explanation: The FROM and USING clauses, in statements which reference cursors, are valid only if the cursor being referenced was declared dynamically. The statements that reference cursors are FETCH, OPEN, and PUT.

System Action: Preprocessing of the SQL statement ends.

User Response: Remove the clause specified from the statement and rerun the preprocessor.

ARI5387E **The SQL statement length exceeds 8192 positions.**

Explanation: The maximum length of an SQL statement is 8192 bytes. This statement exceeds the maximum. The possible causes are:

- The SQL statement is too long.
- The end of statement delimiter is missing.
- A FOR UPDATE OF clause is added to the end of a static cursor query statement.
- The expansion of a host structure reference has increased the size of the SQL statement.

System Action: Preprocessing of the SQL statement ends.

User Response: Either break the SQL statement into smaller statements or remove unnecessary conditions that may exist in the statement.

ARI5390E *statement-type* **statement not referencing** *cursor-type* **cursor.**

Explanation: The FETCH, DELETE, and UPDATE statements must reference a SELECT cursor. PUT statements must reference an INSERT cursor. The *statement-type* identified in the message references a cursor with an incorrect *cursor-type*.

System Action: Preprocessing of the SQL statement ends.

User Response: Change the cursor reference in the FETCH or PUT statement to reference the correct cursor and rerun the preprocessor.

ARI5391E **Preprocessor module** modname **encountered an internal error. Internal Error Code =** code-num

Explanation: The preprocessor module specified failed due to an internal program problem.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: Make a record of the problem and notify your system programmer.

System Programmer Response: Document the problem and contact the designated support group for your installation.

ARI5392E **An error occurred while retrieving the** *value* **value due to a system error.**

Explanation: A severe error occurred while the preprocessor retrieved the *value* value. For DBCS value, the value is retrieved from the LASTING GLOBALV file.

System Action: Processing terminated and the package was not created.

User Response: Refer to message ARI0503E for an error description.

ARI5394E **Statement** statement-name **referenced in a previous DECLARE CURSOR.**

Explanation: The statement identified by *statement-name* was referenced on a previous DECLARE CURSOR statement. A statement can be referenced on one DECLARE CURSOR statement only.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Either correct the statement reference or PREPARE another statement to use with this DECLARE CURSOR. Rerun the preprocessor.

ARI5395E **String beginning with** *string* **is invalid or incorrectly placed in this statement.**

Explanation: The portion of the SQL statement that has or begins with the string *string* is incorrect. Possible causes are:

- A new SQL statement is found (if *string* is EXEC SQL) before a previous SQL statement ends.
- A null string was required, at the end of a statement, but *string* was found.
- An SQL keyword was required (for example, CURSOR), but *string* was found.
- An object identifier was required (for example, statement name), but *string* was found.
- A numeric value was required (for example, CCSID value), but *string* was found. Possibly a host variable is required, but *string* was found.
- A host variable is required, but *string* was found.
- A delimiter character was required (for example, comma), but *string* was found.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Remove or correct the invalid *string* and rerun the preprocessor.

ARI5396E **The preprocessor options file is too large.**

Explanation: The preprocessor options file contains too many parameters. This situation occurs when parameters are duplicated or when parameters that are not permitted in the preprocessor options file are found.

System Action: Preprocessing ends. A new package is not produced. If the package existed previously, it is restored.

User Response: If parameters are duplicated remove the duplicates. If parameters that are not permitted in the preprocessor options file (for example, PREPNAME) have been included, remove them. Rerun the preprocessor.

ARI5397E **Statement** *statement-name* **is not valid because an error occurred while processing the associated DECLARE CURSOR statement.**

Explanation: During processing of the DECLARE CURSOR statement, an error occurred while storing the statement in the package. Because the DECLARE CURSOR and the PREPARE statements are related, processing of the first statement in the pair must be successful before continuing the processing of the second statement.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Correct the problem in the DECLARE CURSOR statement and rerun the preprocessor.

ARI5398E **End of string found before the SQL statement ended. Statement is incomplete.**

Explanation: The end of the SQL statement string was reached before all required segments of the statement were found. This may be caused by an incorrectly placed statement delimiter or by the presence of an open-ended literal in the statement.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Supply the required missing statement segments or correct the problem with the string or statement delimiter. Rerun the preprocessor.

ARI5399I **The preprocessor found errors. [Return Code 8 was changed to 4 | The error statements were accepted] because the preprocessor parameter, ERROR, was specified.**

Explanation: When the preprocessor parameter ERROR is specified, the preprocessor tolerates most statement-parsing errors. The error message is inserted into the modified source and listing files in comment form, and processing continues. The program can be compiled and executed, but the erroneous statement cannot be executed on the application server that issued the error. See the *DB2 Server for VSE & VM Application Programming* manual for a discussion of the ERROR parameter.

System Action: Preprocessing continues. The modified source code is generated and package functions are performed.

User Response: If you intend to execute the erroneous statement on the application server that issued the error, correct the statement and rerun the preprocessor. If you do not intend to execute the statement on that application server, ignore the message.

ARI5400I **Warning:** *statement-type* **is not supported.**

Explanation: The SQL statement shown (*statement-type*) is not supported by the database manager. Examples of *statement-type* values are DECLARE TABLE, DECLARE STATEMENT, etc. In other IBM SQL products, these statements cause code expansions to occur. As the database manager does not support these statements, these code expansions have not been done by the preprocessors. However, your program will be preprocessed successfully unless errors occur and you did not specify ERROR as a preprocessing parameter.

System Action: Preprocessing continues.

User Response: If your program depends on the code expansion, change your application so that it does not depend on either of the above statements.

ARI5401E **Cursor name,** cursorname**, was declared before this statement was prepared.**

Explanation: Your application has a DECLARE for a statement from a quoted literal. The PREPARE for this statement must appear in the application before the DECLARE CURSOR for this statement.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Change your program so that the PREPARE appears before the DECLARE cursor statement. Alternatively, you can assign the quoted literal to a host variable, and use the host variable in the PREPARE statement in place of the quoted literal. In this case, the DECLARE statement can precede the PREPARE statement.

ARI5402I **Preprocessor failed to retrieve the DBCS option. The DBCS option is set to NO.**

Explanation: A DB2 Server for VM error occurred when the preprocessor retrieved the DBCS option from the LASTING GLOBALV file.

System Action: Processing continued and DBCS defaulted to NO.

User Response: Check the for valid DBCS options. If the option does not exist, or is invalid, report this to the system administrator.

Programmer Response: Use SQLINIT QUERY to check that the DBCS option has been set.

ARI5403E **You cannot submit an SQL statement containing host variables by way of a PREPARE statement.**

Explanation: During the preprocessing of an SQL PREPARE statement, a host variable was found in the SQL statement being prepared. The SQL statement must not contain input or output host variables.

System Action: Preprocessing continues. A new package is not produced. If the package existed previously, it is restored.

User Response: Correct the SQL PREPARE statement.

ARI5404E **The preprocessor version and the application server version are incompatible. Preprocessing failed.**

Explanation: The release level of the preprocessor is incompatible with the target application server release level. Preprocessing is terminated and the portable package is not created.

System Action: None.

System Programmer Response: Make sure that the target application server release level supports the preprocessor release level, or preprocess the program to a application server at a compatible release level, and then unload or reload the package to the target application server.

ARI5406I The C preprocessor ignored all input after the semicolon in the SQL external INCLUDE statement.

Explanation: The preprocessor found input after the semicolon in the SQL external INCLUDE statement and on the same line as the statement.

System Action: The preprocessor ignored the input after the semicolon and did not copy it to the modified source file. Preprocessing continues.

User Response: If the ignored input is not a comment starting and ending on the same line, move everything after the semicolon to a separate line.

Chapter 11. ARI5500–ARI5593 FLAGGER Messages

The FLAGGER messages use both SQL and ANSI terms. The ANSI terms are defined below. For a definition of the SQL terms, refer to the *DB2 Server for VSE & VM SQL Reference* manual.

These terms are in uppercase letters in both this manual and the *DB2 Server for VSE & VM SQL Reference* manual.

ALL SET FUNCTION
ALL column function. Any column function where ALL is implicitly or explicitly specified. (In other words, duplicate values are not eliminated.)

AUTHORIZATION IDENTIFIER
The name of the owner or creator of an object, such as a table.

COLUMN DEFINITION
The definition of a column of a table.

COLUMN SPECIFICATION
Reference to a named column.

COMPARISON PREDICATE
A basic predicate. A predicate that compares two values.

DEGREE
The number of columns or expressions selected in an SQL statement.

DISTINCT SET FUNCTION
DISTINCT column function. Any column function with which DISTINCT has been specified. (In other words, duplicate values are to be eliminated.)

GROUPed view or GROUPed table
A view or table that is a result of a GROUP BY clause, a HAVING clause or both of these clauses. A view can also be GROUPed view if any view referenced in the FROM clause is a GROUPed view.

GROUPing column
A column referenced in a GROUP BY clause.

IDENTIFIER
An alphameric token used to form a name.

OUTER REFERENCE
A correlated reference.

QUERY EXPRESSION
A fullselect.

QUERY TERM
A subselect or a fullselect that is surrounded by parentheses.

QUERY SPECIFICATION
A subselect.

RESULT SPECIFICATION
The columns to be searched for in a subquery.

SEARCH CONDITION
One or more Boolean terms.

SELECT LIST
List of columns or expressions, or * specified in the SELECT statement.

SELECT TARGET LIST
One or more TARGET SPECIFICATIONs.

SET FUNCTION SPECIFICATION
One of the following column functions:

- COUNT (*)
- A column function with DISTINCT explicitly specified
- A column function with ALL implicitly or explicitly specified.

SORT SPECIFICATION
The clause comprised of column names, integers, ASC, and DESC that follows the keywords in an ORDER BY clause. For example, in the clause `ORDER BY COL1 DESC, COL1 DESC` would be the SORT SPECIFICATION.

TABLE EXPRESSION
A FROM clause optionally followed by a WHERE clause, a GROUP BY clause, or a HAVING clause.

TARGET SPECIFICATION
Specification of a parameter or variable that can be assigned a value.

VALUE EXPRESSION
A value specified as the value itself, through a variable name, or as an arithmetic expression.

© Copyright IBM Corp. 1987, 2000

VIEW COLUMN LIST
A list of column names provided on the CREATE VIEW statement.

ARI5500I **FLAGGER Message: The FIPS standard is used for syntax flagging.**

Explanation: Information message to identify the standard used for flagging during preprocessing.

System Action: Processing continues.

ARI5501I **FLAGGER Message: The SAA standard is used for syntax flagging.**

Explanation: Information message to identify the standard used for flagging during preprocessing.

System Action: Processing continues.

ARI5502I **FLAGGER Message: The statement deviates from the SQL standard syntax at the token starting with the text** text**.**

Explanation: A syntax deviation from the standard named in the earlier message (FIPS or SAA) was encountered in the SQL statement.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5503I **FLAGGER Message: A data definition statement is not within a CREATE SCHEMA statement.**

Explanation: The FIPS standard requires that all data definition statements are contained within a CREATE SCHEMA statement.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5504I **FLAGGER Message: There is a nonstandard embedded comment.**

Explanation: A comment embedded in an SQL statement does not meet the requirements of the standard being flagged:

- The SAA standard is being checked and the comment is not introduced as required by the host language.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5505I **FLAGGER Message: An SQL syntax deviation occurred. The statement is not complete.**

Explanation: The SQL statement terminated before all required elements were found.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5506I **FLAGGER Message: Identifier** *identifiername* **is more than 18 characters.**

Explanation: AUTHORIZATION IDENTIFIER, table identifier, column name, correlation name, module name, cursor name, procedure name, or parameter name has a length of more than 18 characters.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5507I **FLAGGER Message: Column** column **has an invalid length, precision, or scale attribute.**

Explanation: One of the following conditions has not been met:

- Length value should be greater than zero.
- Precision value should be greater than zero.
- Scale value should not be greater than precision.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5508I **FLAGGER Message: The data type of** *datatype* **should be an exact numeric with a scale of zero.**

Explanation: Data type of an indicator variable should be exact numeric with a scale of zero.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5509I **FLAGGER Message: The SET FUNCTION SPECIFICATION references column** column**.**

Explanation: One of the following conditions has not been met:

- The COLUMN SPECIFICATION of a DISTINCT SET FUNCTION cannot reference a column derived from a SET FUNCTION SPECIFICATION.

- COLUMN SPECIFICATIONs in the VALUE EXPRESSION of an ALL SET FUNCTION cannot reference a column derived from a SET FUNCTION SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5510I **FLAGGER Message: A VALUE EXPRESSION containing** column **cannot include operators.**

Explanation: A VALUE EXPRESSION cannot include operators in an OUTER REFERENCE COLUMN SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5511I **FLAGGER Message: The COLUMN SPECIFICATION is missing or is invalid for the ALL SET FUNCTION** function**.**

Explanation: The VALUE EXPRESSION of an ALL SET FUNCTION SPECIFICATION should include a COLUMN SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5512I **FLAGGER Message: Column** column **is not unique and requires qualification.**

Explanation: The specified column is not unique with the current scope. Qualification must be given to uniquely identify the required column.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5513I **FLAGGER Message: A VALUE EXPRESSION may not contain a SET FUNCTION SPECIFICATION.**

Explanation: The VALUE EXPRESSION of an ALL SET FUNCTION must not contain a SET FUNCTION SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5514I **FLAGGER Message: Column** column **does not identify a column of a table within the current scope.**

Explanation: One of the following conditions has not been met:

- The table or correlation name used as a qualifier does not exist.
- The column name does not exist within the current scope or the scope of the qualifier.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5515I **FLAGGER Message: A column function that contains the OUTER REFERENCEd column** column **is not in the subquery of a HAVING clause.**

Explanation: A column function must be contained in a subquery of a HAVING clause if it contains an OUTER REFERENCE COLUMN SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5516I **FLAGGER Message: The result of a SUM or AVG function cannot be a character string.**

Explanation: Character string is not valid for the result of a SUM or AVG SET FUNCTION SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5517I **FLAGGER Message: Operator** operator **is invalid in this context.**

Explanation: One of the following conditions has not been met:

- A VALUE EXPRESSION containing a DISTINCT SET FUNCTION cannot include dyadic operators.
- The first character following a monadic operator cannot be a plus or minus sign.
- A VALUE EXPRESSION should not contain operators if primary is of type character string.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5518I **FLAGGER Message: The** exptype **EXPRESSION is comparing incompatible data types.**

Explanation: The data types of one of the following (identified by *exptype*) do not match:

- *exptype* = **COMPARISON** - Comparison operators must match.
- *exptype*= **BETWEEN** - The three VALUE EXPRESSIONs must match.
- *exptype* = **IN** - VALUE EXPRESSION, subquery and all VALUE SPECIFICATIONs must match.
- *exptype* = **QUANTIFIED** - VALUE EXPRESSION and subquery must match.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5519I **FLAGGER Message: The operand in the LIKE predicate is not a character string.**

Explanation: One of the following conditions has not been met:

- Data type of the column in a LIKE predicate is not a character string.
- Data type of the pattern in a LIKE predicate is not a character string.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5520I **FLAGGER Message: The ESCAPE character must be a one-byte character string.**

Explanation: Escape character in a LIKE predicate must have the data type of character string with the length of 1.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5521I **FLAGGER Message: The WHERE clause, GROUP BY clause, or HAVING clause is invalid for the GROUPed view** view**.**

Explanation: If the table identified in the FROM clause is a GROUPed view, then the TABLE EXPRESSION must not contain a WHERE clause, GROUP BY clause, or HAVING clause.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5522I **FLAGGER Message:** Name **occurs more than once in a FROM clause.**

Explanation: One of the following conditions has not been met:

- Table name occurs more than once in a FROM clause.
- Correlation name is identical to a table name or to another correlation name in a FROM clause.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5523I **FLAGGER Message: Only one table reference is allowed in the FROM clause for a GROUPed view.**

Explanation: If the table identified by table name is a GROUPed view, then the FROM clause must contain exactly one table reference.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5524I **FLAGGER Message: A reference to column** column **derived from a column function is invalid in a WHERE clause.**

Explanation: A VALUE EXPRESSION directly contained in the SEARCH CONDITION of a WHERE clause must not include a reference to a column derived from a column function.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5525I **FLAGGER Message: A HAVING clause must contain the WHERE clause when the WHERE clause has a column function.**

Explanation: If a VALUE EXPRESSION directly contained in the SEARCH CONDITION is a column function, then the WHERE clause must be contained in a HAVING clause.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5526I **FLAGGER Message: The COLUMN SPECIFICATION for** column **must be an OUTER REFERENCE.**

Explanation: If a VALUE EXPRESSION directly contained in the SEARCH CONDITION is a function, then the COLUMN SPECIFICATION in the column

function expression must be an OUTER REFERENCE.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5527I **FLAGGER Message: Column** column **is duplicated within the current scope.**

Explanation: The specified column is duplicated within the current scope.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5528I **FLAGGER Message: The COLUMN SPECIFICATION for** table **should be a GROUPing column or should be specified within a column function.**

Explanation: Each COLUMN SPECIFICATION contained in a subquery in the SEARCH CONDITION of a HAVING clause must reference a GROUPing column or be specified within a column function.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5529I **FLAGGER Message: DEGREE of table** table **should be 1 when using a RESULT SPECIFICATION of *.**

Explanation: The DEGREE of the TABLE EXPRESSION must be 1 if the RESULT SPECIFICATION * is specified in a subquery of any predicate other than an EXISTS predicate.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5530I **FLAGGER Message: The column function is not valid for the TABLE EXPRESSION beginning with table** table**.**

Explanation: One of the following conditions has not been met:

- If the TABLE EXPRESSION is a GROUPed view, the RESULT SPECIFICATION of a subquery cannot contain a SET FUNCTION SPECIFICATION.
- If the TABLE EXPRESSION is a GROUPed view, the SELECT LIST of a QUERY SPECIFICATION cannot contain a column function.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5531I **FLAGGER Message: The COLUMN SPECIFICATION for** column **is invalid.**

Explanation: One of the following conditions has not been met:

- For a GROUPed table, the COLUMN SPECIFICATION must reference a GROUPing column or be specified within a SET FUNCTION SPECIFICATION.
- If not a GROUPed table and VALUE EXPRESSION includes a SET FUNCTION SPECIFICATION, then each COLUMN SPECIFICATION must be specified within a SET FUNCTION SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5532I **FLAGGER Message: DISTINCT is specified more than once.**

Explanation: One of the following conditions has not been met:

- DISTINCT must not be specified more than once in a QUERY SPECIFICATION, excluding any subquery of that QUERY SPECIFICATION.
- DISTINCT must not be specified more than once in a subquery, excluding any subquery contained in that subquery.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5533I **FLAGGER Message: The COMPARISON PREDICATE subquery cannot contain a GROUP BY or HAVING clause.**

Explanation: If a subquery is specified in a COMPARISON PREDICATE, then the TABLE EXPRESSION whose FROM clause identifies the named table must not contain a GROUP BY clause or a HAVING clause.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5534I **FLAGGER Message: The COMPARISON PREDICATE subquery cannot identify a GROUPed view.**

Explanation: If a subquery is specified in a COMPARISON PREDICATE, then the TABLE EXPRESSION whose FROM clause identifies the named table must not identify a GROUPed view.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5535I **FLAGGER Message: Invalid AUTHORIZATION IDENTIFIER** authid**.**

Explanation: The AUTHORIZATION IDENTIFIER prefixed to the table name is invalid.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5536I **FLAGGER Message: Table or view** name **already exists.**

Explanation: The table *name* or view *name* specified already exists in the catalogue.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5537I **FLAGGER Message: The COLUMN DEFINITION is missing.**

Explanation: At least one COLUMN DEFINITION should be specified for a CREATE TABLE.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5538I **FLAGGER Message: The data type** type1 **of the target is not compatible with the data type** type2 **of the source.**

Explanation: Data types should match between both the target and source:
- In a FETCH statement
- In a SELECT statement.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5539I **FLAGGER Message: You must specify a VIEW COLUMN LIST for** table**.**

Explanation: If any two columns in the table specified by the QUERY SPECIFICATION have the same column name, or if any column of that table is an unnamed column, then a VIEW COLUMN LIST must be specified.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5540I **Warning: There is not enough memory for the FLAGGER. Flagging is suspended for this statement.**

Explanation: The FLAGGER has run out of stack space.

System Action: Flagging continued for subsequent statements.

User Response: Notify the system programmer.

System Programmer Response: Check whether the memory allocation for the preprocessor can be expanded. Allocate more memory for the FLAGGER operation if possible. Refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for more information on failure analysis procedures.

ARI5541I **Warning: An error occurred when attempting to load the FLAGGER** modulename**. Flagging capability is affected.**

Explanation: The FLAGGER data or load module used to describe the syntax for the standard being flagged was not found during FLAGGER initialization.

System Action: If the module that failed to load was a syntax module (ARIFF00 or ARIFS00), then flagging is discontinued. If the module was a semantics module, then semantics checking stops, and only syntax checking continues.

User Response: Notify the system programmer.

System Programmer Response: Refer to the message produced by the operating system for details of causes and proceed as instructed for that message. Refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for more information on failure analysis procedures.

ARI5542I **Warning: An error occurred when initializing the FLAGGER. Flagging is discontinued.**

Explanation: An internal error occurred during FLAGGER initialization.

System Action: Preprocessing continues but FLAGGER operation is discontinued.

User Response: Notify the system programmer.

System Programmer Response: Refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for more information on failure analysis procedures.

ARI5543I **Warning: The FLAGGER could not process the SQL statement. Flagging is discontinued.**

Explanation: An SQL statement passed to the FLAGGER caused an internal error within it.

System Action: Preprocessing continues but FLAGGER operation is discontinued.

User Response: Notify the system programmer.

System Programmer Response: Refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for more information on failure analysis procedures.

ARI5544I **Warning: An error occurred when terminating the FLAGGER.**

Explanation: An internal error occurred during FLAGGER termination.

System Action: Preprocessing continues but FLAGGER operation is discontinued.

User Response: Notify the system programmer.

System Programmer Response: Refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for more information on failure analysis procedures.

ARI5545I **Warning: Error during FLAGGER access to the system catalog. Flagging continues to check syntax only.** **SQLCODE =** nnn **SQLERRP =** modname **SQLERRD1 =** nnn **Creator =** creatorname **Table =** tablename

Explanation: An internal error occurred during FLAGGER access to the system catalog.

System Action: Flagging continues for syntax checking only.

User Response: Notify your system programmer.

System Programmer Response: Refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for more information on failure analysis procedures.

ARI5546I **Warning: An error occurred when processing FLAGGER internal memory blocks.**

Explanation: The FLAGGER could not find an internal memory block.

System Action: Preprocessing continues but FLAGGER operation is discontinued.

User Response: Notify the system programmer.

System Programmer Response: Refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for more information on failure analysis procedures.

ARI5547I **Warning: Not enough memory for FLAGGER operation.**

Return Code = nn**.**

Explanation: The FLAGGER does not have enough memory available to operate.

System Action: Preprocessing continues but FLAGGER operation is discontinued.

User Response: Notify the system programmer.

System Programmer Response: Refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for more information on failure analysis procedures.

ARI5548I **Warning: Error releasing FLAGGER memory during termination.**

Explanation: The FLAGGER could not release memory back to the operating system.

System Action: Flagging termination stops.

User Response: Notify the system programmer.

System Programmer Response: Refer to the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual for more information on failure analysis procedures.

ARI5549I **Warning: You requested semantics flagging while using the DRDA protocol. The SQLFLAG option is ignored.**

Explanation: When preprocessing under DRDA protocol, only syntax flagging is available.

System Action: Preprocessing continues without FLAGGER operation.

User Response: If semantics flagging is required, preprocess in a non-DRDA environment. Refer to the *DB2 Server for VSE & VM Database Administration* manual for a discussion of protocol alternatives for the SQLINIT EXEC.

ARI5570I **FLAGGER Message: You must declare column** column **as NOT NULL.**

Explanation: All columns identified as UNIQUE must be defined with the NOT NULL option.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5571I **FLAGGER Message: The viewed table** table **must be updateable.**

Explanation: When the WITH CHECK OPTION is specified then the viewed table must be updateable.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5572I **FLAGGER Message: Invalid number of column names.**

Explanation: The number of column names in the VIEW COLUMN LIST should be the same as the DEGREE of the table specified by the QUERY SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5573I **FLAGGER Message: You must declare cursor** cursor **before using it.**

Explanation: The cursor specified has not been declared in a DECLARE CURSOR statement.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5574I **FLAGGER Message: Cursor** cursor **is already declared.**

Explanation: The cursor specified has already been declared in a DECLARE CURSOR statement.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5575I **FLAGGER Message: Only * or column names are valid in this context.**

Explanation: When UNION is specified, the SELECT LIST for the specification of the two TABLE EXPRESSIONs identified by the QUERY EXPRESSION and QUERY TERM should consist of * or COLUMN SPECIFICATIONs.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5576I **FLAGGER Message: The descriptions of tables identified by the QUERY EXPRESSION beginning with table** table1 **and the QUERY TERM beginning with** table2 **should be identical.**

Explanation: When UNION is specified, the descriptions of the two tables should be identical, except for the column names.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5577I **FLAGGER Message: The SORT SPECIFICATION** number **is outside the DEGREE of cursor** cursor**.**

Explanation: If the SORT SPECIFICATION contains an unsigned integer, it should be greater than `0` and not greater than the number of columns in the table.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5578I **FLAGGER Message: Table** table **is a read-only table.**

Explanation: A DELETE, an INSERT, or an UPDATE statement was specified for a read-only table.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5579I **FLAGGER Message: Table** table **must not be identified in a FROM clause of any subquery contained in a SEARCH CONDITION.**

Explanation: A table specified in a DELETE or an UPDATE cannot be used in a FROM clause of any subquery contained in its SEARCH CONDITION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5580I **FLAGGER Message: Table** table1 **is not the first table** table2 **specified in the DECLARE CURSOR statement.**

Explanation: The table specified in a DELETE or an UPDATE statement should be the first table specified in the FROM clause of the DECLARE CURSOR statement.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5581I **FLAGGER Message: The number of TARGET SPECIFICATIONs does not match the DEGREE of cursor** cursor**.**

Explanation: The number of TARGET SPECIFICATIONs in the FETCH statement does not match the DEGREE of the table specified.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5582I **FLAGGER Message: The target table** table **of the INSERT statement is also in the FROM clause or in the subquery.**

Explanation: The named table must not be identified in a FROM clause of a QUERY SPECIFICATION or of any subquery contained in the QUERY SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5583I **FLAGGER Message: The number of columns specified does not match the number of values specified.**

Explanation: In an INSERT statement, the number of columns specified does not match the number of values specified.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5584I **FLAGGER Message: The number of columns specified does not match the DEGREE of the QUERY SPECIFICATION beginning with table** table**.**

Explanation: In an INSERT statement, the number of columns specified does not match the DEGREE of the table specified in the QUERY SPECIFICATION.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5585I **FLAGGER Message: There is a data type or length mismatch between column** column **and the INSERT or UPDATE item.**

Explanation: One of the following conditions has not been met:

- If the data type of the column name is a character string, then the corresponding item in the INSERT or UPDATE statement should be a character string of length equal to or less than the length of the column name.
- If the data type of the column name is exact numeric, then the corresponding item in the INSERT or UPDATE statement should be exact numeric.
- If the data type of the column name is approximate numeric, then the corresponding item in the INSERT or UPDATE statement should be approximate numeric or exact numeric.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5586I **FLAGGER Message: You cannot use a GROUP BY or HAVING clause or identify a GROUPed view in this context.**

Explanation: The table identified in the FROM clause of the TABLE EXPRESSION in a SELECT statement should not include a GROUP BY or HAVING clause and should not identify a GROUPed view.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5587I **FLAGGER Message: The number of elements specified in the SELECT LIST should match the number in the SELECT TARGET LIST.**

Explanation: In the SELECT statement, the number of elements specified in the SELECT LIST should match the number of elements in the SELECT TARGET LIST.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5588I **FLAGGER Message: Column function** function **is not allowed in an UPDATE statement.**

Explanation: A VALUE EXPRESSION in a SET clause of an UPDATE statement should not include a column function.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5589I **FLAGGER Message: You cannot specify NULL for NOT NULL column** column**.**

Explanation: If NULL is specified in the SET clause of the UPDATE statement, then the corresponding column should allow nulls.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5590I **FLAGGER Message: Invalid SQL terminator for host language.**

Explanation: The SQL statement was not terminated by the correct construct for the host language being processed.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5591I **FLAGGER Message: You must define the host variable** identifier**.**

Explanation: The host variable named was either:
- Not defined within the program
- Defined outside the SQL Declare Section.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5592I **FLAGGER Message: Table** table **not found in the catalog.**

Explanation: The named table or view does not exist in the system catalog.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

ARI5593I **FLAGGER Message: For the host language** language**, there is an invalid IDENTIFIER or LABEL** name**.**

Explanation: The format of the IDENTIFIER *name* or LABEL *name* displayed does not meet the requirements of the host language.

System Action: Processing continues.

User Response: Correct the SQL statement to meet installation standards.

Chapter 12. ARI0600–ARI0699, ARI6000–ARI6702, ARI0700–ARI0799 CMS EXEC Messages

ARI0600A **Enter** &1**.**

Explanation: SQLGENLD is prompting for saved segment ID.

System Action: The system will read the user-entered saved segment ID for SQLGENLD. If it is a valid CMS identifier, the system will use the value. If it is not a valid CMS identifier, an error message is issued and this message is displayed again.

User Response: Enter the requested value.

ARI0601E **You are missing parentheses for keyword** &1**.**

Explanation: The keyword (&1) was entered on the invocation, but the parameter value associated with the keyword was not enclosed in parentheses. A left or right parenthesis is missing.

System Action: The EXEC ends.

User Response: Re-invoke the EXEC, using the right parenthesis with the value for the indicated keyword.

ARI0602D **Saved segment ID** &1 &2 &3 **already exists. Do you want to continue? Enter 0(No) or 1(Yes).**

Explanation: A bootstrap module with the same name as that specified by the user was found on the DB2 Server for VM production minidisk. You have the option of continuing (and thereby replacing the existing copy) or terminating.

&1 is the saved segment ID value. &1 &2 &3 are the file name, file type, and file mode of the bootstrap module. Values for &2 are:

SQLDBBT
RDS and DBSS/DSC bootstrap

SQLISBT
ISQL bootstrap

SQLRMBT
Resource Adapter bootstrap

System Action: If the reply is 1, processing continues and the existing bootstrap is replaced. If the reply is 0, processing terminates. If the reply is neither 1 nor 0, this message is displayed again.

User Response: Reply either 0(NO) or 1(YES) to the message.

ARI0603I **The database was generated with** &1 **internal DBSPACES.**

Explanation: The number of internal DBSPACES you request to be added will be added to &1, the number specified during database generation.

System Action: Message ARI0648A is issued and processing continues.

ARI0604A **Enter a 1- to 8-byte** &1 &2 **saved segment name.**

Explanation: The requested name will be passed to the bootstrap for use in loading code which will reside in a VM saved segment. Values substituted for &1 and &2 will identify the bootstrap being generated, as follows:

```
&1        &2
--------  --------
RESOURCE  ADAPTER
DSC/DBSS
RDS
ISQL
```

The saved segment name must be the same name as was specified in the SYSNAME parameter of the ARISAVES EXEC when the saved segment was generated.

System Action: If the reply is a valid CMS identifier, processing continues. If the reply is not a valid CMS identifier, an error message is issued, and this message is displayed again.

User Response: Enter the requested saved segment name.

ARI0605E **Invalid response.**

Explanation: The value you entered contains one or more characters which are not valid in a CMS identifier.

System Action: The message for which the incorrect reply was entered is displayed again.

User Response: Enter the requested value.

ARI0606E **Error in** &1 **bootstrap definition process. Return Code =** &2**.**

Explanation: An error occurred while starting a bootstrap skeleton (&1). The return code (&2) from the CMS START command is given. Return code values can be:

© Copyright IBM Corp. 1987, 2000

24 During SQLGENLD EXEC processing, the bootstrap was invoked with an invalid tokenized parameter list.

256 During SQLGENLD EXEC processing, the bootstrap was invoked with an invalid name (one other than ARISDBK, ARISISK, or ARISRMK).

3xxx An error occurred during SQLGENLD EXEC or SQLINIT EXEC processing. The return code displayed is the return code from the CMS GENMOD command, with 3000 added by the database manager. The error occurred while attempting to create the SQLDBBT, SQLISBT, or SQLRMBT MODULE file.

System Action: Processing ends.

User Response: Follow your installation's problem determination procedures.

ARI0607E **An error occurred while the database manager was attempting to copy** &1 &2 &3 **to** &4 &5 &6**. Return Code =** &7**.**

Explanation: An error occurred while trying to copy the named CMS file. (&1 &2 &3 are the old file name, file type, and file mode; &4 &5 &6 are the new file name, file type, and file mode.) The return code from the CMS COPYFILE command is given (&7). The meaning of the return code is given in the *VM/ESA: CMS Command Reference* manual for your system.

System Action: Processing ends.

User Response: Follow your installation's problem determination procedure.

ARI0608E **An error occurred while the database manager was attempting to erase** &1 **MODULE A. Return Code =** &2**.**

Explanation: An error occurred while trying to erase the named CMS file. (&1 is the file name.) The return code (&2) from the CMS ERASE command is given. The meaning of the return code is given in the *VM/ESA: CMS Command Reference* manual for your system.

System Action: Processing ends.

User Response: Follow your installation's problem determination procedure.

ARI0609E **An error occurred in** &1 **while the database manager was attempting to use the** &2 **saved segment. Return Code =** &3**.**

Explanation:

(&1 = ARISDBBT)
An error occurred while the ARISDBBT bootstrap was trying to load the RDS or DSC/DBSS (.&2) saved segment.

(&1 = ARISISBT)
An error occurred while the ARISISBT bootstrap was trying to load the ISQL (&2) saved segment. The return code (&3) can be:

4500 (Issued from the SQLSTART EXEC) - In attempting to load RDS as a saved segment, it was found that the RDS segment would overlay the DSC/DBSS segment.

4600 (Issued from the SQLSTART EXEC or ISQL EXEC) - In attempting to load RDS or DSC/DBSS or ISQL (indicated by &2 in the message) as a saved segment, the storage required to load the saved segment is already in use. Either the required virtual machine storage has already been allocated by CMS or storage outside the virtual machine has been reserved for another saved segment.

4xxx and 5xxx
(For situations other than 4500 or 4600, from the SQLSTART EXEC or ISQL EXEC) - The return code displayed is the return code from the SEGMENT LOAD macro with 4000 added by the database manager. The error occurred while attempting to load RDS or DSC/DBSS or ISQL (indicated by &2 in the message) as a saved segment.

System Action: Processing ends.

User Response: Follow your installation's problem determination procedure.

ARI0610D &1 **is not a valid CUU for this Virtual Machine. Please reenter the CUU.**

Explanation: The cuu (&1) which was entered during database generation is not a valid virtual device address or is not defined for your virtual machine.

System Action: The system waits for you to enter a valid virtual device address.

User Response: Enter a valid virtual device address.

ARI0611E **An untokenized parameter list was not supplied.**

Explanation: The ARISDBBT bootstrap was invoked without an untokenized parameter list. The bootstrap was probably invoked from an EXEC written in a language which does not provide an untokenized parameter list or from a program which did not provide an untokenized parameter list.

System Action: Processing ends.

User Response: Refer this message to your system programmer.

System Programmer Response: If the bootstrap was

invoked from the SQLSTART EXEC, contact the designated support group for your installation. In any other case, correct the invocation of the bootstrap and retry.

ARI0612E **An error occurred in** &1 **while the database manager was attempting to load** &2 **into the free storage area. Return Code =** &3**.**

Explanation: DB2 Server for VM processing was not successful. An error occurred while attempting to load a DB2 Server for VM module into the CMS free storage area.

&1 CMS command or DB2 Server for VM module being executed when the error occurred.

&2 DB2 Server for VM component or load library member name being loaded into free storage.

&3 Return code from the command or module identified by &1. This return code describes the specific free storage load error that occurred. The return code can be:

- 6500 (from the SQLSTART EXEC) - After a successful execution of the CMS NUCXLOAD command for RDS or DSC/DBSS (&2), the CMS NUCEXT QUERY function could not find the code.
- 6xxx (other than 6500, from the SQLSTART EXEC) - The return code displayed is the return code from the CMS NUCXLOAD command with 6000 added by the database manager. The error occurred while attempting to load RDS or DSC/DBSS (&2) into the CMS free storage area as a nucleus extension.
- xxx (from the SQLRMEND EXEC) - The return code displayed is the return code from the CMS NUCXLOAD command. The error occurred while attempting to load ARISRRM into the CMS free storage storage area as a nucleus extension.

System Action: Processing ends.

User Response: Correct the error indicated and rerun the DB2 Server for VM command.

ARI0613E **You did not specify the required EXEC parameter** &1**.**

Explanation: The named keyword parameter (&1) is needed for correct execution of the EXEC.

System Action: Processing ends.

User Response: Reenter the command with the parameter specified.

ARI0614A **Enter virtual address and storage pool number (default = 1) of DBEXTENT** extno**.**

Explanation: During SQLADBEX processing for ADD DBEXTENT, the database manager requires the minidisk virtual address (cuu) for the new DBEXTENT. You can also specify the storage pool number. If you do not give the pool number, the system uses the default (pool 1).

System Action: The system waits for you to enter the required parameters.

User Response: Enter the virtual address and, optionally, the storage pool number.

ARI0615D **Is this acceptable? Enter 0(No) or 1(Yes).**

Explanation: The previous message presented some information for you to evaluate and reply whether the information is acceptable.

System Action: The action depends on the EXEC in use:

- For the SQLINIT EXEC: If reply is Y, processing is complete. If reply is N, the system prompts you for the correct information. If reply is neither Y nor N, the system displays this message again.
- For the SQLLOG EXEC: If reply is Y, processing continues. If reply is N, processing ends.

User Response: Decide whether the information is acceptable, and reply Y or N.

ARI0616E **An error occurred in** &1 **command for DDNAME of** &2**. Return Code =** &3**.**

Explanation: An error occurred during the processing of a CMS FILEDEF or LABELDEF command (&1). &3 is the return code from the FILEDEF or LABELDEF command.

System Action: Processing ends.

User Response: Refer this problem to your system programmer.

System Programmer Response: Follow your installation's problem determination procedures.

ARI0617E **Error in** &1 **bootstrap load module. Return Code =** &2**.**

Explanation: An error occurred in the ARISDBBT bootstrap, which loads and invokes the database machine code. The return code from the bootstrap is given:

255 - The bootstrap module ARISDBBT was invoked with a command name other than ARISDBBT.

1xxx (from the SQLSTART EXEC) - The return code displayed is the return code from the CMS free storage

macro with 1000 added by the database manager. The error occurred while attempting to get working storage for the bootstrap whose name is given by the &1 value in the message.

System Action: Processing ends.

User Response: Refer this problem to your system programmer.

System Programmer Response: For return code 255, an error exists in the logic of the SQLSTART EXEC. Contact the designated support group for your installation. For return code 1xxx, take appropriate action based on the CMS free storage macro return code.

ARI0618I **Initialization for the use of DB2 Server for VM completed successfully.**

Explanation: SQLINIT has ended normally. Bootstraps have been made ready in the user's machine for using the database manager.

System Action: Processing ends.

ARI0619E **You may not specify a saved segment ID unless you also specify a server name.**

Explanation: SQLINIT was invoked with a saved segment ID parameter but without a DBNAME parameter. This combination is not allowed.

System Action: Processing ends.

User Response: Reenter the SQLINIT EXEC, giving a DBNAME parameter or omitting the saved segment ID parameter.

ARI0620I &1 &2 **file successfully copied to production disk.**

Explanation: The named file (&1 &2) was successfully copied to the DB2 Server for VM production minidisk.

System Action: Processing continues.

ARI0621E filename **cannot continue because all access mode letters are in use.**

Explanation: The SQLDBGEN, SQLADBEX, and SQLLOG EXECs require you to use an access mode letter. The SQLCDBEX EXEC requires you to use two access mode letters. All mode letters are now in use. *filename* identifies the DB2 Server for VM EXEC.

System Action: Processing ends.

User Response: Enter a CMS RELEASE command for one or more minidisks. Reenter the DB2 Server for VM EXEC you were trying to run.

ARI0622I **The database manager could not obtain bootstrap module** &1 &2 &3**. The return code from the COPYFILE command was** &4**.** &5 &2 &3 **will be used instead.**

Explanation: An attempt was made to copy the named bootstrap (&1 &2 &3). The attempt was not successful; the return code (&4) from the CMS COPYFILE command is given. The last line of the message names the bootstrap (&5 &2 &3) which was used instead.

System Action: Processing continues.

ARI0623D **Do you want to do a CMS FORMAT command on the database disks? Enter 0(No), 1(Yes), or 111(quit).**

Explanation: During SQLDBGEN, you replied that you want to use the existing dbname SQLFDEF file. You are now being asked whether CMS FORMAT and RESERVE commands should be issued for the database minidisks described in the dbname SQLFDEF file.

System Action: If your reply is 1 for YES, the SQLFDEF file is read, and CMS FORMAT and RESERVE commands are issued for all database minidisks.

User Response: Decide whether you want to format the database minidisks. All database minidisks must be formatted and reserved before they can be used. Enter your reply.

ARI0624A **Saved segment ID** &1 **contains an invalid character. Please reenter the saved segment ID.**

Explanation: The saved segment ID you specified contains one or more characters which are not valid in a CMS identifier.

System Action: The system waits for you to enter a correct saved segment ID.

User Response: Enter a valid saved segment ID.

ARI0625D **Do you want to create a bootstrap for the** &1 &2 **? Enter 0(No) or 1(Yes).**

Explanation: SQLGENLD lets you generate one or more of the following bootstraps (&1 &2), as follows:

```
&1              &2
------------    --------
RESOURCE        ADAPTER
DSC/DBSS/RDS
ISQL
```

System Action: Any reply other than 1 or 0 causes the system to display the message again.

User Response: Decide whether you want the named bootstrap to be created. Reply 1 or 0.

ARI0626E **An error occurred from the LOAD command while attempting to load** &1**. Return Code =** &2**.**

Explanation: The SQLGENLD EXEC or the SQLINIT EXEC received a nonzero return code (&2) from the CMS LOAD command while trying to load the named bootstrap text file (&1). The meaning of the return code is given in the *VM/ESA: CMS Command Reference* manual for your system.

System Action: Processing ends.

User Response: Follow your installation's problem determination procedures.

ARI0627E **The left parenthesis at position** &1 **does not follow a keyword.**

Explanation: During the syntax checking of the command, a left parenthesis was found which does not follow a keyword. The position of the parenthesis (&1) in the command argument string is given. (For example, 1 is the first position of the first parameter.)

System Action: Processing ends.

User Response: Reenter the command with correct syntax.

ARI0628E &1 **is an invalid keyword.**

Explanation: The specified keyword is not valid.

System Action: Processing ends.

User Response: Reenter the command with a correct keyword.

ARI0629E **Keyword** &1 **is invalid or has a value that is invalid or a value not properly enclosed within parentheses.**

Explanation: The specified keyword is either not valid, or it needs a parameter value.

System Action: Processing ends.

User Response: Reenter the command with a correct keyword and parameter value, if needed.

ARI0630E **Unable to obtain** &1 &2 **bootstrap module.**

Explanation: During SQLINIT, the EXEC did not find any bootstrap module identified by *&1 &2* on the DB2 Server for VM production minidisk. The error could be caused by a saved segment ID that was not spelled correctly. The error could also occur if the SQLGENLD EXEC has not yet been run for that saved segment ID, or if no DB2 Server for VM default bootstraps exist.

System Action: Processing ends.

User Response: Check the spelling of the saved segment ID parameter.

- If it is not correct, correct the spelling and rerun the EXEC.
- If the spelling is correct or if no saved segment ID was specified, contact your system programmer to verify that the SQLGENLD EXEC has been run. After the SQLGENLD EXEC has been run, access the DB2 Server for VM production minidisk again and then rerun the EXEC. If the error still occurs, follow your installation's problem determination procedures.

ARI0631E **Unable to obtain** &1 **access to the DB2 Server for VM** &2 **minidisk.**

Explanation: DB2 Server for VM processing was not successful. An error occurred while attempting to acquire read or write access to the DB2 Server for VM production minidisk or service minidisk.

&1 READ or WRITE access.

&2 PRODUCTION or SERVICE minidisk.

System Action: Processing ends.

User Response: Refer this problem to your system programmer.

System Programmer Response: Follow your installation's problem determination procedures.

ARI0632D **Database generation has previously been done for** &1 **because** &1 &2 &3 **already exists. You can use or replace this file. Enter USE to use it, or REPLACE to replace it.**

Explanation: SQLDBGEN found the indicated file (&1 &2 &3) on the DB2 Server for VM production minidisk. This file can be used for the database generation or a new version can be created. &1 is the database name.

System Action: Processing continues after you reply either USE or REPLACE.

User Response: Decide whether you want to use the existing file or replace it. Enter your reply.

ARI0633A **Please enter the** &1 **of the** &2 &3 &4 &5 &6 **disk.**

Explanation: SQLDBGEN is prompting you for the virtual device address (cuu) of a database minidisk (described by &2 through &6).

System Action: Processing continues after you enter the requested value.

User Response: Enter the needed value.

ARI0634A &1 **has previously been used as the CUU for the** &2**. Please reenter the CUU.**

Explanation: The virtual device address you entered (&1) was previously used during this database generation for the minidisk whose ddname is &2. The same minidisk virtual device address (cuu) cannot be used twice for a database.

System Action: The system waits for you to enter another virtual device address (cuu).

User Response: Enter another virtual device address (cuu).

ARI0635D **Do you want dual logging? Enter 0(No), or 1(Yes).**

Explanation: SQLDBGEN is asking whether dual logging is wanted for the database being generated.

System Action: Processing continues after you reply either 1 for YES or 0 for NO.

User Response: Decide whether the database should have dual logging. Your reply should be based upon whether you defined one or two log minidisks in the Virtual Machine Directory. Enter your reply.

ARI0636A **Please enter the number of database extents.**

Explanation: The SQLDBGEN EXEC is asking for the number of DBEXTENTs in the database being generated.

System Action: Processing continues after you reply with a valid number.

User Response: Decide how many DBEXTENTs the database should have. Enter the number.

ARI0637D **Do you want to continue? Enter 0(No), or 1(Yes).**

Explanation: Refer to the previous message given at the terminal. Because of the condition presented by the previous message, you are being given a choice of whether to continue processing.

System Action: Processing continues or ends, based on your reply.

User Response: Decide whether to continue processing. Enter your reply.

ARI0638D **Do you want to modify the** &1 &2 **file? Enter 0(No) or 1(Yes).**

Explanation: The named file (&1 &2) will be used as input to the database generation, ADD DBSPACE, or ADD DBEXTENT process. You are given a chance to change the file before the actual process begins.

System Action: The system waits for your reply. If you reply 1 for YES, the VM Editor will be invoked. If you reply 0 for NO, the VM Editor will not be invoked, and processing will continue.

User Response: Decide whether to change the file. Enter your reply.

Note: It is generally recommended that you reply Y and review (and revise as needed) the file.

ARI0639E **Error in using EXECIO for file** &1 &2 &3**. Return Code =** &4**.**

Explanation: An error occurred while EXECIO was being used to read or write the named CMS file (&1 &2 &3). The EXECIO return code (&4) is given. EXECIO is described in the *VM/ESA: CMS Command Reference* manual for your system.

System Action: Processing ends.

User Response: Refer this problem to your system programmer.

System Programmer Response: Follow your installation's problem determination procedures.

ARI0640I &1 **of** &2 **records copied to output disk.**

Explanation: This message is written to the operator console after every 4000 records are copied onto the output disk by the SQLDCBEX EXEC. If copying or expanding the directory, the records refer to 512 byte blocks. If copying the extents or the logs, the records refer to 4K pages.

Note: This message will not be displayed if the Data File Storage Management Subsystem (DFSMS™) Utility is used.

&1 is the current number of output records copied

&2 is the total number of output records which will be copied.

System Action: SQLCDBEX EXEC processing continues.

Note: No user or programmer action required.

ARI0641I **DB2 Server for VM** *service-name* **service applied. --***additional-instructions-given*

Explanation: *service-name* can be one of the following installation services: HELP text, DBS Utility load module, bootstrap file, datetime user exit router, system control and accounting, ISQL package, Flagger package, DBS Utility package, Resource Adapter, text module, MACRO, or a component stored in saved segments.

additional-instructions-given is a paragraph outlining any special action the System Programmer should take or conditions that the System Programmer should be

aware of as well as brief manual references.

System Action: Processing continues.

System Programmer Response: Read the paragraph following the first line of the message. Follow any instructions given.

ARI0642I **The minidisk currently accessed as the {Q | V} disk is not a DB2 Server for VM** type **minidisk accessed in write mode. Please acquire write access to a DB2 Server for VM {Q | V} minidisk by using file-mode** &1 **and rerun this EXEC.**

Explanation: This is the expected answer from the database manager when either the Q disk (usually SQLDBA 195) or the V disk (usually SQLDBA 193) have not been accessed in write mode.

Type is either service or production.

System Action: Processing ends.

Programmer Response: Ensure that the disk referred to above is accessed in write mode. For more information, refer to *DB2 Server for VM Program Directory*

ARI0643I **No DB2 Server for VM service files are on the A disk. Load and prepare these files, and then rerun this EXEC.**

Explanation: The DB2 Server for VM service EXEC was invoked. No service files were found on the A-Disk.

System Action: The service EXEC terminates.

Programmer Response: Load the DB2 Server for VM service files on to the A-Disk and rerun the service EXEC. Refer to the *DB2 Server for VM Program Directory* for detailed information on DB2 Server for VM product services.

ARI0644E **An error occurred during** &1 **command for** &3 &4 &5 &6 &7 &8**. Return Code =** &2**.**

Explanation: An error occurred during the indicated CP or CMS command (&1). The return code (.&2) is given, along with other parameters (&3 through &8) that provide information related to the failing command. This message will be preceded by a message from the failing command. The meaning of that message is given in the *VM/ESA: System Messages and Codes* manual.

System Action: Processing ends.

User Response: Refer this problem to your system programmer.

System Programmer Response: Follow your installation's problem determination procedures.

ARI0645E **A server with the server name** *server-name* **cannot be found by** *filename***.**

Explanation: The SQLADBSP, SQLADBEX, SQLCIREO, SQLLOG, or SQLCDBEX EXEC was invoked, but no *server-name* SQLFDEF file was found for the indicated application server *server-name* on the DB2 Server for VM production minidisk. *filename* identifies the EXEC.

System Action: Processing ends.

User Response: If *server-name* was misspelled, correct the spelling and reenter the command. If *server-name* was spelled correctly and no application server generation has been done, invoke SQLDBGEN or SQLDBINS to generate the application server. If *server-name* was spelled correctly and an application server generation has been done, follow your installation's problem determination procedures.

ARI0646D &1 &2 &3 **was found. Should the database manager use this file for processing? Enter: 0(No) to erase the file, or 1(Yes) to use the file.**

Explanation: During the ADD DBSPACE, ADD DBEXTENT or DELETE DBEXTENT processing, the database manager found the indicated CMS file (&1 &2 &3). You can use it for processing, or delete it and generate a new version.

System Action: Processing continues after you reply 1 for YES or 0 for NO.

User Response: Decide whether to use the indicated file. Enter your reply.

ARI0647D **Do you want to do a CMS FORMAT/RESERVE command on disk** &1**? Enter 0(No) or 1(Yes).**

Explanation: CMS FORMAT and RESERVE commands will be issued for the database minidisk whose virtual device address (cuu) is &1. This process will erase all files on the minidisk.

System Action: Processing continues or ends, based on your reply.

User Response: Decide whether to continue. Enter your reply.

ARI0648A **Enter number of** &1. &2. **to be added to the database.**

Explanation: You are being prompted for the number of PUBLIC, PRIVATE, or INTERNAL DBSPACEs, or the number of DBEXTENTs to be added to the database.

System Action: Processing continues after you reply with a valid number.

User Response: Decide the correct number and enter it.

ARI0649A **Enter the number of pages and storage pool assignment for** &1 &2 **DBSPACE.**

Explanation: You are being prompted for the number of pages and the storage pool number for the indicated DBSPACE (for example, FIRST PUBLIC DBSPACE).

System Action: Processing continues after you reply with two valid numbers.

User Response: Decide the correct numbers and enter them.

ARI0650E **An error occurred during** &1 &2 **processing.**

Explanation: DB2 Server for VM processing was not successful. An error occurred during the process identified by &1 &2 in the message. Refer to previous messages displayed at the virtual console or to the SYSPRINT file for a description of the specific error that occurred.

System Action: Processing ends.

User Response: Follow your installation's problem determination procedures.

ARI0651E **Invalid** &1 **keyword parameter value** &2**.**

Explanation: The indicated keyword (.&1) was specified with the indicated parameter value (.&2). This value is not valid for the keyword.

This message can be issued by the ARISAVES EXEC if &2 = DBSS or RDS.

System Action: Processing ends.

User Response: Reenter the command with the correct keyword and parameter value.

ARI0652E &1 **file** &2 &3 &4 **not found. STATE command Return Code =** &5**.**

Explanation: The indicated CMS file (&2 &3 &4) could not be found. The return code (.&5) from the CMS STATE command is given. &1 is a description of the file.

System Action: Processing ends.

User Response: Refer this problem to your system programmer.

System Programmer Response: Follow your installation's problem determination procedures.

ARI0653E &1 &2 **error.** &3 **Return Code =** &4**.**

Explanation: An error occurred during the process (*&1 &2*) indicated in the message. *&3.* is the process or EXEC that detected the problem and is sending back the return code (*&4*).

Possible values for *&3.* are:
- CMS command
- CP command
- EXEC
- A DB2 Server for VM EXEC name.

System Action: Processing ends.

User Response: Refer this problem to your system programmer. The system programmer will need all of the information in the text of this message as well as the message text of any error messages displayed prior to this message.

System Programmer Response: Different sources of information for each process or EXEC are displayed as **&3.**. If **&3.** is:

- CMS command

  To diagnose the problem, refer to the *VM/ESA: CMS Command Reference* manual for your operating system using the process name and return code provided in the message text.

- CP command

  To diagnose the problem, refer to the *VM/ESA: CP Command and Utility Reference* manual for your operating system using the process name and return code provided in the message text.

- EXEC
  - During SQLSTART processing

    The error occurred during a call to the ARISFDEF EXEC. (This EXEC is dynamically created by copying the dbname SQLFDEF file from the DB2 Server for VM production minidisk to the A-disk.)

    This message is preceded by message ARI0644E or message ARI0617E, which gives the return code from a failing CP LINK or CMS FILEDEF command. To diagnose the problem, refer to the above instructions for the CMS and CP return codes.

  - During SQLINIT processing

    The error occurred during a call to the SQLDBID EXEC. Return codes from that EXEC are:

    4 Incorrect parameter list (more than two parameters, first parameter not LIFO or FIFO, or second parameter not ALL).

    8 Character string "SQLDBID " not found in ARISRMBT MODULE.

    12 ARISRMBT MODULE not found on A-disk.

  - A DB2 Server for VM EXEC

The named DB2 Server for VM EXEC (**&3.**) has detected a serious problem. For all return codes, the EXEC has to be traced in order to find the cause of the problem.

Use the procedures described for this message for your initial problem diagnosis. Follow your installation's problem determination procedures.

ARI0654E **An error occurred in** &1 **while attempting to remove** &2 **from the free storage area. Return Code =** &3**.**

Explanation: An error occurred while deleting a DB2 Server for VM module from the CMS free storage virtual storage area.

&1 is the command used to delete the module from the free storage area.

&2 is the module name.

&3 is the return code supplied by the command identified by &1.

System Action: Virtual print files containing the virtual console and SYSPRINT messages displayed by database manager processing are created. The EXEC processing ends.

User Response: Refer to the virtual printer file messages to confirm that all database processing completed successfully. If necessary, rerun the EXEC after the free storage processing error has been corrected.

ARI0655I **Input file (SYSIN):** &1 &2 &3**.**

Explanation: This message identifies the SYSIN file being used for DB2 Server for VM processing.

&1 identifies the virtual reader (READER), virtual console (TERMINAL), or the file name of a CMS file.

&2 is the file type of a CMS file (optional).

&3 is the file mode of a CMS file (optional).

System Action: Processing continues.

ARI0656I **Message file (SYSPRINT):** &1 &2 &3**.**

Explanation: This message identifies the SYSPRINT file being used for DB2 Server for VM processing.

&1 identifies the virtual printer (PRINTER), virtual console (TERMINAL), or the file name of a CMS file.

&2 is the file type of a CMS file (optional).

&3 is the file mode of a CMS file (optional).

System Action: Processing continues.

ARI0657I **EMSG function value restored to** &1**.**

Explanation: The CP EMSG function value has been restored to the value &1 after being temporarily reset by DB2 Server for VM processing.

System Action: Processing continues.

ARI0658E **Invalid or conflicting** &1 **EXEC parameter(s) specified.**

Explanation: EXEC processing was not successful. The parameters specified in the EXEC identified by &1 are not valid.

System Action: EXEC processing ends.

User Response: Rerun the EXEC, specifying correct parameters.

ARI0659I **Line-edit symbols reset: LINEND=**&1 **LINEDEL=**&2 **CHARDEL=**&3 **ESCAPE=**&4 **TABCHAR=**&5**.**

Explanation: This message identifies the CP terminal line-edit symbol values being temporarily used by DB2 Server for VM processing.

System Action: Processing continues.

User Response: To eliminate this message, you can update your virtual machine PROFILE EXEC to set the CP terminal line-edit symbols to the values identified in this message. Otherwise, no action is required.

ARI0660I **Line-edit symbols restored: LINEND=**&1 **LINEDEL=**&2 **CHARDEL=**&3 **ESCAPE=**&4 **TABCHAR=**&5**.**

Explanation: The CP terminal line-edit symbol values have been restored to the values shown. (They had been temporarily reset by DB2 Server for VM processing.)

System Action: Processing continues.

ARI0661E **Duplicate keyword** &1 **specified.**

Explanation: EXEC processing was not successful. This EXEC keyword parameter &1 has been specified more than once.

System Action: EXEC processing ends.

User Response: Rerun the EXEC without duplicate keyword parameters.

ARI0662I **EMSG function value reset to** &1**.**

Explanation: The CP EMSG function value has been temporarily set to the value &1 by the database manager.

System Action: Processing continues.

User Response: To eliminate this message, you can update your virtual machine PROFILE EXEC to set the CP EMSG function value to the value identified in this message. Otherwise, no action is required.

ARI0663I FILEDEFS in effect are:

Explanation: This message is followed by a list of the CMS FILEDEF commands now in effect.

System Action: Processing continues.

ARI0664E DB2 Server for VM database installation/service/migration error occurred while executing the &1 &2**.**

Explanation: The DB2 Server for VM database was not successfully installed or updated. An error occurred during the database installation, maintenance, service, or migration processing.

The parameters &1 &2 identify the DB2 Server for VM component that was executing when the error occurred (SQLDBGEN EXEC, DBS UTILITY, or ASSEMBLER PREP).

System Action: Processing ends. Virtual print files containing the virtual console and SYSPRINT messages displayed by the database manager are created.

User Response: Refer to the virtual printer file messages to identify the actual error that occurred. Ensure that all preliminary steps described in the *DB2 Server for VM Program Directory* or the *DB2 Server for VM System Administration* manual have been performed. Correct the errors and rerun the EXEC.

ARI0665E DB2 Server for VM database was not processed successfully. Refer to the previous error message(s) written to the terminal and/or contained in the virtual printer files created.

Explanation: An error occurred while installing, migrating, or servicing the DB2 Server for VM database.

System Action: Virtual print files containing the virtual console and SYSPRINT messages displayed by the database manager are created. Processing ends.

User Response: Refer to the virtual printer file messages to identify the actual error that occurred. Correct the error and rerun the EXEC.

ARI0666I DB2 Server for VM database installed, serviced or migrated successfully.

Explanation: This message is displayed by the installation processor, migration processor, or maintenance processor after a DB2 Server for VM database has been successfully installed or updated.

System Action: Virtual print files containing the virtual console and SYSPRINT messages displayed by the database manager are created. Processing completes successfully.

ARI0673I All &1 &2 **processing completed successfully.**

Explanation: This message is displayed by a DB2 Server for VM EXEC after the process identified by &1 &2 has successfully completed.

System Action: EXEC processing continues.

ARI0674E You did not establish application server access. Run the SQLINIT EXEC: SQLINIT DB (server-name**)**

Explanation: Access to a DB2 Server for VM application server was not established prior to running the SQLDBSU EXEC in multiple user mode. The ARISRMBT MODULE A is not found.

System Action: Command is ended.

User Response: Run the SQLINIT EXEC to establish access to an application server and rerun the SQLDBSU EXEC.

ARI0675E There are no keyword parameters.

Explanation: EXEC processing was not successful. Required DB2 Server for VM EXEC parameters were not specified.

System Action: EXEC processing ends.

User Response: Rerun the EXEC, specifying the required parameters.

ARI0676E &1 **is not a valid preprocessor to execute.**

Explanation: EXEC processing was not successful. An incorrect DB2 Server for VM preprocessor identifier (.&1) is specified as the first parameter. The first SQLPREP EXEC parameter must be ASM, C, COBOL, FORTRAN, or PLI.

System Action: The SQLPREP EXEC processing ends.

User Response: Rerun the SQLPREP EXEC, specifying a valid DB2 Server for VM preprocessor identifier (ASM, C, COBOL, FORTRAN, or PLI) as the first parameter.

ARI0677I You may not specify &1 **without DBNAME.**

Explanation: &1 is a DB2 Server for VM single user mode parameter. It is meaningful only if the DBNAME keyword is specified.

System Action: Processing ends.

User Response: If you want single user mode

operation, specify the DBNAME parameter. Otherwise, omit the keyword indicated in the message.

ARI0678E Matching right parenthesis missing after the keyword &1.

Explanation: EXEC processing was not successful. The keyword (.&1) is followed by a left parenthesis but the matching right parenthesis is missing.

System Action: Processing ends.

User Response: Reenter the command with the proper pairing of parentheses.

ARI0680I If message ARI0922I has been displayed, you must rerun SQLADBEX with the same input after correcting the cause of the abnormal termination. If message ARI0922I has not been displayed, follow the restart procedure outlined in the *DB2 Server for VM System Administration* manual.

Explanation: This message follows message ARI0650E. The SQLADBEX EXEC received a nonzero return code from SQLSTART processing. As a result, the database manager does not update the SQLFDEF file. If message ARI0922I is displayed, the system commits all the ADD and DELETE actions to the database. To update the control file, you must run the SQLADBEX EXEC to completion again by using the same input.

System Action: The SQLADBEX EXEC ends with the return code returned by the SQLSTART EXEC.

User Response: Review the console output. If message ARI0922I is displayed, correct the cause of the abnormal termination and rerun the SQLADBEX EXEC with the same input to update the control file. If the error occurs before message ARI0922I is displayed, correct the cause of the error and proceed with restarting the ADD or DELETE DBEXTENT function. For a detailed description, refer to information on the SQLADBEX EXEC in the *DB2 Server for VM System Administration* manual.

ARI0681E The keyword &1 has no or too many parameters.

Explanation: EXEC processing was not successful. Either no parameters or too many parameters were specified in the pair of parentheses after the keyword &1.

System Action: Processing ends.

User Response: Correct the parameter and reenter the command.

ARI0682E It is invalid to specify LOG2 without specifying LOG1.

Explanation: You invoked the SQLLOG EXEC with LOG2 specified but LOG1 omitted. This is not a valid combination of parameters.

System Action: Processing terminates.

User Response: If you want to make a change in logs, reenter the command with both LOG1 and LOG2 specified. If you want to do a COLDLOG function without changing the log configuration, reenter the command without LOG1 or LOG2.

ARI0683E It is invalid to use the same value for LOG1 and LOG2.

Explanation: You invoked the SQLLOG EXEC with the same virtual device address specified for both logs. The logs must have different virtual device addresses.

System Action: Processing ends.

User Response: Reenter the command with proper virtual device addresses specified.

ARI0684I *cuu-1* will replace *cuu-2* as *ddname*.

Explanation: Currently, the database is defined with the second virtual device address (*cuu-2*) as the LOGDSK1 or LOGDSK2. (as indicated by *ddname*). The invocation of SQLLOG has requested that the first virtual device address (*cuu-1*) become the LOGDSK1 or LOGDSK2.

This informational message may be followed by another informational message. Message ARI0615D will then give you the opportunity to continue or stop the process.

System Action: Processing continues.

ARI0685I &1 will be {added | dropped} as LOGDSK2.

Explanation: The invocation of SQLLOG has requested that the indicated virtual device address (.&1) be added to or dropped from the database definition as LOGDSK2. This informational message will be followed by message ARI0615D which gives you the opportunity to continue or stop the process.

System Action: Processing continues.

ARI0686E *cuu* was given for *ddname-1* but is already defined as *ddname-2*.

Explanation: The virtual device address (*cuu*) was given as LOG1 or LOG2 (*ddname-1*) but was previously used in the database as indicated by the ddname displayed (*ddname-2*). Each minidisk in the database must have a unique virtual device address.

System Action: Processing ends.

User Response: Reenter the command with a proper virtual device address.

ARI0687E **File** &1 &2 &3 **is not in the expected format.**

Explanation: The contents of the indicated CMS file (&1 &2 &3) do not match that which is expected.

This error can occur during the processing of the following EXECS:
- SQLGLOB
- SQLINIT

SQLINIT and SQLGLOB retrieve CCSIDSBCS, CCSIDMIXED, and CCSIDGRAPHIC information from the file ARISCCS MACRO. However, the information retrieved is not valid.

System Action: Processing ends.

User Response: Contact your database administrator.

Database Administrator Response: For an SQLINIT EXEC or SQLGLOB EXEC processing error, ensure the user is linking to the correct copy of ARISCCS MACRO. If the system defined ARISCCS macro has been corrupted, then do the following:
- Ensure that SYSTEM.SYSCCSIDS is correct.
- Follow the steps in the *DB2 Server for VM Program Directory* for the ARISDBMA EXEC, specifying the DBNAME and component (CCSID) to create a new version of the ARISCCS MACRO.

ARI0688D **COLDLOG will initialize the log on disk** *cuu1***.**

Reformatting will erase only log data. Reconfiguring will erase both log data and the log history.
Warning: If you are running LOGMODE=L, reconfiguring invalidates previous or current log archive restore sets unless this log history is restored before the next COLDLOG.
Enter CANCEL to cancel this COLDLOG or CONTINUE to continue.

Explanation: This message is issued from the SQLLOG EXEC. It describes the purpose of a coldlog and its effects on the log disk identified by its virtual device address (cuu1).

WARNING: If you are running with LOGMODE=L and you decide to reconfigure, you will not be able to restore the log archives that were taken prior to this coldlog, unless you have a backup copy of the log history to be erased. In order to continue running with LOGMODE=L, you will need to take a database archive after performing this coldlog. Refer to the *DB2 Server for VM System Administration* manual for more information on log reconfiguration and reformatting.

System Action: Processing continues or ends, depending on your reply.

Operator Response: Decide whether to reply CANCEL or CONTINUE. Enter your reply.

ARI0689E **Database** &1 **has not been generated. Initialization for use of DB2 Server for VM failed.**

Explanation: The dbname SQLDBN file for the requested database (.&1) was not found on the DB2 Server for VM production minidisk. The DBNAME parameter may be misspelled, or the database generation has not been done.

System Action: The invoked EXEC terminates.

User Response: If the DBNAME parameter is misspelled, reenter the EXEC with the correct spelling. If the DBNAME is spelled correctly, notify your system programmer to determine the status of the database.

ARI0690I **Virtual address** cuu **is already in use for server** server-name**.**

Explanation: The virtual device address **cuu** that you entered is already in the server, *server-name*, SQLFDEF file. (It is already defined as an application server disk).

System Action: The system issues the prompt again.

User Response: Enter another **cuu** when prompted.

ARI0691E **EXECIO command error occurred while {reading|writing} the file** *filename***. EXECIO Return Code =** *&1***.**

Explanation: The CMS EXECIO command failed during execution. *&1* is the return code generated by the EXECIO command. *filename* can be ARISPDEC MACRO *fm*, ARIDATA FILE A, or any files. *fm* is the file mode.

System Action: The read operation is terminated. If it is the write operation, the output file is not created.

User Response: Investigate the return code in the *VM/ESA: CMS Command Reference* manual and take the required action.

ARI0692E **ISQL cannot start because of an unexpected system error. CMS Return Code =** return code**.**

Explanation: An unexpected error occurred during ISQL start up. The CMS return code was encountered while attempting to execute module ARISISBT.

System Action: ISQL processing stops.

User Response: Look up the CMS return code in the

System Codes section of the *VM/ESA: System Messages and Codes* If you do not understand the error, notify your system programmer.

System Programmer Response: Perform CMS problem determination.

ARI0693E **ISQL cannot begin. The keyword** keyword **is not correct. The only keyword that you may enter is ROUTINE. Try again.**

Explanation: You entered an incorrect keyword when using an EXEC to start ISQL.

System Action: ISQL processing stops with a return code of 999.

User Response: When you start ISQL, use only the keyword ROUTINE (or none at all).

ARI0694I **Too many parameters were passed to the ISQL EXEC.** data **was ignored, but ISQL will continue.**

Explanation: You entered something after the ROUTINE parameter when using an EXEC to invoke ISQL.

Data is the excess data you entered.

System Action: ISQL processing continues.

ARI0695I **Warning:** &1 : &2 **not to be installed.**

Explanation: The user has chosen not to execute the installation EXEC.

&1 is the product identifier.

&2 is the product name.

System Action: The processing is terminated. The product specified is not installed.

User Response: Re-issue the command. You will then again be offered the choice of installation.

ARI0696E **ISQL cannot begin. Precede your keyword value with a keyword. Try again.**

Explanation: You entered a keyword value in parentheses without entering a keyword before it.

System Action: ISQL processing stops with a return code of 999.

User Response: Start ISQL with the keyword followed by the keyword value. The keyword value must be in parentheses.

ARI0697E **ISQL cannot begin. The format of your keyword is not correct. Reenter the ISQL EXEC with the keyword followed by the keyword value. The keyword value must be in parentheses.**

Explanation: The format of your keyword was not correct.

System Action: ISQL processing stops with a return code of 999.

User Response: Reenter the ISQL EXEC with parentheses around each keyword value you specify.

ARI0698E **ISQL cannot begin. The routine name and parameters specified are more than 250 characters long. Try again.**

Explanation: The routine name and parameter list you entered exceeded the limit of 250 characters.

System Action: ISQL processing stops with a return code of 999.

User Response: Start ISQL with a routine name and parameter list of not more than 250 characters.

ARI0699E **ISQL cannot begin because of an error during ISQL start up. See the previous error message for a description of the error.**

Explanation: An error occurred during ISQL bootstrap processing.

System Action: ISQL processing stops with a return code of 999.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a DB2 Server for VM system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI6000I **The application server environment was not successfully restored.**

Explanation: An SQLINIT failed when trying to restore to the application server environment that was present before the SQLPREP EXEC was executed.

User Response: When the SQLPREP EXEC terminates, the user should issue a successful SQLINIT in order to return to the environment previous to executing SQLPREP EXEC.

ARI6001E **Duplicate server-names** &1 **were specified in** &2.

Explanation: Duplicate server-names were found during processing identified by &2 in the message during the execution of the SQLPREP EXEC.

System Action: Processing ends.

User Response: Reenter the command with the removal of the duplicate server-name from the list or from the file contents.

ARI6002E **You may not specify keyword** &1 **when running in single user mode.**

Explanation: SQLPREP EXEC cannot be executed when DBLIST/DBFILE and DBNAME keywords are specified together. DBLIST/DBFILE keyword can only be specified when running in multiple user mode. DBNAME keyword can only be used in single user mode.

System Action: Processing ends.

User Response: Reenter the command to execute in multiple user mode (DBNAME keyword may not have been specified).

ARI6003E **No server names were specified in application server file** &1 &2 &3**.**

Explanation: There were no server names found in the file &1 &2 &3 when processing DBF or DBFILE keywords.

System Action: Processing ends.

User Response: Reenter the command with appropriate server names specified in the file.

ARI6004E **Invalid server name** &1 **specified in** &2**.**

Explanation: There was an invalid server name, identified by &1, found in the list or file when processing DBLIST/DBFILE keyword. The server name could be greater than eight characters long or the comma delimiters were incorrectly used.

System Action: Processing ends.

User Response: Correct and reenter the command.

ARI6005I **The** &1 **EXEC finished normally. Maintenance was done on the CMS HELP text files.**

Explanation: The ARISDBMC EXEC has completed successfully.

ARI6006E **The** &1 **EXEC finished with an error. Maintenance has not been performed on the CMS HELP text files.**

Explanation: The ARISDBMC EXEC has completed successfully.

User Response: Investigate any previous error messages and correct the errors. Rerun the EXEC that invoked ARISDBMC.

ARI6007D **Where do you want to install the CMS HELP text files?**

Explanation: Because several different language versions of the CMS HELP text may exist simultaneously, you must specify where to install the CMS HELP text for the language being added.

This message will be followed by messages ARI6009D and ARI6046D to prompt the user for the required information.

User Response: Review the explanation of message ARI6028W, and respond appropriately to ARI6009D and ARI6046D.

ARI6008D **From where do you want to delete the CMS HELP text files?**

Explanation: Because several different language versions of the CMS HELP text may exist simultaneously, you must specify the location of the CMS HELP text for the language whose text is being deleted.

This message will be followed by messages ARI6009D and ARI6046D to prompt the user for the required information.

User Response: Respond appropriately to ARI6009D and ARI6046D.

ARI6009D **Enter the owner user ID of the minidisk to use or enter SKIP to abort the process for this language.**

Explanation: The CMS HELP Text for a language is about to be serviced (INSTALLED or DELETED).

System Action: Processing continues.

User Response: Enter the ownerid of the minidisk which the CMS HELP Text will be installed on or deleted from, or enter SKIP to abort the service to the language.

ARI6010D **Do you want to install English DB2 Server for VM HELP text? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: During installation or migration of the database manager, the installer can chose to reload the English language HELP text, suppress loading, or

111(QUIT) from the installation or migration process.

System Action: The system waits for you to enter your response.

System Programmer Response: Respond 1(YES) if you want to reload the English help text, 0(NO) if you want to suppress loading of help text, or 111(QUIT) if you want to exit from the installation or migration process.

ARI6011I **---> DB2 Server for VM &1 minidisk update complete.**

Explanation: This is the expected response from the database manager. The DB2 Server for VM &1 (service or production) minidisk has been successfully updated.

System Action: The processing continues.

ARI6012A ****DB2 Server for VM production minidisk updated. Update all secondary production minidisks according to instructions in the DB2 Server for VM Program Directory.**

Explanation: This is the expected response from the database manager. The DB2 Server for VM production minidisk has been updated.

System Action: The processing continues.

User Response: You must update any secondary DB2 Server for VM production minidisks. Refer to the *DB2 Server for VM Program Directory* for instructions.

ARI6013D **Begin installing DB2 Server for VM as a VM feature program product.** &1 - &2 **Version** &3 **Release** &4 **Modification** &5**. Do you want to install this product now? Enter** 0**(No), 1(Yes), or 111(Quit).**

Explanation: This is the expected response from the database manager. The specified level of the database manager indicated by (&1) (&2) (&3) is about to be installed. You are asked if you wish to continue or not.

&1 is the product identifier for the database manager.

&2 is the product name for the database manager.

&3 is the version of the database manager to be installed.

&4 is the release level of the database manager to be installed.

&5 is the modification level of the database manager to be installed.

System Action: The processing continues or terminates based on the user's response.

User Response: You must choose to: not install the DB2 Server for VM program product at this time, or install the DB2 Server for VM program product at this time, or QUIT the installation EXEC.

ARI6020E **Unable to locate ARI*xxxx* MACRO on the service disk.**

Explanation: The ARISDBMA EXEC is unable to find the ARI*xxxx* file on the service minidisk during installation of National Language HELP text. The *xxxx* in the file name is the language key.

System Action: The system waits for you to enter the next command.

System Programmer Response: Verify that the language specified is entered correctly, and that the MACRO file for the National Language resides on the service disk.

ARI6021E **Invalid ARI*xxxx* MACRO file on the service disk.**

Explanation: A file ARI*xxxx* MACRO file was found on the service disk but is not in a valid format for ARISDBMA EXEC. The *xxxx* will appear as a language key value.

System Action: The system waits for you to enter the next command.

System Programmer Response: Verify that the ARI*xxxx* MACRO file contains only records of the form READ FILE file name MACRO where file name is a macro file name, and *xxxx* is the language key. Correct if necessary.

ARI6022D **Do you want to make this the default language? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: The National Language in which the new help text is written can be made the default language.

System Action: The system waits for you to enter your response.

System Programmer Response: Enter YES or NO, depending on whether you want to change the default language setting, or enter QUIT to exit.

ARI6023E **An error was encountered while updating** &1 &2 &3**.**

Explanation: The EXEC was unable to update the file with file name (&1) and file type (&2) on the minidisk accessed as file mode (&3).

System Action: The processing is terminated.

User Response: Verify that the file specified exists on the minidisk accessed as file mode (&3).

Follow the steps listed in the *DB2 Server for VM Program Directory*, then re-issue the command.

ARI6024I The CMS HELP text files were erased successfully.

Explanation: The CMS help text files have been erased successfully.

System Action: Processing continues.

ARI6025E An error occurred while erasing the CMS HELP text files. Return Code = &1**.**

Explanation: An error occurred, while erasing the HELP text, on the specified minidisk. This message gives the ERASE return code.

User Response: Take appropriate action to resolve the error based on the ERASE return code. For an explanation of the ERASE return code, you can use the HELP ERASE command.

ARI6026E You must keep at least one language.

Explanation: At least one language must be installed in the database. You are trying to delete the last or only language from the database.

System Action: Processing ends.

System Programmer Response: Ensure that at least one language remains in the database.

ARI6027D Indicate the key value for the languages to be deleted. Enter a null line to process your selections, or enter QUIT to exit.

Explanation: A list of currently installed languages is displayed. You must specify the languages to delete.

System Action: The system waits for your input.

System Programmer Response: Enter one or more language keys (separated by blanks or commas), or enter QUIT to exit. If you type QUIT, no languages are deleted.

ARI6028I Warning: The {minidisk | SFS directory} specified is the DB2 Server for VM {production minidisk | production SFS directory}. You may use this {minidisk | SFS directory} if you are installing HELP text files in one national language only. If you are going to install more than one national language, you must use other {minidisks | SFS directories} to prevent overwriting the current HELP text.

Explanation: It is possible to install CMS HELP text in more than one language. For the CMS HELP text of all languages to be available, CMS HELP text should not be installed on the production disk because the mode of the production disk would always take precedence over the other CMS HELP disks.

User Response: Determine if your installation will be supporting CMS help for multiple languages before responding to the next message.

ARI6029I The CMS HELP text files are not on the disk specified.

Explanation: The files containing the text were not found, while you tried to delete the CMS HELP text.

User Response: Determine if the correct disk was specified as the location of the CMS help to be deleted. If necessary, rerun the EXEC that invoked ARISDBMC to delete the HELP text. Refer to the *DB2 Server for VM Program Directory* for further information on removing HELP text.

ARI6030D Indicate the key value for the languages to delete, or enter ALL to delete all languages. Enter a null line to process your selections, or enter QUIT to exit.

Explanation: A list of currently installed languages is displayed. You must specify which languages to delete.

System Action: The system waits for your input.

System Programmer Response: Enter one or more language keys (separated by blanks and/or commas) or ALL for all languages, or QUIT to exit. If you type QUIT, no languages are deleted.

ARI6031E You have not chosen any languages yet.

Explanation: A list of currently installed languages is displayed. You must specify which languages to delete. If Enter was pressed before any language keys were specified, this message is displayed.

System Action: The system waits for your input.

System Programmer Response: Enter one or more language keys (separated by blanks and/or commas) or ALL for all languages, or QUIT to exit. If you type QUIT, no languages are deleted.

ARI6032E Invalid language key: &1**.**

Explanation: You specified a language key &1 that does not exist in SYSLANGUAGE.

System Action: The system waits for your input.

System Programmer Response: Enter one or more language keys (separated by blanks or commas), ALL for all languages, or QUIT to exit. If you type QUIT, no languages are deleted.

ARI6033E &1 **has already been selected for deletion.**

Explanation: You specified a language key that has already been marked for deletion.

System Action: The system waits for your input.

System Programmer Response: Enter one or more language keys (separated by blanks or commas) or ALL for all languages, or QUIT to exit. If you type QUIT, no languages are deleted.

ARI6034D **You have requested that the current default language (KEY =** &1**) be deleted. Choose a new default language by typing its key, or enter QUIT to exit.**

Explanation: You chose the default language as one of the languages to be deleted. More than one language would remain on the system, therefore, you must choose a new default language.

System Action: The system waits for your input.

System Programmer Response: Enter one of the language keys displayed, or type QUIT to exit. If you type QUIT, no languages are deleted.

ARI6035D **These are the languages that you chose to delete. Are these correct? Enter 0(No) or 1(Yes).**

Explanation: A list of the languages that you chose to delete is displayed. You must now confirm these to be correct for the deletion to take place.

System Action: The system waits for you to enter the next command.

System Programmer Response: Enter YES or NO, depending on whether you wish delete the indicated languages.

ARI6036I **DB2 Server for VM HELP text deletion complete.**

Explanation: The database manager completed the deletion of the help text you specified.

System Action: The system waits for you to enter the next command.

ARI6037E **An error was encountered while deleting DB2 Server for VM HELP text.**

Explanation: Your specified help text was not successfully deleted.

System Action: The system waits for you to enter the next command.

System Programmer Response: This message will probably be preceded by other error messages. Perform problem determination based on the other error messages.

ARI6038I **You did not delete any languages.**

Explanation: You chose to exit without deleting any help text.

System Action: The system waits for you to enter the next command.

ARI6039E **The attempt to retrieve status of HELP text failed.**

Explanation: The database manager was unable to obtain data from one of the following database manager tables:

SYSTEM.SYSOPTIONS
SQLDBA.SYSLANGUAGE
SYSTEM.SYSCATALOG
SYSTEM.SYSDBSPACES

System Action: The database manager terminates with a return code from the DBS utility.

System Programmer Response:

Verify the existence of the tables listed above.
Ensure that you have SELECT authority to these tables.
Make sure you supply the correct password to SQLDBA.
Also verify the existence of SQLDBA.SYSTEXT1 and SQLDBA.SYSTEXT2 tables.

ARI6040E **No HELP text is installed in this application server.**

Explanation: You have invoked the ARISDBHD EXEC or ARISDBLD EXEC and specified an application server in which no HELP text presently exists.

System Action: The system waits for you to enter the next command.

System Programmer Response: None, or invoke ARISDBHD or ARISDBLD again with another server name.

ARI6041E **Invalid response:** &1**.**

Explanation: You did not provide an acceptable response to a question.

System Action: The system waits for you to enter an acceptable response.

System Programmer Response: Review the question, and provide an acceptable response.

ARI6042I Warning: Application server &1 has not been generated on this processor. Ignore this message if the application server has been defined as a global resource on another processor.

Explanation: The DBNAME SQLDBN file for the requested application server (&1) was not found on the DB2 Server for VM production minidisk. The DBNAME parameter may have been misspelled, or the application server generation may not have been done, or the application server is a global resource on another processor in the TSAF collection.

System Action: The SQLINIT EXEC continues processing.

User Response: If the DBNAME parameter is misspelled, reenter the EXEC with the correct spelling. If the DBNAME is spelled correctly, and the application server is a global resource on another processor in the TSAF collection, ignore this message. If the DBNAME is spelled correctly, and the application server is a local resource notify your system programmer to determine the status of the application server.

ARI6043I One or more of the files are not on the service disk.

Explanation: The required files were not found on the DB2 Server for VM service disk, while trying to copy the CMS help text to the specified disk.

User Response: Refer to the *DB2 Server for VM Program Directory* and the *DB2 Server for VM System Administration* manual for further information on loading or reloading HELP text.

ARI6044D Do you want to use this storage device? Enter 0(No) or 1(Yes).

Explanation: This message follows the warning message ARI6028W to obtain confirmation from the user to continue.

User Response: If you do not wish to use the production disk for CMS HELP, enter `0` and you will be prompted for the address information of a new disk. If you do wish to install the CMS HELP text on the production disk, enter 1 to continue processing.

ARI6045E Entry for this language not in ARISNLSC MACRO.

Explanation: An entry for the language for which the ARISAVES EXEC is attempting to create a discontiguous shared segment (saved segment) has not been found in the ARISNLSC MACRO.

System Action: Processing terminates.

User Response: Ensure that this language has already been installed, i.e. that the ARISDBMA EXEC has been executed for this language, so that the required entry is properly created.

ARI6046D Enter the CUU of the minidisk to use.

Explanation: This prompt is requesting the cuu of the minidisk to be used for the addition or deletion of the CMS help files for the database manager.

User Response: Enter the cuu of the minidisk to be used for this language.

ARI6047E The origin for component &1 is greater than 16 megabytes. The segment must be defined and saved below 16 megabytes.

Explanation: Invalid origin has been specified for the indicated component (&1). Only the RA component may be saved as a segment above 16 megabytes.

System Action: Processing terminates.

System Programmer Response: Make sure that the saved segment for the component has been defined below 16 megabytes ('1000000'X). Re-run the ARISAVES EXEC with the correct origin less than 16 megabytes.

ARI6048E Errors occurred while recreating the default DB2 Server for VM bootstrap files on the production minidisk. Refer to previous console messages for a description of the error.

Explanation: Errors occurred during the re-creation of the bootstrap files on the Production minidisk.

System Action: Processing terminates.

System Programmer Response: Refer to the previous console messages or to the virtual printer messages for a description of the errors encountered. Refer to this manual for details regarding the messages listed. Perform any corrective steps and re-issue the command.

ARI6049E Invalid ARISPDEC MACRO &1. Required last line entry in the MACRO was not found.

Explanation: The format of the ARISPDEC MACRO is invalid. The line 'END ARISPDEC MACRO' was not found as the last entry in the file.

&1 is the file mode on which the ARISPDEC MACRO file exists.

System Action: Processing terminates.

System Programmer Response: Restore the ARISPDEC MACRO from the DB2 Server for VM Product tape, or insert the required last line entry as the last line in the ARISPDEC MACRO file. The format of the line is as follows: 'END ARISPDEC MACRO'

ARI6050E **You may not invoke** &1 &2 &3 **from CMS subset mode.**

Explanation: The CMS EXEC specified has been invoked from subset mode, but this is not allowed.

System Action: Processing terminates.

User Response: Exit subset mode and re-execute the EXEC or re-execute the EXEC with different parameters.

ARI6051I **DB2 Server for VM system minidisk access and load begun.**

Explanation: This is the expected response from the database manager. The database manager minidisk has been accessed and loading from the distribution tape has begun.

System Action: Processing continues.

ARI6052I **Linking to the DB2 Server for VM** &1 **minidisk (** &2 &3 **) by using the default write access password of** &4**.**

Explanation: This is the expected response from the database manager. This message is issued from the installation EXEC. The EXEC is attempting to link to the minidisk specified (service or production) with the default password specified.

&1 is the minidisk being linked to (service or production).

&2 is the owner ID of the minidisk being linked to.

&3 is the cuu of the DB2 Server for VM minidisk being linked to.

&4 is the default password being used to link to the minidisk.

System Action: Processing continues.

ARI6053I **First DB2 Server for VM** &1 **minidisk link attempt failed. Return Code =** &4**. Will attempt to link the** &2 &3 **minidisk again. If prompted, enter the** &2 &3 **minidisk write password.**

Explanation: The EXEC was not able to link to the minidisk specified.

&1 is the minidisk being linked to (service or production).

&2 is the owner ID of the minidisk being linked to.

&3 is the cuu of the minidisk being linked to.

&4 is the return code generated from the LINK command.

System Action: Another attempt to link to the minidisk specified is made. If this fails, the processing will terminate.

User Response: If prompted, enter the write password for the minidisk being linked. If the EXEC fails to link to the minidisk specified when it retries, investigate the return code issued in the *VM/ESA: CMS Command Reference* manual.

ARI6054E **Error linking to** &1 &2 **as** &3 **with** &4 **mode. The return code is** &5 **from CP LINK command.**

Explanation: The EXEC was not able to link to the minidisk specified.

&1 is the owner ID of the minidisk being linked to.

&2 is the cuu of the minidisk being linked to.

&3 is the cuu which the minidisk being linked to will be linked as.

&4 is the mode in which the minidisk was to be linked. (R/RR/RW/WR/W/MW)

&5 is the return code generated from the LINK command.

System Action: The processing is terminated.

User Response: Refer this problem to your system programmer.

System Programmer Response: Review the installation procedures outlined in the *DB2 Server for VM Program Directory*. Refer to the *VM/ESA: CMS Command Reference* manual for an explanation of the return code generated by the LINK command.

ARI6055I **Checking if the production minidisk needs formatting.**

Explanation: This is the expected response from the database manager. The message is issued during the installation process. If the files ARISPIDC MACRO or ARISQLLD LOADLIB exist on the DB2 Server for VM production minidisk, then the minidisk is not formatted.

System Action: Processing continues.

ARI6056I **Formatting of the production minidisk is not required.**

Explanation: The message is issued during the installation process. If the files ARISPIDC MACRO or ARISQLLD LOADLIB exist on the DB2 Server for VM production minidisk, then the minidisk is not formatted.

System Action: Processing continues.

ARI6057I **Neither &1 MACRO nor ARISQLLD LOADLIB is on the production minidisk. The production minidisk will be formatted.**

Explanation: The message is issued during the installation process. If the files ARISQLLD LOADLIB and the (&1) MACRO are not found on the production minidisk, then the minidisk is formatted. If the full product is being installed, (&1) will be ARISPIDC.

System Action: Processing continues.

ARI6058I **Verifying that the service minidisk is in CMS format and that no files are on it.**

Explanation: This is the expected response from the database manager. The message is issued during the installation process. If the DB2 Server for VM service minidisk contains files or is not in CMS format, it is then formatted.

System Action: Processing continues.

ARI6059I **Formatting of the service minidisk is not required.**

Explanation: This is the expected response from the database manager. The message is issued during the installation process. The DB2 Server for VM service minidisk does not contain files and is in CMS format, so it is not formatted.

System Action: Processing continues.

ARI6060I **Files are on the service minidisk. The minidisk will be formatted.**

Explanation: The message is issued during the installation process. Files were found on the DB2 Server for VM service minidisk, therefore it will be formatted before being used for installation.

System Action: Processing continues. The service minidisk is formatted.

ARI6061I **Formatting the DB2 Server for VM &1 minidisk.**

Explanation: The message is issued during the installation process. Files were found on the &1 (service) DB2 Server for VM minidisk, therefore it will be formatted before being used for installation. When files are found on the &1 (production) DB2 Server for VM minidisk, the minidisk is not formatted.

System Action: Processing continues.

ARI6062E **Error formatting &1 &2 minidisk. The return code is &3 from the CMS FORMAT command.**

Explanation: The CMS FORMAT command failed during execution.

&1 is owner ID of the minidisk being formatted.

&2 is cuu of the minidisk being formatted.

&3 is the return code generated by the FORMAT command.

System Action: The processing is terminated.

System Programmer Response: Investigate the return code in the *VM/ESA: CMS Command Reference* manual and take the required action.

ARI6063I **Begin loading the DB2 Server for VM &1 minidisk: &2 &3.**

Explanation: This is the expected response from the database manager. The DB2 Server for VM &1 (service or production) minidisk is being loaded.

&2 is owner ID of the minidisk being loaded.

&3 is cuu of the minidisk being loaded.

System Action: Processing continues.

ARI6064I **DB2 Server for VM &1 minidisk loaded successfully.**

Explanation: This is the expected response from the database manager. The DB2 Server for VM &1 (service or production) minidisk was loaded successfully.

System Action: Processing continues.

ARI6065E **DB2 Server for VM &1 minidisk load failed.**

Explanation: The DB2 Server for VM &1 (service or production) minidisk load failed.

System Action: The processing is terminated.

User Response: See previous messages to determine what caused the load to fail. Investigate the messages preceding this error message and perform the corrective action required.

ARI6067I **DB2 Server for VM default bootstrap files updated successfully.**

Explanation: This is the expected response from the database manager. The message is issued after the invocation of the SQLBOOTS EXEC. The bootstrap files have been successfully created.

System Action: Processing continues.

ARI6068I The above values are being used.

Explanation: This is the expected response from the database manager. The message is a confirmation message issued when you choose to continue with the values specified in the previous message.

System Action: Processing continues.

ARI6069A Please verify new values.

Explanation: This is the expected response from the database manager. The user is asked to verify the values which have been chosen.

System Action: Processing continues.

User Response: Verify the values displayed in the following or preceding messages. You will have an opportunity to choose to continue or stop processing.

ARI6070I User ID SQLDBA password not restored.

Explanation: The password for SQLDBA was not restored during processing. The password has been set to the default password 'SQLDBAPW'.

System Action: Processing continues.

User Response: Use the GRANT command to change the password. Refer to the *DB2 Server for VM System Administration* manual for instructions.

ARI6071I Proceeding to do a complete DB2 Server for VM system link edit.

Explanation: This is the expected response from the database manager. The message is issued prior to performing a complete DB2 Server for VM system link edit.

System Action: Processing continues.

ARI6072E ARISQLLD LOADLIB &1 **could not be copied to ARISQLLD LOADLIB** &2**. COPYFILE Return Code =** &3**.**

Explanation: An error occurred when copying the ARISQLLD LOADLIB from file mode (&1) to file mode (&2). The CMS COPYFILE command failed with a return code of (&3).

System Action: Processing terminates.

User Response: Investigate the return code generated by the CMS COPYFILE command. Refer to the *VM/ESA: CMS Command Reference* manual for a detailed explanation of the return code. The ARISQLLD LOADLIB file may not exist on the old file mode specified, or the disk with file mode (&2) may not be accessed in WRITE mode.

ARI6073I ARISQLLD LOADLIB &1 **updated to latest service level.**

Explanation: This is the expected response from the database manager. The ARISQLLD LOADLIB file with mode (&1) has been updated to the latest service level.

System Action: Processing continues.

ARI6074I &1 &2 &3 **copied to** &1 &2 &4**.**

Explanation: This message is a completion message issued when a CMS COPYFILE command was successfully completed.

&1 is the file name of the file that was copied

&2 is the file type of the file that was copied

&3 is the file mode of the old file

&4 is the file mode of the new file

System Action: Processing continues.

ARI6075E COPYFILE command error occurred while copying *filename-1 filetype-1 filemode-1* **to** *filename-1 filetype-2 filemode-2***. COPYFILE Return Code =** *return-code***.**

Explanation: An error occurred when copying the file with *filename-1* from *filetype-1 filemode-1* to *filetype-2 filemode-2*. The return code from the COPYFILE command was *return-code*.

System Action: Processing ends.

ARI6076I Proceeding to update the DB2 Server for VM service minidisk.

Explanation: This is the expected response from the database manager. The service minidisk is being updated.

System Action: Processing continues.

ARI6080I Note: The DB2 Server for VM service minidisk file &1 **MACRO has been updated to identify the minidisks used for the database manager installation.**

Explanation: This is the expected response from the database manager. The file (&1) MACRO has been updated to indicate the minidisks used for the database manager installation. &1 is ARISPIDC if the full product is being installed.

System Action: The processing continues.

ARI6081E **An error occurred while the operating system was copying the** &1 **MACRO file from the service minidisk to the production minidisk.**

Explanation: An error occurred when copying the &1 (ARISPIDC if the full product is being installed) MACRO from the DB2 Server for VM service minidisk to the DB2 Server for VM production minidisk.

System Action: The processing is terminated.

User Response: Refer to the *VM/ESA: CMS Command Reference* manual for an explanation of the CMS COPYFILE error. The file (&1) MACRO may not be on the DB2 Server for VM service minidisk or the DB2 Server for VM production minidisk may not be accessed in WRITE mode.

ARI6082I **Subsequent DB2 Server for VM service processing required:**

Explanation: This is the expected response from the database manager. This message is displayed during the application of corrective service. The messages following it will indicate to which components corrective service has been applied and what subsequent processing is required.

System Action: The processing continues.

User Response: Refer to the *DB2 Server for VM Program Directory* for instructions on the application of corrective service.

ARI6084I &1 - &2 **installation and verification completed successfully.**

Explanation: This is the expected response from the database manager. The product specified was verified and installed successfully.

&1 is the product identifier for the database manager.

&2 is the product name for the database manager.

System Action: The processing continues.

ARI6086I **User requests exit from** &1 **EXEC.**

Explanation: &1 is an EXEC name. This is the expected response from the database manager. You requested to stop the process from the EXEC specified by (&1).

System Action: The processing is terminated.

ARI6087I **The following are the default DB2 Server for VM system minidisk definitions as described in the** *DB2 Server for VM Program Directory* **:**

Explanation: This is the expected response from the database manager. The following messages will display the default system minidisk definitions for the DB2 Server for VM installation.

System Action: The processing continues.

ARI6088D **This EXEC will try linking to** &1 &2 **with** &3 **mode. Do you want to force** &1**? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: The EXEC will attempt to link to a minidisk address specified by (&2) and owned by (&1) in the mode specified by (&3). If the minidisk is currently linked you will be asked if you wish to force the user (&1) to release the minidisk.

System Action: The processing continues.

User Response: You must choose to: QUIT the installation procedure, not to force the user to release the minidisk, or to force the user to release the minidisk. If you do not force the user the EXEC will not be able to link in the mode specified by (&3) to the minidisk.

ARI6089I **CP FORCE command failed to force a logoff of** &1**. If command still fails after two attempts, check installation user ID CLASS privileges.**

Explanation: The FORCE command failed because the userid being used to install the database manager does not have sufficient class privileges to force a user.

System Action: The processing is terminated.

User Response: Verify that you have sufficient privileges to force a userid, or re-execute the process at a time when the minidisk needed is not being used.

ARI6090E **VMFPLC2 command failed. Return Code =** &2**. DB2 Server for VM** &1 **minidisk load failed.**

Explanation: The VMFPLC2 tape command failed with the return code (&2). The minidisk specified &1 (service or production) was not loaded.

System Action: The processing is terminated.

User Response: Re-issue the command. Verify that the tape is positioned properly. The tape drive may need to be cleaned.

ARI6091I **DB2 Server for VM system minidisks are successfully loaded. Continue with the DB2 Server for VM installation process described in the DB2 Server for VM Program Directory.**

Explanation: This is the expected response from the database manager. The DB2 Server for VM service and production minidisks have been successfully loaded. Continue with the installation process described in the *DB2 Server for VM Program Directory*.

System Action: The processing continues.

ARI6092I **DB2 Server for VM system minidisks have not been successfully loaded. Do not continue with the installation process. Please correct the error identified by previous messages and/or review the DB2 Server for VM installation procedure. Then, reissue the command** &1**.**

Explanation: An error occurred during the installation process. The system minidisks (service and production) were not loaded successfully. &1 is a command.

System Action: The processing is terminated.

User Response: Do not continue with the DB2 Server for VM installation process. Review the installation procedure and correct any errors indicated by the preceding messages. Then re-issue the command specified by (&1).

ARI6093E **Error forwarding over tape file** &1**. The return code is** &2 **from the CMS VMFPLC2 FSF command.**

Explanation: The VMFPLC2 tape command failed with the return code (&2). The VMFPLC2 FSF command failed while forwarding over the tape file specified by (&1).

System Action: The processing is terminated.

User Response: Re-issue the command. Verify that the tape is positioned properly. The tape drive may need to be cleaned. Refer to the *VM/ESA: CMS Command Reference* manual for the return code from the VMFPLC2 command.

ARI6094I **DB2 Server for VM corrective service processing was unsuccessful. Correct the errors identified by previous console or virtual print file messages and rerun this EXEC.**

Explanation: The corrective service EXEC failed due to conditions specified in previous messages.

System Action: The processing is terminated.

User Response: Retrieve the console log and the virtual print file and correct any errors encountered. Then re-issue the command. If the errors cannot be corrected, contact your designated support group.

ARI6095I **Continue checking link books.**

Explanation: This is the expected response from the database manager. The link books are being checked.

System Action: The processing continues.

ARI6096E **Error reading the contents of DB2 Server for VM link book:** &1 **TEXT** &2**. EXECIO Return Code =** &3**.**

Explanation: The EXECIO command failed with a return code of (&3). The EXEC was unable to read the contents of the link book (&1) on the minidisk accessed as (&2).

System Action: The processing is terminated.

User Response: Determine the cause of the EXECIO command failure by consulting the *VM/ESA: CMS Command Reference* manual. Correct the problem and re-issue the command.

ARI6097E **Error reading ARISPDEC MACRO** &1**. EXECIO Return Code =** &2**.**

Explanation: The EXECIO command failed with a return code of (&2). The EXEC was unable to read the contents of the ARISPDEC MACRO on the minidisk accessed as (&1).

System Action: The processing is terminated.

User Response: Determine the cause of the EXECIO command failure by consulting the *VM/ESA: CMS Command Reference* manual. Correct the problem and re-issue the command.

ARI6098I **You have chosen to abort the maintenance process for the CMS HELP text for language** &1**.**

Explanation: The user has chosen to abort the service to the language specified by &1. (&1 is the Language Key)

System Action: Processing continues.

ARI6099I **Proceeding to check for and do required DB2 Server for VM production minidisk file updates.**

Explanation: This is the expected response from the database manager.

ARI6100D **{service | production} minidisk owner =** owner-id **{service | production} minidisk address =** virtual-address **{service | production} minidisk password =** minidisk-write-password **Do you want to use these values? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: This is the expected response from the database manager.

System Action: If you reply `0` for No, the system will prompt you for the owner id, virtual address and minidisk write password using ARI6101D. If you reply 1 for Yes, the values shown in this message will be

used. If you reply 111 for Quit, processing will terminate.

User Response: You must verify the owner id, virtual address and minidisk write password of the minidisk (service or production) that you wish to use. If the response is 0 for No, ARI6101D will prompt you for your input. If the response is 1 for Yes, the values shown will be used. If the response is 111 for Quit, processing will terminate. Enter the appropriate response.

ARI6101A **Enter** &1 &2 **(or a null for existing value).**

Explanation: This is the expected response from the database manager when you replied 0 for No to ARI6100D. The following pairs of values will be substituted for &1 and &2:

&1	&2
USER	ID
MINIDISK	ADDRESS
SERVICE	PASSWORD
PRODUCTION	PASSWORD

System Action: If you enter a response, that value will be used instead of the default value shown in ARI6100D for owner id, virtual address, or write password to the service or production minidisk. If you reply NULL, the default value shown in ARI6100D will be used instead.

User Response: You must either enter the owner id, virtual address or write password of the minidisk (service or production) that you wish to use or enter NULL to use the default previously shown in ARI6100D.

ARI6102A **Enter DBEXTENT number (or LOGDSK1, LOGDSK2, or BDISK) to copy. (Enter a null response to end input or enter QUIT to exit.)**

Explanation: COPY DBEXTENT (SQLCDBEX) needs to know the minidisk to copy. If you want to copy a DBEXTENT, enter the DBEXTENT number. If the input is LOGDSK1, LOGDSK2 or BDISK, the program copies log disk 1, log disk 2, or the directory disk respectively. You can get information for the DBEXTENTs and their numbers by issuing the SHOW POOL command. Enter a null line to end the prompt for input.

System Action: The system waits for the user's reply.

System Programmer Response: Enter the DBEXTENT number, LOGDSK1, LOGDSK2, BDISK or a null line.

ARI6103A **Enter virtual address for new** copy-disk**. (Enter a null response to end input or enter QUIT to exit.)**

Explanation: COPY DBEXTENT (SQLCDBEX) requires the virtual address of the output minidisk. Depending on how you reply to the message ARI6102A, *copy-disk* can be a DBEXTENT number, LOGDSK1, LOGDSK2, or BDISK.

System Action: The system waits for the user's reply.

System Programmer Response: Enter the virtual address for a minidisk that is already defined.

ARI6104E **The** in-out **disk is either not on a reserved disk, is not fixed-length, or has an invalid blocksize. Reason Code =** n**.**

Explanation: A DBEXTENT, log disk or directory disk must reside on a disk that you reserve by issuing the CMS RESERVE command and format for fixed-length files. The blocksize for DBEXTENTs and log disks must be 4096 bytes, and 512 bytes for the directory. The input or output minidisk does not satisfy these conditions. *In-out* is input or output. The error is signified by the value of *n* where *n* is one of the following:

1. More than one file resides on the minidisk.
2. The file mode number is not 6.
3. The file is not a fixed-length file.
4. The blocksize is invalid.

System Action: Processing ends.

System Programmer Response: Check the minidisks. Correct the input and try again.

ARI6105D **Output minidisk** cuu **(**nblock **blocks of blocksize =** blksize**) is larger than the input minidisk (**mblock **blocks). Do you want to continue? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: The output minidisk is larger than the input minidisk. This message indicates the size of both disks. For devices of different types, the database manager may not be able to duplicate the amount of space exactly. Therefore, if the difference is small, consider continuing. If you want to increase the size of the DBEXTENT, use the ADD and DELETE DBEXTENT functions to do so.

cuu is the virtual address of the output minidisk.

nblock is the size of the output minidisk in blocks.

blksize is the block size.

mblock is the size of the input minidisk.

System Action: Processing continues or ends based on your reply.

System Programmer Response: Decide whether to continue. Enter your reply.

ARI6106E **Output minidisk** cuu **(**nblock **blocks of blocksize =** blksize**) is too small to hold the minidisk to be copied. Rerun the EXEC and specify another minidisk that has at least** mblocks **blocks. The old minidisk remains in use.**

Explanation: The output minidisk must be at least as large as the input minidisk to hold the DBEXTENT, log disk, or directory. The database manager continues to use the old minidisk.

cuu is the virtual address of the output minidisk.

nblock is the size of the output minidisk in blocks.

blksize is the block size.

mblock is the size of the input minidisk.

System Action: Processing ends.

System Programmer Response: Specify another minidisk of a larger size if one exists; otherwise, define one. Rerun the EXEC. Alternatively, if the SHOW POOL command indicates that the storage pool has sufficient space, you can use the ADD and DELETE DBEXTENT functions to perform the copying.

ARI6107E **An error occurred in COPY DBEXTENT. The output file is not identical to the input file. The** dbname **SQLFDEF file remains unchanged. The old minidisk remains in use.**

Explanation: COPY DBEXTENT processing failed. The file copied to the output minidisk is not identical to the input file. A disk I/O error may have occurred. The *dbname* SQLFDEF file remains unchanged. The database manager continues to use the old disk.

System Action: Processing ends.

System Programmer Response: Rerun the EXEC with the proper input or follow your installation's problem determination procedure.

ARI6108I **Minidisk copied successfully. The** &1 **SQLFDEF file will be updated.**

Explanation: The database manager has successfully copied the DBEXTENT, log disk or directory. The program proceeds to update the &1 SQLFDEF file.

System Action: Processing continues.

System Programmer Response: If this message is followed by the message ARI6109I, no action is required. Otherwise, correct the cause of the error and run the EXEC again.

ARI6109I dbname **SQLFDEF file has been updated on the A disk.**

Explanation: The database manager has successfully updated the *dbname* SQLFDEF file on the user's A disk.

System Action: Processing continues.

System Programmer Response: No action is required.

ARI6110D **Disk** cuu **is already formatted. Continuing will erase all data on this disk. Do you want to use the disk? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: You have specified a formatted disk to hold a copy of a DBEXTENT.

System Action: Processing continues. If the reply is yes, the program uses the disk you specified. If the reply is no, the program prompts you to ignore this entry and prompts for new input. If the reply is quit, the program terminates.

System Programmer Response: Decide whether to continue. Enter the reply.

ARI6111A **Enter action (ADD or DELETE) to be taken. (Enter a null response to end input or enter QUIT to exit.)**

Explanation: You are prompted to choose an action for a DBEXTENT. You can enter QUIT to end the ADD or DELETE processing. If the first character of the response is an asterisk (*), the response is a comment that the function adds to the *dbname* SQLADBEX file.

System Action: The system waits for you to enter a reply.

System Programmer Response: Enter the required response.

ARI6112A **Enter DBEXTENT number to use for the new extent. The default is** extno**. (Enter a null response to use the default value or enter QUIT to exit.)**

Explanation: You can reuse a DBEXTENT number if it was deleted previously. Enter the DBEXTENT number if desired. The default is the number of DBEXTENTs currently defined plus one. *extno* is the default DBEXTENT number that the EXEC will use. Use the SHOW POOL command to determine the deleted DBEXTENT numbers. You can terminate the ADD or DELETE processing by entering QUIT.

System Action: The system waits for you to enter a reply.

System Programmer Response: Enter the required response.

ARI6113A **Enter DBEXTENT number to delete. (Enter QUIT to exit.)**

Explanation: Enter the DBEXTENT number that you want to delete. If necessary, use the SHOW POOL command to determine the DBEXTENT numbers. You can terminate the ADD/DELETE processing by entering quit.

System Action: The system waits for you to enter a reply.

System Programmer Response: Enter the required response.

ARI6114A **Do you want to do a database archive (ARCHIVE), user archive (UARCHIVE), or no archive (NOARCHIVE) at the end of the run? (Warning: Database may not be restorable if you choose NOARCHIVE.) Enter one of the values or enter a null response to use the default (ARCHIVE).**

Explanation: You cannot restore a database from an archive if you have removed a DBEXTENT from the database after taking the archive. Choosing NOARCHIVE is not recommended.

System Action: The system waits for you to enter a reply.

System Programmer Response: Enter the option desired.

ARI6115E **DBEXTENT number** extno **to be deleted is not in the database.**

Explanation: The database does not contain the DBEXTENT number *extno* that you specified for deletion.

System Action: Program continues. The program will prompt you for a new action.

System Programmer Response: Enter the correct values when prompted.

ARI6117E **DBEXTENT number** extno **is at virtual address** cuu **in the database.**

Explanation: The DBEXTENT number *extno* that you entered for the ADD action is already in the database at the virtual address *cuu*.

System Action: The program continues. The program prompts you for a new action.

System Programmer Response: Enter the correct values when prompted.

ARI6118I **Formatting in progress. Please wait...**

Explanation: The minidisk is being formatted.

System Action: Processing continues.

System Programmer Response: No action is required.

ARI6119E copy-disk **has already been copied from** old-cuu **to** new-cuu**. It can be copied only once.**

Explanation: The minidisk *copy-disk* has already been copied during this invocation of the SQLCDBEX EXEC. A minidisk cannot be copied twice in the same invocation.

System Action: Processing continues. SQLCDBEX will prompt you to specify another minidisk.

System Programmer Response: Enter the appropriate response when prompted.

ARI6120I **The following** *component* **flag file was created or replaced:** *filename filetype filemode***.**

Explanation: A new flag file enabling either DRDA or DSS has been created. *filename* is the filename of the flag file, *filetype* is the filetype of the flag file, *filemode* is the filemode of the flag file. Valid names for the DRDA flag files are SQLDAR MACRO and SQLDAS MACRO. The name of the DSS flag is SQLVMD MACRO.

System Action: ARISDBMA EXEC processing continues.

ARI6121I **The** *component* **flag file** *filename filetype filemode* **erased successfully.**

Explanation: A new flag file enabling either DRDA or DSS has been erased. *filename* is the filename of the flag file, *filetype* is the filetype of the flag file, *filemode* is the filemode of the flag file. Valid names for the DRDA flag files are SQLDAR MACRO and SQLDAS MACRO. The name of the DSS flag is SQLVMD MACRO.

System Action: ARISDBMA EXEC processing continues.

ARI6122E **Error when erasing** *component* **flag file** *filename filetype filemode***. Return Code =** *rc***.**

Explanation: A flag file enabling DRDA or DSS could not be erased. *filename* is the filename of the flag file, *filetype* is the filetype of the flag file, *filemode* is the filemode of the flag file, *rc* is the return code of the erase command.

System Action: ARISDMBA EXEC processing continues.

User Response: If the specified file is located on the

service disk but cannot be erased, contact your system programmer. If it is not there, continue with the procedure of disabling DRDA or DSS as outlined in the *DB2 Server for VM System Administration* manual.

ARI6126D **Make any desired changes to the input file.**

Explanation: You can edit the input file.

System Action: The system waits for your response.

User Response: Make any changes or corrections you want to the input file, and follow the instructions given by the next message that displays.

ARI6127I **Use PF3 or PF15 to continue.**

Explanation: This is the instruction to continue processing.

System Action: The system waits for your response.

User Response: Press PF3 or PF15 when you want to continue processing.

ARI6128I **No help is available for** exec-name**.**

Explanation: There is no HELP available for this EXEC.

System Action: The processing is terminated.

User Response: Look in the *DB2 Server for VSE & VM Master Index and Glossary* manual to find the appropriate publication for an explanation of this EXEC.

ARI6129D **COLDLOG will initialize the log on disks** *cuu1* **and** *cuu2***. Reformatting will erase only log data. Reconfiguring will erase both log data and the log history. Warning: If you are running LOGMODE=L, reconfiguring invalidates previous or current log archive restore sets unless this log history is restored before the next COLDLOG. Enter CANCEL to cancel this COLDLOG or CONTINUE to continue.**

Explanation: This message is issued from the SQLLOG EXEC. It describes the purpose of a coldlog and its effects on the log disks identified by their virtual device addresses (cuu1 and cuu2). In the case of dual logging, coldlog will be performed on both the primary and secondary logs.

Attention: If you are running with LOGMODE=L and you decide to reconfigure, you will not be able to restore the log archives that were taken prior to this coldlog unless you have a backup copy of the log history to be erased. In order to continue running with LOGMODE=L, you will need to take a database archive after performing this coldlog. For more information, refer to the *DB2 Server for VM System Administration* manual.

System Action: Processing continues or ends, based on your reply.

Operator Response: Decide whether to reply CANCEL or CONTINUE. Enter your reply.

ARI6131I **Copying in progress. Please wait...**

Explanation: The minidisk is being copied.

System Action: Processing continues.

System Programmer Response: No action is required.

ARI6132I **Now copying ARISQLLD LOADLIB.**

Explanation: The DB2 Server for VM load library ARISQLLD LOADLIB is being copied.

System Action: Processing continues.

ARI6133E copy-disk **to be copied is not in the database.**

Explanation: The minidisk *copy-disk* that you wanted to copy cannot be found in the database.

System Action: Processing continues. The program will prompt you for new input.

System Programmer Response: Enter the correct input when you are prompted.

ARI6134A **Please enter NETID parameter.**

Explanation: You indicated to the system that you wanted to specify a NETID value. The system prompts you for that value.

System Action: The system waits for a user response.

User Response: Enter a valid NETID value. It must be a 1- to 8- character string comprised of uppercase letters(A-Z), numerals (0-9) and special characters ($, #, and @). The first character must not be numeric.

ARI6135D **DBEXTENT number** extno **at virtual address** cuu **is in the database. Do you want to move it to another pool? Enter 0(No) or new storage pool number of DBEXTENT** extno.

Explanation: The DBEXTENT number that you want to add already exists in the database. If you want to move the DBEXTENT from one pool to another, supply

the new storage pool number. Otherwise, reply no.

System Action: The program continues. If you respond with a number greater than zero, the program adds an entry to the input file and proceeds without formatting the minidisk. If the reply is no, the program prompts for a new action.

System Programmer Response: Enter the required reply.

ARI6138E **Keyword** &1 **is not allowed with keywords already specified.**

Explanation: The CMS EXEC has multiple keywords specified that are in conflict with each other. Only one of the keywords or sets of keywords can be specified.

System Action: The processing is terminated.

User Response: Try the EXEC with only one of the keywords specified.

ARI6139E **Keyword** &1 **should have a numeric value in the range from** &2 **to** &3**.**

Explanation: The CMS EXEC has a numeric value that is incorrect for the specified keyword. The valid range is from &2 to &3.

System Action: The processing is terminated.

User Response: Try the EXEC with a numeric value that is in the specified range.

ARI6140I **Warning: Invalid keyword** &1 **in the file** filename filetype filemode**.**

Explanation: The file that has keyword &1 specified has an incorrect value. This parameter is required by the current CMS EXEC being invoked.

System Action: The processing is terminated.

User Response: Issue the EXEC that creates the file with the invalid keyword, then try the current EXEC exec again.

ARI6141E **Insufficient free storage.**

Explanation: The COPY DBEXTENT function requires at least 40K of virtual storage. The virtual machine does not have sufficient free storage.

System Action: Processing ends.

System Programmer Response: Redefine storage for the virtual machine to have at least 40K of free storage and rerun the EXEC.

ARI6142E **No user ID is in the** dbname **SQLFDEF file.**

Explanation: SQLCDBEX found the indicated file on the DB2 Server for VM production minidisk but did not find a user ID in that file. If no input error occurred, the database may be corrupted. *dbname* is the name of the database.

System Action: Processing ends.

System Programmer Response: Follow your installation's problem determination procedure.

ARI6143I **Warning: DBEXTENT number** extno1 **specified is out of sequence. The default value** extno2 **is used.**

Explanation: You must add new DBEXTENTs to the database in sequence. DBEXTENT number *extno1* is out of sequence. The function used the next available number *extno2* instead.

System Action: The program continues.

System Programmer Response: No action is required.

ARI6145D **Do you want to review the SQLADBEX file? You will not be able to modify this file. Enter 0(No) or 1(Yes).**

Explanation: The system allows you to review the input file created before starting up the database manager machine. No modification to the file is accepted.

System Action: The program continues. If you choose yes, the VM Editor is invoked. However, no changes to the input file are saved. If you choose no, the VM Editor is not invoked.

System Programmer Response: Decide if you want to review the file. Enter the reply.

ARI6146D **Are you expanding the DB2 Server for VM directory? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: You can use SQLCDBEX to expand the DB2 Server for VM directory while SQLCDBEX copies the directory to a larger minidisk.

System Action: Processing continues unless you enter the reply to quit. If you choose to expand the directory, the CMS command EXECIO performs the I/O.

System Programmer Response: Decide whether or not to expand the directory. Enter your reply.

ARI6147D **Do you want to continue the ADD/DELETE DBEXTENT process? Enter 0(No) or 1(Yes).**

Explanation: After reviewing the input file, you can stop processing before you start the machine.

System Action: The program continues or ends depending on the reply.

System Programmer Response: Decide whether or not to continue processing. Enter a reply.

ARI6148I **New disk** cuu **has not been formatted. Program will continue to format the disk before copying.**

Explanation: The new disk at virtual address *cuu* that the database manager will use to copy a DBEXTENT, or to copy or expand the directory, has not been formatted. The program proceeds to format the disk before copying.

System Action: Processing continues.

System Programmer Response: No action required.

ARI6149E **Pool number** poolno **is out of range.**

Explanation: The pool number *poolno* that you entered is less than 1 or greater than 999.

System Action: The program prompts you for new input.

System Programmer Response: Reply with the correct values when prompted.

ARI6151E **Keyword** *&1* **follows** *&2***. If you specify the** *&2* **parameter, it must follow the other** *&3* **parameters.**

Explanation: The format or the sequence of the parameters is incorrect.

System Action: Processing ends.

User Response: Correct the error identified by this message, and rerun the program.

ARI6154E *command* **failed with RETCODE =** *return-code***. REASON CODE =** *reason-code*

Explanation: An error occurred while processing *command*. *command* is a CMS command, CP command, or CSL routine. The *return-code* and *reason-code* are those set by the failing command or routine.

System Action: Processing ends.

User Response: Refer to the VM/ESA manual that describes *command* for the meaning of *return-code* and *reason-code*.

ARI6157E **DBEXTENT number** extno**, that you specified to delete is invalid. You cannot delete DBEXTENTs more than once in each execution of the ADD/DELETE function.**

Explanation: Because of the design of the database manager, you cannot delete a DBEXTENT more than once in a single execution of the ADD or DELETE DBEXTENT function, even though the DBEXTENT may have been added back to the database in the interim.

System Action: The program prompts you for new input.

System Programmer Response: Reply with the correct values when prompted.

ARI6160E **An incorrect release level, RELEASE** *&1***, was specified.**

Explanation: The version and release level that you specified for the database manager does not match the version and release level of the database manager.

System Action: Processing ends.

User Response: Specify the correct release level, and make sure that you are accessing the right database manager. Try again.

ARI6161I **The status shared segment successfully initialized.**

Explanation: The status shared segment has been initialized. All entry slots are now available for use.

System Action: This is the expected response from the database manager.

ARI6162E **The specified server name (***server-name***) is invalid.**

Explanation: The server *server-name* cannot be found.

System Action: Processing ends.

User Response: Specify the correct server name.

ARI6164A **Enter Service directory (or a null response for the existing value).**

Explanation: The name of the service SFS directory is requested.

System Action: The system waits for a response to the prompt.

User Response: Type the name of the service directory.

ARI6165A **Enter Production directory (or a null response for the existing value).**

Explanation: The name of the production SFS directory is requested.

System Action: The system waits for a response to the prompt.

User Response: Type the name of the production directory.

ARI6166E **Error occurred when accessing top directory of** &1**.**

Explanation: The top directory of the specified service or production directory could not be accessed successfully.

System Action: The EXEC terminates.

User Response: Ensure that the service and the production directories can be accessed. Rerun the EXEC.

ARI6167I **Begin loading the DB2 Server for VM production directory.**

Explanation: The DB2 Server for VM production files are being copied from the service directory to the production directory that you specified.

System Action: Processing continues.

ARI6168I **DB2 Server for VM production directory was loaded successfully.**

Explanation: The DB2 Server for VM production files have been successfully copied to the production directory.

ARI6169E **File** &1 &2 **was not found.**

Explanation: The above CMS file was not found on the system disk.

&1 is the file name VMFNLS and &2 is the file type LANGLIST.

System Action: Processing terminates.

User Response: Notify your system programmer.

System Programmer Response: Ensure that the National Language Support File Naming Conventions APARs were applied to your system.

ARI6170I **The VMFNLS LANGLIST does not contain the language identifier** lang-id**.**

Explanation: The VMFNLS LANGLIST CMS file contains the National Language Support file naming conventions. This file is on all VM systems. This file is required by the database manager to identify the language you are working with. If the language identifier is not found, your VM installation may not support this language.

System Action: Processing terminates.

User Response: Notify your system programmer.

System Programmer Response: Ensure that the National Language Support File Naming Conventions APARs were applied to your system.

ARI6171E **The package for** program-name **was not found.**

Explanation: The package for the program being executed cannot be found. *program-name* is the name of the program being executed.

System Action: The program ends unsuccessfully.

User Response: Repreprocess the program and rerun it. If the program has already been repreprocessed and the same problem occurs, contact the designated support group.

ARI6172I **The creation and load of the DB2 Server for VM directories have begun.**

Explanation: The database manager has begun loading the SFS directories.

ARI6173I **Begin loading the DB2 Server for VM** &1 **directory.**

Explanation: The database manager is loading a production or service directory.

ARI6174I **The DB2 Server for VM** &1 **directory was loaded successfully.**

Explanation: The database manager has successfully loaded the SFS directories.

ARI6175E **The DB2 Server for VM production directory load failed.**

Explanation: The downloading on to the specified production directory was not successful.

System Action: The EXEC terminates. Rerun it.

User Response: Check any previous messages. Ensure that the production directory has the proper access mode. Rerun the EXEC.

ARI6177E **There is no segment defined for SYSNAME** name **in the current VM environment.**

Explanation: The ARISTAT EXEC is unable to find the DMKSNT entry (or the saved segment definition) for the SYSNAME specified in the message.

System Action: The ARISTAT EXEC processing is ended.

Programmer Response: Specify the entry for SYSNAME in the DMKSNT (or the saved segment definition is done for SYSNAME). Reissue the ARISTAT EXEC.

ARI6178I **CCSID CMS files generated successfully.**

Explanation: The system informs you that the CCSID CMS files are successfully created or updated.

ARI6180I **The following users hold a write link to disk** &1**. USER ID ADDRESS LINKED TO _____ _____**

Explanation: When displayed, this message is always followed by one or more ARI6181I messages. It indicates that the product has checked the links to the disk where the requested archive is to be written. There exists at least one user other than the database machine holding a write link.

System Action: The system displays one or more copies of message ARI6181I.

ARI6181I &1 &2**.**

Explanation: This message identifies:

- Userid of the person who still has a write link to the disk specified in message ARI6180I
- Address of the disk to which this person is linked.

System Action: After displaying one or more user IDs, the product will prompt the operator through ARI6183D or ARI6185D.

ARI6182A **Too many users are linked to disk** &1 **to produce a valid report. Detach some links.**

Explanation: Too many links exist, including read-only and read-write, to the disk where the requested archive is to be written. The result returned by the CP QUERY LINKS command is too large for the DB2 Server for VM product to display.

System Action: The product checks for write links in the truncated result. If they do, the links are displayed through messages ARI6180I and ARI6181I. Message ARI6182A is always followed by message ARI6183D or ARI6185D.

Operator Response: To get a complete result, the operator can issue a #CP QUERY LINKS *daddr* command where *daddr* is the disk address specified in the message on the database manager operator console. The operator should then request that user IDs holding a write link, release and detach the disk.

If this message redisplays after all the write links are detached and message ARI6183D has been answered with READY or `0`, drop some of the read-only links.

System Programmer Response: Ensure that a limit has been established on the total number of links to the disk that contains the log archives.

ARI6183D **Ensure each user detaches all write links before proceeding.**

Enter 0(Ready) - ready to proceed
1(Quit) - switch to other medium

Explanation: This message appears whenever:

- The DB2 Server for VM product detects write links to the disk where the requested log archive is to be written. These links are displayed through messages ARI6180I and ARI6181I.
- Too many User IDs are linked to the disk. This message is preceded by message ARI6182A.

System Action: The product waits for the operator to reply. If the reply is valid, the product proceeds; otherwise, this message is redisplayed.

Operator Response: Type Ready or `0` when this disk is ready to be used and all of the write links identified by message ARI6181I have been dropped. If this message is preceded by message ARI6182A, some of the read-only links may need to be detached too. Type Quit or 1 to change the log archive medium. The DB2 Server for VM product then prompts for the new medium.

System Programmer Response: If it is difficult to notify those who have links that need to be dropped, decide quickly whether or not to drop the links or to switch to another medium. (Links can be detached by a user who has class B privilege.) The product remains suspended until either the links have been dropped or a switch has been made to another medium.

ARI6184D **Do you want to continue the ADD DBSPACE process? Enter 0(No) or 1(Yes).**

Explanation: After reviewing the input file, you can stop processing before you start the machine.

System Action: Depending on your reply, processing continues or ends.

System Programmer Response: Decide whether to continue processing or not. Enter a reply.

ARI6185E **Server name &1 is larger than 8 bytes and no RESID was specified.**

Explanation: In order to use a server name larger than 8 bytes a RESID is required.

System Action: Processing terminates.

System Programmer Response: Specify a RESID on

the SQLDBINS EXEC if the server name is larger than 8 bytes. Otherwise correct the server name and retry.

ARI6186D **Output minidisk** cuu **of size** total-outdisk-size **(**outblock **blocks of blocksize** outblock-size**) is larger than the required minidisk size** total-required-size **(**required-number **blocks of blocksize** outblock-size**). Do you want to continue? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: The output minidisk is larger than the minimum required size. This message indicates the size of the output disk and also the required size. For devices of different types, the database manager may not be able to duplicate the amount of space exactly. Therefore, if the difference is small, consider continuing. If you are copying dbextents and you want to increase the size of the dbextent, use the ADD or DELETE DBEXTENT functions to do so. If you are copying the directory and you want to increase the size of the directory, use the expand directory option when prompted.

cuu is the virtual address of the output minidisk.

total-outdisk-size is the size of the output minidisk.

outblock is the number of blocks that make up the output minidisk.

outblock-size is the block size for the output minidisk.

total-required-size is the minimum size of the output minidisk.

required-number is the number of blocks that make up the minimum size.

System Action: Processing continues or ends based on your reply.

System Programmer Response: Decide whether to continue. Enter your reply.

ARI6187E **Output minidisk** cuu **of size** total-outdisk-size **(**outblock **blocks of blocksize** outblock-size**) is too small to hold the minidisk to be copied. Rerun the SQLCDBEX EXEC and specify another minidisk that has at least** min-blocks **blocks of blocksize** desired-block-size**. The old minidisk remains in use.**

Explanation: The output minidisk must be at least as large as the input minidisk to hold the dbextent, log disk or directory. The database manager continues to use the old minidisk.

cuu is the virtual address of the output minidisk.

total-outdisk-size is the size of the output minidisk.

outblock is the number of blocks that make up the output minidisk.

outblock-size is the block size for the output minidisk.

min-block is the minimum number of blocks needed in the output minidisk.

desired-block-size is the optimum block size of the output minidisk.

System Action: Processing ends.

System Programmer Response: Specify another minidisk of a larger size if one exists; otherwise, define one. Rerun the EXEC. Alternatively, if the SHOW POOL command indicates that the storage pool has sufficient space, you can use the ADD and DELETE DBEXTENT functions to perform the copying.

ARI6188A **Enter the output block size of the directory. Enter 512 or 4096, or a null response to use the original size, or 111(Quit) to exit.**

Explanation: To copy the directory, it is required to know the block size of the output minidisk. To input a null, hit enter without typing any other keys. If the input is 512 or 4096, the program copies the directory to a minidisk of block size 512 or 4096 bytes respectively. If the input is null, the block size of the input directory is used. If the input is 111 or Q or QUIT, the program terminates.

System Action: The database manager waits for the user's reply.

System Programmer Response: Enter the size of directory to define an output file.

ARI6189I **Initiating an HSDS archive on the A-disk.**

Explanation: The database manager is attempting to write the current log history area (HSDS) to the A-disk.

System Action: Since the HSDS archive is done implicitly by the database manager, this message appears only when it detects an error on the A-disk. The error will then be displayed and followed by message ARI6190D.

ARI6190D **Ensure A-disk is linked as R/W by the DB2 Server for VM application server.**

Enter 0(Ready) - ready to proceed
1(Quit) - cancel the archive.

Explanation: This message appears whenever:

1. The database manager detects write links to the A-disk. These links are displayed through ARI6180I and ARI6181I.

2. Too many User IDs are linked to the disk. It will be preceded by message ARI6182A.
3. There are disk errors, such as disk does not exist. These will be displayed by system messages.

System Action: The database manager waits for the operator reply. If the reply is valid, it will proceed; otherwise, this message will be redisplayed.

Operator Response: Reply Ready or `0` when all the write links identified by message ARI6181I have been dropped. If this message is preceded by ARI6182A, some of the read-only disks may need to be detached too. If the A-disk is detached, re-link the disk by entering #CP LINK on the operator console. Reply QUIT or 1 to cancel the HSDS archive only when you cannot fix the disk problem.

System Programmer Response: If it is difficult to notify those who have links that need to be dropped, decide quickly whether or not to drop the links or to cancel the archive. (Links can be detached by a user who has class B privilege.) The database manager remains suspended until either the links have been dropped or the archive cancelled. Make sure that all the disks that are defined for DB2 Server for VM configuration are never detached. Be sure that you understand the consequences of replying 1(QUIT) which may cause the database manager to lose valuable information for restoration. So after cancelling the HSDS archive, you should bring down the machine and fix the disk problem. If it still cannot be resolved, contact the designated support group.

ARI6191D **Do you want to modify the** filename filetype **file? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: The named file (*filename filetype*) will be used as input to the database generation, DBGEN, process. You are given a chance to change the file before the actual process begins.

System Action: The system waits for your reply. If you reply 1, Y or YES for yes, the VM Editor will be invoked. If you reply 0, N or NO for no the VM Editor will not be invoked, and processing will continue. If you reply 111, Q or QUIT to quit, the database generation process will be terminated.

User Response: Decide whether to change the file. Enter your reply.

Note: It is generally recommended that you reply affirmatively (enter 1, Y or YES) to review the file and to revise it as needed.

ARI6192D **The old BDISK is blocked at 4 K. The new BDISK will be reblocked at 512 bytes. Do you want to continue? Enter 0(No), 1(Yes), or 111(Quit).**

Explanation: The blocksize of the directory disk (BDISK) that is being copied is 4096 bytes (4 K). This directory blocksize is used only when DB2 VM Data Spaces Support is present. The database manager has determined that DB2 VM Data Spaces Support is not present and therefore, if you decide to continue, will block the new directory disk at 512 bytes.

System Action: Processing continues or ends based on your reply.

System Programmer Response: Decide whether to continue. Enter your reply.

ARI6193D **Begin installing DB2 DSS.** *product-id* **-** *product-name* **Version** *version-number* **Release** *release-number* **Modification** *mod-number***. Do you want to install this product now? Enter 0(No), or 1(Yes).**

Explanation: This is the expected response from the DB2 DSS install in the application server. The specified level of the application server indicated by *version-number release-number mod-number* is the level on which the DB2 DSS will be installed. You are asked if you wish to continue or not.

System Action: The processing continues or terminates based on your reply.

User Response: Decide whether or not to install the DB2 DSS at this time. Enter your reply.

ARI6197E **Cannot copy** *copydisk* **to** *cuu* **because size and type of output disk is not identical to size and type of input disk. Input disk type is** *type1* **and size is** *n cylinders***. Output disk type is** *type2* **and size is** *m cylinders***. Cannot use the SQLCDBEX EXEC to change the size of a log disk. Use the SQLLOG EXEC instead.**

Explanation: This message is displayed when a log disk, *copydisk* is LOGDSK1 or LOGDSK2, is being copied to new disk, *cuu*, that is not the same size or type as the input disk.

System Action: The SQLCDBEX EXEC ends. No changes have been made.

User Response: If the log disk must be replaced with a new disk on a different type of DASD or to a different size of disk, the SQLLOG EXEC must be used to perform a COLDLOG to do log reconfiguration. Otherwise, find a new disk such that the type and size in cylinders matches the type and size of input disk.

ARI6198D

Current maximum DBEXTENT pages :
nnnnnnnnnn
New maximum DBEXTENT pages :
nnnnnnnnnn
DBEXTENT pages added :
nnnnnnnnnn
Current maximum DBSPACE pages :
nnnnnnnnnn
New maximum DBSPACE pages :
nnnnnnnnnn
DBSPACE pages added :
nnnnnnnnnn
Current directory size (blocks) :
nnnnnnnnnn
New directory size (blocks) :
nnnnnnnnnn
Do you wish to continue with expanding the directory to allow the directory to hold additional DBSPACE and DBEXTENT pages ?
Enter 0(No), 1(Yes) or 111(Quit)

Explanation: This message shows the maximum number of DBSPACE and DBEXTENT pages that the old directory and the new directory can hold, as well as the number of additional DBSPACE and DBEXTENT pages that will be available in the new directory. You have the option to continue with the expansion or to cancel the operation.

System Action: The program continues or ends depending on the reply.

System Programmer Response: Decide whether to continue processing or not. Enter a reply.

ARI6199E **The new directory disk is not large enough to hold the previously expanded page map table and the new allocation bitmaps.** *nnnnnnnnnn* **additional directory blocks of blocksize** *nnnn* **are required.**

Explanation: The directory has been expanded previously and the new directory is not large enough to hold the previously expanded directory plus the new allocation bitmaps to be added. Refer to the *DB2 Server for VM System Administration* for a complete explanation.

System Action: Processing ends.

System Programmer Response: This message indicates how much additional space is required. Specify a larger minidisk and re-run the EXEC. Refer to the *DB2 Server for VM System Administration* to determine the required size of the new minidisk.

ARI6200D **You have requested to expand the directory. Enter 1 to expand the directory to hold more DBSPACE pages. Enter 2 to expand the directory to hold more DBSPACE and DBEXTENT pages. Enter 111 to quit. Enter 1,2 or 111(Quit).**

Explanation: If you wish to increase the size of the page map table to allow more DBSPACE pages to be added, enter 1. If you wish to increase the size of the allocation bitmaps and the page map table concurrently to allow more DBEXTENT and DBSPACE pages to be added, enter 2. You have the option to continue with the expansion or to cancel the operation.

System Action: Processing continues or ends based on your reply.

System Programmer Response: Decide to expand the directory to hold more DBSPACE pages only or to expand the directory to hold more DBSPACE and DBEXTENT pages or to quit the process. Enter your reply.

ARI6201I **Directory expansion completed successfully. It is strongly recommended that a database archive be taken immediately.**

Explanation: The directory expansion process completed successfully. It is strongly recommended that a database archive be taken immediately. Any updates made to the database after the directory expansion may not be applied correctly during a restore if a database archive taken before the directory expansion is used as the starting point of the restore set. Refer to System Administration, Enhanced Directory expansion for a complete explanation.

System Action: Processing ends.

ARI6202E **Pool Recovery is in progress, expansion not allowed.**

Explanation: The Data Restore storage pool recovery process was initiated, but has not yet been completed by starting the database manager with STARTUP=U. Storage Pool recovery must be completed before directory expansion may be done.

System Action: Directory expansion is cancelled. The original directory has not been changed.

Operator Response: Complete the storage pool recovery process by starting the database with STARTUP=U.

ARI6700I **Now invoking DB2 Server for VM to reload the SUPPLIER/PARTS sample tables.**

Explanation: The DB2 Server for VM SUPPLIER/PARTS sample tables are loaded.

User Response: Check that no message follows indicating that problems occurred during the downloading of these sample tables.

ARI6701I **Warning: An error occurred while reloading the SUPPLIER/PARTS sample tables. The sample tables may not be available while you are accessing this database. After the SQLDBMIG EXEC finishes, refer to the printer file for a description of the error encountered and determine if the sample tables were reloaded.**

Explanation: An error occurred while reloading the SUPPLIER/PARTS sample tables.

System Action: The SQLDBMIG EXEC continues, but the sample tables may not be available when you access the database. A printer file is being generated with a description of the error encountered.

User Response: Check the printer file after the SQLDBIMG EXEC is finished. Check the installation EXEC to find out how to reload the SUPPLIER/PARTS sample tables.

ARI6702I **SUPPLIER/PARTS sample tables reloaded.**

Explanation: The SUPPLIER/PARTS sample tables have been successfully reloaded.

System Action: The SQLDBMIG EXEC continues.

ARI0700I **SQLINIT on *&1* successful.**

Explanation: SQLINIT successfully completed for the database named in &1.

ARI0701E **SQLINIT on *&1* failed.**

Explanation: SQLPREP requested an SQLINIT to a different database. The SQLINIT failed; therefore, SQLPREP could not preprocess the program on the named database.

The database name was specified in either the DBNAME parameter or was in the list of databases referenced by the DBFILE parameter.

System Action: SQLPREP bypasses this database and continues with the next database.

Programmer Response: A previous message from SQLPREP may indicate the nature of the error. If the database name is invalid, correct the database name and re-execute the SQLINIT.

If the database is not available, contact your support organization.

ARI0702I **Create/recreate the required DB2 Server for VM default bootstrap files.**

Explanation: This is the expected response from the database manager.

ARI0703I **Preprocessing on *dbname* successful.**

Explanation: This is the expected response from the database manager.

ARI0704I **Warning: Preprocessing on *dbname* successful though warnings occurred.**

Explanation: Processing for this database returned a return code of 4 indicating that some warning occurred. Inspect the LISTPREP file produced to verify that the warnings do not require additional action.

ARI0705E **Preprocessing on *dbname* failed.**

Explanation: An error occurred during preprocessing that resulted in a nonzero return code (but less than 8).

System Action: SQLPREP bypasses this database and continues with the next database.

Programmer Response: Examine the LISTPREP file generated by the preprocessor to identify the nature of the error.

ARI0706E **Preprocessing on *dbname* failed because of a severe error.**

Explanation: A severe error occurred during preprocessing that resulted in a nonzero return code (higher than 8).

System Action: SQLPREP bypasses this database and continues with the next database.

Programmer Response: Examine the LISTPREP file generated by the preprocessor to identify the nature of the error.

ARI0707E **SQLPREP did not attempt preprocessing on *&1*.**

Explanation: Preprocessing was not attempted for this database because of an error attempting to connect to the database. This message should be preceded by an ARI0701E message.

System Action: SQLPREP bypasses this database and continues with the next database.

Programmer Response: A previous message from SQLPREP may indicate the nature of the error. If the database name is invalid, correct the database name and re-execute the EXEC.

If the database is not available, contact your support organization.

ARI0708I **All *exec-name* EXEC processing completed successfully.**

Explanation: This is the expected response from the database manager.

ARI0709I **Warning: Warnings occurred during *exec-name* EXEC processing.**

Explanation: This message indicates that the highest level of error from processing this EXEC was WARNING.

Programmer Response: Inspect the LISTPREP file produced to verify that the warning does not require additional action.

ARI0710E **Errors occurred during *&1* EXEC processing.**

Explanation: This message indicates that at least one error has occurred during execution.

If this message was issued at the end of the SQLPREP EXEC, either a previous message has been issued or the LISTPREP listing contains an error message. If this message was not issued at the end of the SQLPREP EXEC, refer to the previous message issued.

Programmer Response: Take appropriate action based on the previous messages.

ARI0711I ***exec-name* called with the following parameters: *input parameters*.**

Explanation: This message shows the *input parameters* that you provided when you invoked an EXEC *exec-name*. For example, this message can list the options you used for SQLSTART when preprocessing in single user mode.

ARI0712I **Now invoking DB2 Server for VM to change the password for the user ID SQLDBA to SQLDBAPW.**

Explanation: This is the expected response from the database manager.

ARI0713I **Preprocessor *prepname* called with the following parameters:**

Explanation: This message lists the SQLPREP options in effect in this invocation.

ARI0714I **DB2 Server for VM Version *&1* Release *&2* Modification *&3*.**

Explanation: This is the expected response from the database manager.

ARI0715E **Invalid parameter value (*&1*) specified.**

Explanation: The only valid user parameter for SQLBOOTS is REMOTE. All other parameters are valid only when invoked from other system EXECs.

System Programmer Response: If you were re-creating the bootstrap module, use REMOTE or no parameter.

If this was an internal invocation of SQLBOOTS, contact the designated support group for your installation.

ARI0716E **An error occurred while creating the default DB2 Server for VM bootstrap file *filename filetype* on the production disk.**

Explanation: A nonzero return code occurred during SQLBOOTS processing. Message ARI0718I follows this message and indicates the function that failed.

System Programmer Response: If the command was LOAD or START, ensure you are accessing the production disk.

If the command was COPYFILE, ensure you are accessing the production disk or take the appropriate action for the return code given by COPYFILE.

If the command was GENMODE, take the appropriate action for the return code given by COPYFILE.

ARI0717I **Start *&1* EXEC: *&2 &3 &4*.**

Explanation: This is the expected response from the database manager.

&1 is the EXEC name.

&2 is the date.

&3 is the time.

&4 is the time zone.

ARI0718I ***&1* command Return Code = *&2*.**

Explanation: This is the expected response from the database manager.

ARI0719I **Note: Ensure that the DB2 Server for VM production minidisk is accessed in write mode.**

Explanation: This is the expected response from the database manager.

ARI0720I **Default DB2 Server for VM bootstrap file SQLDBA *&1* created on the production disk.**

Explanation: This is the expected response from the database manager.

ARI0721I Get DB2 Server for VM *&1* minidisk *&2* access: *&3* *&4*.

Explanation: The production disk will be relinked to provide the required access mode.

&1. is the minidisk accessed (service or production).

&2. is the access mode (read or write).

&3. is the machine id of the minidisk accessed.

&4. is the cuu of the minidisk accessed.

ARI0722E Unable to acquire write access to mode M or W. User=*&1* Device=*&2* Return Code=*&3*.

Explanation: During a relink procedure (as mentioned in message ARI0721I), a write mode could not be obtained.

System Action: The function is terminated.

System Programmer Response: Verify that this user ID has write access to the production minidisk. The error could be caused by another user having gained write access before this user ID. In that case, detach the production minidisk from the other user ID and re-execute the function.

ARI0723E Unable to access *&1* *&2*.

Explanation: Access to a required disk failed.

System Action: The function is terminated.

System Programmer Response: Verify that this user ID has write access to the production minidisk. The error could be caused by another user having gained write access before this user ID. In that case, detach the production minidisk from the other user ID and re-execute the function.

ARI0724I If the EXEC was run in a CMS batch machine, disregard preceding invalid function message.

Explanation: The ARISESCP EXEC was invoked to preserve some of the terminal settings. While the message is valid from a display terminal, it is not from CMSBATCH. The resulting INVALID FUNCTION messages can be ignored.

System Action: Processing continues.

User Response: Ignore the INVALID FUNCTION message if this message is the result of a CMSBATCH run.

ARI0725I FULLSCREEN *&1* has been suspended *&2*

Explanation: VM FULLSCREEN function was in effect and has been suspended for SQLSTART.

ARI0726I CP *&1* *&2* *&3* command was successful.

Explanation: This is the expected response from the database manager.

ARI0727I Load continuing. Text file count = *&1*.

Explanation: ARISAVES is proceeding with the required LOAD to perform the SAVESYS or SAVESEG function. This message is issued after 30 text files have been loaded.

System Action: Processing continues.

ARI0728I CMS LOAD command successful. Text file count = *&1*.

Explanation: ARISAVES has completed the loading of all the required text files.

ARI0729I Default storage key (STORKEY) = 13.

Explanation: No optional storage key was requested. The default storage key will be used.

ARI0730I No CMS HELP is available for the new default language.

Explanation: This is the expected response from the database manager.

ARI0731D Do you want to install the DB2 Server for VM HELP text? Enter 0(No), 1(Yes), or 111(Quit).

Explanation: This is the expected response from the database manager.

User Response: You must specify whether you want to install the DB2 Server for VM CMS HELP files and ISQL HELP Text. If the response is yes, both types of help will be installed. If the response is no, neither type of help will be installed. If the response is quit, processing is terminated. Enter the appropriate response.

ARI0732A Enter current password for application server user ID SQLDBA:

Explanation: To proceed with the requested function, the EXEC must first connect as SQLDBA. You must provide the SQLDBA password.

System Programmer Response: Enter the current password used to connect SQLDBA. This should be the DB2 Server for VM password used in a CONNECT

SQLDBA IDENTIFIED BY pw.

ARI0733I **Halting further execution on your request.**

Explanation: This is the expected response from the database manager.

ARI0734I **User ID SQLDBA password is restored.**

Explanation: This is the expected response from the database manager.

ARI0735I **Proceeding to restore original user ID SQLDBA password.**

Explanation: This is the expected response from the database manager.

ARI0736I **Load/reload of** *&1 &2 &3 &4* **completed.**

Explanation: This is the expected response from the database manager.

ARI0737I **Proceeding to load/reload** *&1 &2 &3 &4*

Explanation: This is the expected response from the database manager.

ARI0738I **Now invoking DB2 Server for VM to update the system catalog**

Explanation: This is the expected response from the database manager.

ARI0739I **Proceeding to create/recreate the** *&1 &2* **packages.**

Explanation: This is the expected response from the database manager.

ARI0740I *&1 &2* **package** *&3* **and** *&4* **created/recreated.**

Explanation: This is the expected response from the database manager.

ARI0741E **The application server did not reset SQLDBA's password to SQLDBAPW. The password you supplied (***&1***) may be incorrect.**

Explanation: To perform the requested function, you must set the password for SQLDBA to SQLDBAPW.

The attempt to change the password from the one you provided to SQLDBAPW failed.

System Action: The function is terminated.

System Programmer Response: Re-issue the requested function using the correct SQLDBA password.

ARI0742I **Password for user ID SQLDBA is reset to SQLDBAPW.**

Explanation: This is the expected response from the database manager.

ARI0743I **Reset user ID SQLDBA password to SQLDBAPW.**

Explanation: This is the expected response from the database manager.

ARI0744E **COPYFILE Return Code =** *&1***.**

Explanation: This is the expected response from the database manager.

ARI0745E **An error has occurred while copying** *&1 &2* **from the DB2 Server for VM service minidisk to** *&1 &3* **on the specified minidisk. Refer to the DB2 Server for VM NLS program directory for this language for a listing of all files that should reside on the user specified minidisk for the DB2 Server for VM HELP text.**

Explanation: An error occurred while you were copying the HELP text to the selected minidisk.

Message ARI0744E gives the COPYFILE return code.

System Programmer Response: Take appropriate action to resolve the error based on the COPYFILE return code. For an explanation of the COPYFILE return code, you can use the HELP COPYFILE command.

ARI0746I **Proceeding to copy CMS HELP text files from the DB2 Server for VM service minidisk to the specified minidisk.**

Explanation: ARISDBMC is starting to copy the HELP files to the specified disk.

System Action: Processing continues.

System Programmer Response: If message ARI0747I follows this message, no action is required.

ARI0747I **The copying of the CMS HELP text files was successful.**

Explanation: The copying of the HELP files started when message ARI0746 was terminated successfully.

ARI0748E **The language repositories are not on the service disk. Processing terminates.**

Explanation: The language repositories for the language being installed were not found on the service disk.

System Action: The function is terminated.

System Programmer Response: Reload the language repositories from the NLS tape for the language being installed and rerun the ARISDBMA EXEC.

ARI0749I **ISQL and FLAGGER support completed in the server.**

Explanation: The function is started when message ARI0750I completes successfully.

ARI0750I **Proceeding to complete ISQL and FLAGGER support.**

Explanation: Support for ISQL and the FLAGGER will now be added to this database. The required packages will be reloaded.

ARI0752I *&1* **to create** *&2 &3 &4 &5* **package.**

Explanation: This is the expected response from the database manager.

ARI0753I *&1* **package** *&2* **created in database.**

Explanation: This is the expected response from the database manager.

ARI0754I **Database setup** *&1 &2 &3* **completed.**

Explanation: This is the expected response from the database manager.

ARI0755I **Proceeding to set up the database** *&1 &2 &3 &4*

Explanation: This is the expected response from the database manager.

ARI0756I **DB2 Server for VM database generation (DBGEN) complete.**

Explanation: This is the expected response from the database manager.

ARI0757I **Table SQLDBA.SYSLANGUAGE created/recreated.**

Explanation: This is the expected response from the database manager.

ARI0758D **The starter database specifications and the virtual minidisk addresses 200, 201, and 202 will be used for the database. All existing data on these minidisks will be destroyed. Enter USE to use these minidisks or enter QUIT to stop.**

Explanation: If you set up the database, the starter database specifications and addresses specified will be used, and all the existing data on these minidisks will be destroyed.

System Programmer Response: If you want to use the default generation parameters, enter USE. Otherwise enter QUIT, and perform the database generation separately.

ARI0759I **CMS LOADLIB LIST output for:** *&1 &2 &3***.**

Explanation: This is the expected response from the database manager.

ARI0760A **Obtain write access to the** *&1* **disk and rerun this EXEC.**

Explanation: The minidisk mentioned in the message must be accessed in write mode.

System Programmer Response: Obtain write access to the minidisk before re-executing this EXEC.

ARI0761I **Now invoking DB2 Server for VM to migrate the system catalog to** *&1***.**

Explanation: This is the expected response from the database manager.

ARI0762I **DB2 Server for VM default bootstrap files created.**

Explanation: This is the expected response from the database manager.

ARI0763A *&1* **195 minidisk updated successfully; reaccess the minidisk in read mode.**

Explanation: The 195 minidisk was accessed in write mode for update purposes.

System Programmer Response: Re-access the 195 production minidisk in read mode.

ARI0764I **DB2 Server for VM default bootstrap files created successfully.**

Explanation: This is the expected response from the database manager.

ARI0765I **The SQLBOOTS EXEC is being invoked to create/recreate the DB2 Server for VM default bootstrap files.**

Explanation: This is the expected response from the database manager.

ARI0766I **ARISQLLD LOADLIB Q was created successfully.**

Explanation: This is the expected response from the database manager.

ARI0768I **The following file was created or replaced:** *&1 &2* **Q.**

Explanation: This is the expected response from the database manager.

ARI0769E **DB2 Server for VM service minidisk not accessed as V disk.**

Explanation: The production disk must be accessed as the V disk to execute ARISPDFC. The disk location is verified by locating the file ARISSVCC MACRO V.

System Action: Processing is terminated.

System Programmer Response: Access the production minidisk (193) as the V disk, and re-execute ARISPDFC.

ARI0770E **** &1* **195 minidisk was not updated successfully.**

Explanation: An error occurred in ARISPDFC. Messages ARI0771E precedes this message.

System Programmer Response: Identify the reason the COPYFILE failed, and correct the error.

ARI0771E **An error has occurred while copying file** *&1 &2* **from the service minidisk to file** *&3 &4* **on the production minidisk.**

Explanation: An error occurred while copying the specified file.

System Programmer Response: Identify the reason the COPYFILE failed, and correct the error.

ARI0772I **Erase and recreate the DB2 Server for VM load library:** *&1 &2 &3***.**

Explanation: The new load library was created using a temporary name. The actual library will now be erased and re-created from the temporary library. &1, &2, &3 are file name, file type, and file mode.

ARI0773I **Now invoking DB2 Server for VM to recreate the DBS Utility package &1.**

Explanation: This is the expected response from the database manager.

ARI0779A **Enter the current password for application server user ID SQLDBA (or enter QUIT to exit).**

Explanation: To proceed with the requested function, the EXEC must first connect as SQLDBA. You must provide the SQLDBA password.

System Programmer Response: To continue, enter the current password used to connect SQLDBA. This should be the DB2 Server for VM password used in a CONNECT SQLDBA IDENTIFIED BY pw. To stop processing, enter QUIT.

ARI0780I **Formatting minidisk** *&1***.**

Explanation: This is the expected response from the database manager.

ARI0781I **Note: Any one of Return Codes 0 through 4 is acceptable.**

Explanation: This is the expected response from the database manager.

ARI0782I **Password for the user ID SQLDBA is now {reset | SQLDBAPW}**

Explanation: This is the expected response from the database manager.

ARI0783I **Now invoking DB2 Server for VM to reset password for user SQLDBA.**

Explanation: This is the expected response from the database manager.

ARI0784I **DB2 Server for VM HELP text tables created.**

Explanation: This is the expected response from the database manager.

ARI0785I **Now invoking DB2 Server for VM to reload the HELP text tables** *&1* **and** *&2***. -or- Now invoking DB2 Server for VM to reload ISQL routine ARINEWUS.**

Explanation: This is the expected response from the database manager.

ARI0786A **This process assumes that the HELP text tables presently reside in a DBSPACE named PUBLIC.HELPTEXT. If this is correct, press Enter. If the tables have been moved, enter the name of the new DBSPACE (owner.dbspacename) and press Enter.**

Explanation: This is the expected response from the database manager.

System Programmer Response: Identify the DBSPACE containing the help text tables.

ARI0787I Now migrating HELP text tables.

Explanation: This is the expected response from the database manager.

ARI0788I Now invoking DB2 Server for VM to create/recreate the *&1 &2* package *&3*.

Explanation: This is the expected response from the database manager.

ARI0789I Warning: An error occurred while the database manager was reloading the DB2 Server for VM sample tables. The sample tables may not be available while you are accessing this database. After the SQLDBMIG EXEC finishes, refer to the printer file for a description of the error encountered and determine if the sample tables were reloaded.

Explanation: The sample tables may not be usable.

System Action: Processing continues.

System Programmer Response: You may want to notify your users to delay using the sample tables until the problem has been corrected.

Use the printer file to identify how to correct the problem.

ARI0790I DB2 Server for VM sample tables reloaded.

Explanation: This is the expected response from the database manager.

ARI0791I Now invoking system to reload the DB2 Server for VM sample tables.

Explanation: This is the expected response from the database manager.

ARI0792I Warning: An error occurred while reloading the ISQL routine ARINEWUS. The new routine may not be available. After the SQLDBMIG EXEC finishes, refer to the printer file for a description of the error encountered and determine if the routine was reloaded.

Explanation: This is the expected response from the database manager.

ARI0793I ISQL routines loaded/reloaded.

Explanation: This is the expected response from the database manager.

ARI0794I DB2 Server for VM trace formatter execution begins.

Explanation: This is the expected response from the database manager.

ARI0795I The DB2 Server for VM catalog was updated.

Explanation: The updates to the catalog started with message ARI0738I completed successfully.

ARI0796I End *&1* EXEC: *&2 &3 &4*.

Explanation: This is the expected response from the database manager.

ARI0797I Incorrect invocation. Correct syntax is: ARISDBMC langkey serv_fm prod_fm action where:
langkey=the language key.
serv_fm=the mode of the service minidisk.
prod_fm=the mode of the production minidisk.
action=the action to perform where: C=Copy, D=Delete, I=Installation.

Explanation: The parameters passed to the ARISDBMC EXEC are invalid.

System Programmer Response: This message has resulted from an internal error in the EXEC that invoked ARISDBMC EXEC. Contact the designated support group for your installation.

ARI0798I DB2 Server for VM corrective service processing successful.

Explanation: This is the expected response from the database manager.

ARI0799I Begin checking link book contents for possible service.

Explanation: This is the expected response from the database manager.

Chapter 13. ARI7001–ARI7970 ISQL Messages

ARI7001E ISQL is not supported on this terminal.

Explanation: You tried to run ISQL on a terminal which is not supported. ISQL supports IBM 3270 terminals or 3270-compatible display devices having a 24 X 80 or larger mode (a line size of at least 80 characters and 24 lines on the screen).

System Action: ISQL ends.

User Response: Run ISQL only on terminals which are supported. Ensure that you are using a supported terminal type for your operation.

ARI7020E System problem. The message normally displayed giving the result of the SQL command just entered cannot be printed. The SQL statement ended with an SQLCODE of *code*. Report this message to your system programmer.

Explanation: ISQL tried to display message ARI0500I, ARI0501I, ARI0502I, ARI0503E, ARI0504I, or ARI0505I, but the system had problems getting it.

System Action: ISQL printed this message instead of the one it normally would. The system waits for you to enter the next statement.

User Response: Look up the text for the SQLCODE indicated in the message. Notify your system programmer.

System Programmer Response: There is a problem with the DB2 Server for VM language message text module ARIMS001 or equivalent. Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7021E The application server has issued a ROLLBACK statement. All work entered for processing since the last COMMIT statement was rolled back. You may have to reenter some statements.

Explanation: During DB2 Server for VM error processing, it was discovered that the database manager issued a ROLLBACK command for the user. The reason is stated in the DB2 Server for VM message immediately preceding this one. If SET AUTOCOMMIT was:

ON Any work entered for this command was rolled back.

OFF Any work entered for the database manager since the beginning of the logical unit of work was rolled back.

System Action: The system issued a ROLLBACK command. The system waits for you to enter the next statement.

User Response: Read the SQL error message for the message preceding this one, its system and user action. If the problem is still not clear, notify your system programmer. If you want to reenter the work that was rolled back when SET AUTOCOMMIT was:

ON Reenter the last command you had entered.

OFF Reenter all commands entered since you began your logical unit of work.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7022E System message error. Notify your system programmer.

Explanation: An internal error occurred while ISQL was trying to display a message.

System Action: The system waits for you to enter the next statement.

User Response: Notify your system programmer. You can continue your ISQL session.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7039D Do you want to end this ISQL sign-on attempt? Enter 0(No) or 1(Yes).

Explanation: This message allows you to exit from the sign-on procedure when all agents are busy.

System Action: If you enter YES, you will return to CMS. If you enter NO, you will wait for an available agent.

User Response: Enter YES or NO.

ARI7040I CANCEL processing complete.

Explanation: CANCEL processing has completed.

System Action: The system waits for you to enter the next statement.

© Copyright IBM Corp. 1987, 2000

ARI7041D AUTOCOMMIT is OFF. Do you still want to CANCEL? Enter 0(No) or 1(Yes).

Explanation: You entered CANCEL while autocommit is off. If the cancel process is done, not only will the command running now be cancelled, but all the work you did since you set autocommit off or since the last COMMIT WORK will be rolled back. You are asked if the cancel should still be done.

System Action: The system waits for your reply.

User Response: Enter YES if the cancel should still be done. Enter NO if the cancel should not be done.

ARI7042E Routine ended. It was cancelled because of your request or an error detected by ISQL.

Explanation: A routine was executing at the time a CANCEL command was entered. The command could have been entered from the terminal or it could have been read from a routine. The CANCEL command could also have been entered by ISQL if an error was detected while processing a routine. Additional messages will be displayed when ISQL finds an error condition.

System Action: A ROLLBACK WORK RELEASE is performed. The routine is ended immediately. The system waits for you to enter the next statement.

User Response: If additional error messages are displayed, correct the error and rerun the routine.

ARI7043I CANCEL processing caused a ROLLBACK RELEASE. The ROLLBACK RELEASE was successful.

Explanation: The specified ROLLBACK operation was performed because a CANCEL command had been entered previously. The command could have been entered from the terminal or it could have been read from a routine. The CANCEL command could also have been entered by ISQL if an error was detected while processing a routine. The ROLLBACK processing was successful.

The ROLLBACK RELEASE statement causes the database manager to treat any later SQL requests in this terminal session as coming from the default user ID for the virtual machine in which ISQL is running. Most ISQL users are not affected by this action. However, if you had issued an explicit CONNECT statement so that the database manager treated your SQL requests as coming from a user ID *other than* the user ID of the virtual machine you are using, and you want to enter more SQL requests in this terminal session, you must reenter that CONNECT statement to fully recover from the ROLLBACK RELEASE action. The SQL user ID and target database name will be reset to their defaults after the ROLLBACK RELEASE action.

System Action: The system waits for you to enter the next statement.

ARI7045E CANCEL processing caused a ROLLBACK RELEASE. The ROLLBACK RELEASE was unsuccessful. See previous messages for more details.

Explanation: The specified ROLLBACK operation was performed because a CANCEL command had been entered previously. The ROLLBACK processing was not successful. Additional messages are displayed to indicate why the ROLLBACK processing failed.

The ROLLBACK RELEASE statement causes the database manager to treat any later SQL requests in this terminal session as coming from the default user ID for the virtual machine in which ISQL is running. Most ISQL users are not affected by this action. However, if you had issued an explicit CONNECT statement so that the database manager treated your SQL requests as coming from a user ID *other than* the user ID of the virtual machine you are using, and you want to enter more SQL requests in this terminal session, you must reenter that CONNECT statement to fully recover from the ROLLBACK RELEASE action. The SQL user ID and target database name will be reset to their defaults after the ROLLBACK RELEASE action.

System Action: ROLLBACK processing is always performed when a CANCEL command is entered. Additional messages are displayed to indicate the reason for the failure.

User Response: See the other messages for the reason for the failure.

ARI7047I CANCEL request accepted.

Explanation: You entered a CANCEL request with AUTOCOMMIT on, or you entered YES when asked if a CANCEL request should be done. This message indicates that the request has been accepted and the system will do the necessary cancel processing.

System Action: The system continues to process the cancel request.

ARI7048I CANCEL request ignored.

Explanation: You entered NO when asked if a CANCEL request should be done. This message indicates that the request will be ignored.

System Action: The system waits for you to enter the next statement.

ARI7049E ISQL could not complete CANCEL initialization. This session ended. Report this message and the previous SQL message to your system programmer.

Explanation: An SQL error occurred while ISQL was trying to enable CANCEL processing during ISQL startup.

System Action: ISQL startup stops.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7050I CANCEL already in progress.

Explanation: You entered the CANCEL command while your command was already being cancelled.

System Action: This CANCEL command is ignored. Your original CANCEL command processing continues.

ARI7051E Routine or EXEC processing ended. It was cancelled as you requested or because ISQL detected an error.

Explanation: A routine or EXEC was executing when a CANCEL command was entered. The command could have been entered by the user from the terminal or it could have been read from a routine or EXEC. The CANCEL command could also have been entered by ISQL if an error was detected while processing a routine or EXEC. Additional messages will be displayed when ISQL finds an error condition.

System Action: A ROLLBACK WORK RELEASE is performed. The routine or EXEC is ended immediately. The system waits for you to enter the next statement.

The ROLLBACK RELEASE statement causes the database manager to treat any later SQL requests in this terminal session as coming from the default user ID for the virtual machine in which ISQL is running. Most ISQL users are not affected by this action. However, if you had issued an explicit CONNECT statement so that the database manager treated your SQL requests as coming from a user ID *other than* the user ID of the virtual machine you are using, and you want to enter more SQL requests in this terminal session, you must reenter that CONNECT statement to fully recover from the ROLLBACK RELEASE action. The SQL user ID and target database name will be reset to their defaults after the ROLLBACK RELEASE action.

User Response: If additional error messages are displayed, correct the error and rerun the the routine or EXEC.

ARI7060I Maximum input length of 2048 characters has been reached. Remainder of input is lost.

Explanation: You entered too many continuation lines. Your input must not be longer than 2 048 characters. The first 2048 characters of your input have been accepted, but the rest is lost.

System Action: The system will try to process your command anyway. More messages will be displayed to let you know whether or not the command was processed successfully.

User Response: If the command you entered was an SQL command, you can shorten it with the CHANGE command so that it is not longer than 2 048 characters. Then enter the START command to process it again.

ARI7061I Previous input ignored.

Explanation: An IGNORE command was entered on a multi-line input.

System Action: All lines for this multi-line input are ignored. The system waits for you to enter the next statement.

ARI7062E Report the following error to your system programmer. Invalid call to ARIIRWI. Length=0.

Explanation: An ISQL module has called module ARIIRWI, but the length of the information to be displayed is given as 0.

System Action: The error message is displayed and module ARIIRWI returns to the caller. Processing continues.

User Response: Notify your system programmer. Give the system programmer a list of the five previous commands entered.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7063I Continuation character detected and ignored. Make sure your input is correct and without the continuation character.

Explanation: You entered a continuation character on a message reply. The continuation is ignored.

System Action: The system uses the reply without the continuation character as the full reply. If the reply is not valid, the original message will be issued again.

ARI7064I Routine processing cancelled. The error indicated by a previous message cancelled the processing of your routine, as you requested with the SET RUNMODE command.

Explanation: While a routine was running, an error was detected by ISQL. ISQL cancelled the routine processing because you requested that routines should be cancelled when errors are detected. See the SET RUNMODE command for additional information.

System Action: ISQL issues a ROLLBACK WORK command to undo all work done since the last COMMIT WORK. Routine processing ends. The system waits for you to enter the next statement.

User Response: Correct the cause of the error as indicated by previous error messages, and rerun the routine.

ARI7065I Routine processing stopped. The error indicated by a previous message has caused processing of your routine to be stopped, as you requested with the SET RUNMODE command.

Explanation: While a routine was running, an error was detected by ISQL. ISQL stopped the routine processing because you requested that routines should be stopped when errors are detected. See the SET RUNMODE command for additional information.

System Action: Routine processing ends. ROLLBACK WORK is not issued. The system waits for you to enter the next statement.

User Response: Correct the cause of the error as indicated by previous error messages, and rerun the routine.

ARI7066I Your input is being held.

Explanation: You entered a HOLD command or pressed the PF9 key. Your command is being held and may be continued.

System Action: The system waits for you to enter the rest of the command, unless the command was complete. If the command was complete, the HOLD command will be processed.

User Response: Press ENTER if the command is complete. If the command is not complete, you may enter the rest of the command, or IGNORE, or CANCEL.

ARI7068I Your input is continued. Enter more input or press Enter.

Explanation: You entered an input line with a continuation character or used PF9. Continue the input and press enter.

System Action: The system waits for you to enter additional input lines.

User Response: Enter additional input lines, IGNORE to have the input ignored, CANCEL to have the command cancelled, or press the ENTER key to have the command processed.

ARI7069I Your input has been held. Use the START command to process your input.

Explanation: You requested that an SQL statement be held by pressing the PF9 key or by entering the HOLD command. The SQL statement is held in the current command buffer. The SQL statement will be processed if you enter a START command or press the PF2 key.

System Action: The system waits for you to enter the next statement.

User Response: Enter another command: START - to start the held statement, CHANGE - to change the held statement, or another command.

ARI7070I Routine or EXEC processing cancelled. The error indicated by a previous message cancelled the processing of your routine or EXEC, as you requested with the SET RUNMODE command.

Explanation: While a routine or EXEC was running, an error was detected by ISQL. ISQL cancelled the routine or EXEC processing because you requested routines and EXECs be cancelled when errors are detected. See the SET RUNMODE command for additional information.

System Action: ISQL issues a ROLLBACK WORK RELEASE to undo all the work done since the last COMMIT WORK. The routine or EXEC processing will be terminated. The system waits for you to enter the next statement.

The ROLLBACK RELEASE statement causes the database manager to treat any later SQL requests in this terminal session as coming from the default user ID for the virtual machine in which ISQL is running. Most ISQL users are not affected by this action. However, if you had issued an explicit CONNECT statement so that the database manager treated your SQL requests as coming from a user ID *other than* the user ID of the virtual machine you are using, and you want to enter more SQL requests in this terminal session, you must reenter that CONNECT statement to fully recover from the ROLLBACK RELEASE action. The SQL user ID and target database name will be reset to their defaults after the ROLLBACK RELEASE action.

User Response: Correct the cause of the error as indicated by previous error messages, and rerun the routine or EXEC.

ARI7071I Routine or EXEC processing stopped. The error indicated by a previous message cancelled the processing of your routine or EXEC, as you requested with the SET RUNMODE command.

Explanation: While a routine or EXEC was running, an error was detected by ISQL. ISQL stopped the routine or EXEC processing because you requested routines and EXECs be stopped when errors are detected. See the SET RUNMODE command for additional information.

System Action: The routine or EXEC processing ends. ROLLBACK WORK is not issued. The system waits for you to enter the next statement.

User Response: Correct the cause of the error as indicated by previous error messages, and rerun the routine or EXEC.

ARI7079I ISQL initialization complete.

Explanation: ISQL initialization has completed.

System Action: System waits for you to enter an ISQL or SQL command.

User Response: Enter an ISQL or SQL command.

ARI7080A Please enter an ISQL command or an SQL statement.

Explanation: You must enter a command. A null line will not be accepted.

System Action: The system waits for you to enter a command.

User Response: Enter an ISQL or SQL command.

ARI7085E There is not enough system storage available. [This SIGNON ended.] Try again later.

Explanation: There is not enough system storage available to support your ISQL session. More storage must be obtained.

System Action: Your signon ends.

User Response: Free some storage and try again. If you still can not signon, notify your system programmer.

ARI7101E Internal processing error. Invalid call to ARIITIO.

Explanation: An internal processing error occurred in ISQL. Module ARIITIO was called to display data, but the length of the data was zero.

System Action: Control is returned to the module that called module ARIITIO. Processing continues.

ARI7102E Parameters are not supported on the CMS command.

Explanation: You have entered a CMS command with a parameter. The parameter is ignored. You are placed in the CMS SUBSET environment.

System Action: The system waits for you to enter a CMS SUBSET command (or RETURN to return to ISQL).

ARI7119E ISQL has abnormally terminated. Notify your system programmer.

Explanation: A severe error has occurred in ISQL. Your ISQL session ends immediately.

System Action: CMS abend processing and clean-up occurs.

User Response: Report the messages on your screen to your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7120E ISQL could not process your command because of a system problem. Your command is too large. Report this problem to your system programmer.

Explanation: One of the following conditions occurred:

- The length of the current SQL command was greater than 2048 characters.
- The length of the formatting information may be calculated as follows:

  ```
  FORMAT BLOCK SIZE = x + (n * y)
  ```

  where `x` is the size of the format block header (504 bytes), `n` is the number of columns in the select-list for the stored query (not to exceed 45 columns), and `y` is the size of the format information stored for each column (44 bytes).

System Action: No substitutions were performed. The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Determine which length was exceeded and why. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7121E **Command failed. ISQL could not process your SQL statement because it is too large. Substitutions would make the command line longer than 2048 characters.**

Explanation: When substitutions made for the placeholders (&1., &2., etc.), your command was longer than 2048 characters.

System Action: The system waits for you to enter the next statement.

User Response: Change the SQL command so that its total length, including substitutions, is less than 2048 characters.

ARI7123E **Command failed. You entered an invalid character after the &. Reenter the command with a number between 1 and 9999 following the &.**

Explanation: A nonnumeric character was found after &.

System Action: Parameter substitutions were performed until the invalid character was found. The system waits for you to enter the next statement.

User Response: Replace the invalid character after &, to a number between 1 and 9999 and reenter the command.

ARI7124E **Command failed. You entered a 0 after the &. Reenter the command with a number between 1 and 9999 following the &.**

Explanation: The first number after the & is a zero, which is not valid.

System Action: Parameter substitutions were performed until the zero was found.

User Response: Change the SQL command so that it no longer has a 0 after the &. The system waits for you to enter the next statement.

ARI7125E **Command failed. You entered a number greater than 9999 after the &. Reenter the command with a number between 1 and 9999 following the &.**

Explanation: The only valid numbers which can be substituted for a parameter are between 1 and 9999. A number greater than 9999 was found after an &.

System Action: Parameter substitutions are done until the invalid number was found. The system waits for you to enter the next statement.

User Response: Change invalid number to a number between 1 and 9999.

ARI7126E **Command failed. The command line was null after ISQL made substitutions. Change your command or routine and try again.**

Explanation: When substitutions made for the placeholders (&1., &2., etc.), your command was null.

System Action: No action is taken. The system waits for you to enter the next statement.

User Response: Change your command or routine and try again.

ARI7127I **Your command contains unpaired quotation marks.**

Explanation: The command you entered contained either an unpaired single quote (') or an unpaired double quote (").

System Action: The system finishes processing the command with the unpaired quotation mark. The system waits for you to enter the next statement.

User Response: Check the results of the command. If they are not what you wanted, reenter the command.

ARI7128E **Run command failed. A command line or data line in your routine is too large. Substitutions would make the line more than 254 characters long.**

Explanation: Substitutions made for the placeholders (&1., &2., etc.) would cause a line in your routine to exceed 254 characters. The line is displayed after the message.

System Action: Substitutions were performed until the line would have exceeded 254 characters with the next substitution. The system waits for you to enter the next statement.

User Response: Change the routine so that each line, including substitutions, has fewer than 254 characters.

ARI7180E **Command failed. The ISQLTRACE command has a missing or invalid parameter. Reenter the command specifying DUMP, DISPLAY, or N, where N is the desired number of entries in the trace table. N must be between 50 and 1000.**

Explanation: You entered an incorrect option on the ISQLTRACE command.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Reenter the command, specifying the option you want.

ARI7181I ISQL trace table display

First entry: address1
Current entry: address2
Last entry: address3

Explanation: You entered an ISQLTRACE DISPLAY command.

address1 is the address of the start of the trace table.

address2 is the address of the current entry in the table.

address3 is the address of the last entry in the table (not necessarily the last entry used in the table).

System Action: Trace table entries will be displayed following the heading. They will be displayed in reverse order, starting with the latest entry placed in the table.

ARI7182E Command failed. The number you specified on the ISQLTRACE command is not between 50 and 1000. Reenter the command specifying a number between 50 and 1000.

Explanation: You did not enter a number between 50 and 1000. When changing the number of entries in the trace table, only the numbers between 50 and 1000 are accepted.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Reenter the command with a number from 50 to 1000.

ARI7183E Command failed. The number you specified on the ISQLTRACE command has more than 4 digits. Reenter the command specifying a number between 50 and 1000.

Explanation: You entered a number that has more than four digits. When changing the number of entries in the trace table, only the numbers between 50 and 1000 are accepted.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Reenter the command with a number from 50 to 1000.

ARI7184I ISQLTRACE processing complete.

Explanation: The ISQLTRACE command processing you requested is complete.

System Action: The system waits for you to enter the next statement.

ARI7185I There is not enough storage available to change the number of trace entries in the trace table. The number of entries is unchanged.

Explanation: There was not enough storage available to obtain a new trace table. The size of the trace table remains unchanged.

System Action: The system waits for you to enter the next statement.

User Response: Retry the command later.

ARI7186I The trace table can now hold *number* entries.

Explanation: The size of the trace table has been changed to hold the number of entries you specified on the ISQLTRACE command.

number is the number of entries you specified.

System Action: The system waits for you to enter the next statement.

ARI7300E Command ended. The input command could not be processed because of system problems. You may enter another command.

Explanation: INPUT could not get the storage it needed for processing. INPUT processing ends.

System Action: The INPUT command ended. The system waits for you to enter the next statement.

User Response: You are free to enter another command or INPUT can be tried again. If you enter the INPUT command again and continue to receive this message, notify your system programmer.

System Programmer Response: If this condition persists, increase the size of the user's virtual machine.

ARI7301E Command failed. Specify the table name on the input command.

Explanation: No table name was specified after INPUT.

System Action: The INPUT command has ended. The system waits for you to enter the next statement. No SQL COMMIT WORK or ROLLBACK WORK has been issued.

User Response: Reenter the INPUT command specifying table name and optionally (column names).

ARI7302E Command failed. Table name is incorrect. You must use double quotation marks in pairs.

Explanation: During the checking of the table name a double quote was encountered. However, another

double quote was not found. They must be specified in pairs.

System Action: The INPUT command has ended. The system waits for you to enter the next statement. No SQL COMMIT WORK or ROLLBACK WORK has been issued.

User Response: Reenter the INPUT command making sure that the double quotes around the table name are specified in pairs.

ARI7303E Command failed. Table name is incorrect. You must specify the table name after the creator name.

Explanation: During checking of the {owner.}table name, no table name was found after owner.

System Action: The INPUT command has ended. The system waits for you to enter the next statement. No SQL COMMIT WORK or ROLLBACK WORK has been issued.

User Response: Reenter the INPUT command specifying table name after owner.

ARI7304E Command failed. Column names are specified incorrectly. A left parenthesis must come before column names.

Explanation: During checking of the INPUT command, it was discovered that the column names did not start with a left parenthesis.

System Action: The INPUT command has ended. The system waits for you to enter the next statement. No SQL COMMIT WORK or ROLLBACK WORK has been issued.

User Response: Reenter the INPUT command. Make sure that any column names specified begin with a left parenthesis “(” and end with a right parenthesis ")" as required.

ARI7305E Command failed. Column names are specified incorrectly. A right parenthesis must follow column names.

Explanation: During checking of the INPUT command it was discovered that the column names did not end with a right parenthesis.

System Action: The INPUT command has ended. The system waits for you to enter the next statement. No SQL COMMIT WORK or ROLLBACK WORK has been issued.

User Response: Reenter the INPUT command. Make sure that any column names specified begin with a left parenthesis "(" and end with a right parenthesis ")" as required.

ARI7306E The input command is ending. Read the previous SQL error message and its action for the reason.

Explanation: INPUT cannot continue processing for reasons described by the SQL error message printed prior to this one.

System Action: INPUT is ending. A ROLLBACK WORK will be issued for a user-entered or system-entered CANCEL, or if SET AUTOCOMMIT is ON. If SET AUTOCOMMIT is OFF and INPUT is not being ended because of a CANCEL, a ROLLBACK WORK will not be performed. INPUT ends. The system waits for you to enter the next statement.

User Response: Read the SQL error message and its action. If the problem is still unclear, notify your system programmer. Otherwise, take action to correct the problem and enter the INPUT command again.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7307A Enter data values, separated by commas, for the following columns. Enclose character data in single quotation marks.

Explanation: Enter data for the columns specified after this message. The character data must be enclosed in single quotes.

System Action: Immediately following this message the system writes the names of the columns for which you are to enter data, the data type, and whether null values are allowed in this field. The keyboard is then unlocked and you are requested to enter data or an INPUT command.

User Response: You should note this carefully. You will be asked to enter data soon. Data must be entered in the order specified and must be separated by commas.

ARI7308A Data not accepted. Reenter data row.

Explanation: INPUT encountered a problem as described by the SQL error message printed prior to this one. The data could not be processed.

System Action: The data just entered was not accepted. No other data has been lost. You are asked to enter data or another INPUT command.

User Response: Read the SQL error message and its action. If the problem is still unclear, notify your system programmer. Otherwise, if you want this data entered into the table, correct the problem and enter it again.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the

designated support group for your installation.

ARI7309E **Command failed. At least one space must separate table name and column names.**

Explanation: During checking of the INPUT command, a left parenthesis "(" was found as the next character after table name. There must be at least one blank between the table name and the start of the column names.

System Action: The INPUT command has ended. The system waits for you to enter the next statement. No SQL COMMIT WORK or ROLLBACK WORK has been issued.

User Response: Reenter the INPUT command. Put at least one blank between table name and the left parenthesis "(" which starts the specification of column names.

ARI7310E **Command ignored. Input cannot process this ISQL command. Reenter this command after you finish using the INPUT command.**

Explanation: The ISQL command entered was not valid at this time.

System Action: The command is ignored. You are asked to enter more data or an INPUT command.

User Response: If you want the command to be processed, bypass the INPUT command by typing END or CANCEL. Otherwise, enter more data or an INPUT command.

ARI7311E **Command ignored. You entered neither data nor a valid input command. Try again.**

Explanation: You entered neither data nor a valid INPUT command.

System Action: The command is ignored. The system waits for you to enter either data or an INPUT command.

User Response: Enter either data or an INPUT command.

ARI7312I **Since AUTOCOMMIT is OFF, SAVE was ignored. To commit data, enter an END command and then a COMMIT statement.**

Explanation: Normally, INPUT will commit the data you entered into the table when you enter SAVE. Because you entered SET AUTOCOMMIT OFF before you entered INPUT, you will have to commit data into the table yourself with the COMMIT statement.

System Action: No action is taken. The system asks you to enter more data or an INPUT command.

User Response: If you want to commit your data now, leave INPUT by entering the END command. To commit your data, enter a COMMIT statement. Otherwise, enter either more data or an INPUT command.

ARI7313I *Command* **processing continuing. Excess parameters on** command **are ignored.**

Explanation: During checking of the command (where command is SAVE, BACKOUT, or END) a parameter was discovered. Parameters are not valid.

System Action: The parameter is ignored. Processing of the command continues.

ARI7314I **SAVE processing is complete.**

Explanation: The data which you had previously entered was committed to a DB2 Server for VM table.

System Action: The system waits for you to enter data or another INPUT command.

ARI7315E **SAVE command failed. No data was committed to the table. The INPUT command has ended. Notify your system programmer.**

Explanation: ISQL tried to save your input data by issuing a COMMIT WORK command. An error occurred as described in the SQL error message above.

System Action: The system will issue a ROLLBACK WORK. INPUT processing has ended.

User Response: Read the SQL error message and its action. Notify your system programmer. You are free to enter another ISQL command. You will not be able to enter another SQL command during this session.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7316E **Since AUTOCOMMIT is OFF, BACKOUT was ignored. To ROLLBACK data, enter an END command and then a ROLLBACK statement.**

Explanation: Normally, INPUT will roll back the data you entered into the table when you enter BACKOUT. Because you entered SET AUTOCOMMIT OFF before you entered INPUT, you will have to roll back data yourself with the ROLLBACK statement.

System Action: No action is taken. The system asks you to enter more data or an INPUT command.

User Response: If you want to roll back your data now, leave INPUT mode by entering the END

command. To roll back your data, enter a ROLLBACK statement. Otherwise, enter either more data or an INPUT command.

ARI7317I BACKOUT processing is complete.

Explanation: The data which you had previously entered was rolled back from the table.

System Action: The system waits for you to enter the next statement.

ARI7318E BACKOUT command failed. The INPUT command has ended. Notify your system programmer.

Explanation: ISQL tried to backout the data by issuing an SQL ROLLBACK WORK command. This command encountered an error specified by the SQL error message above.

System Action: The database manager will roll back your input during error processing. INPUT processing has ended. The system waits for you to enter the next statement.

User Response: Read the SQL error message and its action and notify your system programmer. You are free to enter another ISQL command. You cannot enter another SQL command during this session.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7319I Since AUTOCOMMIT is OFF, END does not commit data into the table. Enter a COMMIT statement to save your data.

Explanation: Normally, INPUT will commit the data you entered into the table when you enter END. Because you entered SET AUTOCOMMIT OFF before you entered INPUT, you will have to commit data into the table yourself with the COMMIT statement.

System Action: No action is taken. The INPUT command has ended. The system waits for you to enter the next statement.

User Response: If you want to commit data now, enter a COMMIT statement. Otherwise, you are free to enter another command.

ARI7320I END committed data to the table.

Explanation: The data which you had previously entered was committed to the DB2 Server for VM table.

System Action: INPUT is ending. The system waits for you to enter the next statement.

ARI7321E An error occurred during END command processing. Read the previous SQL error message and its action. The database manager issued a ROLLBACK statement.

Explanation: ISQL tried to save the input data by issuing a COMMIT statement. This statement encountered an error specified by the SQL error message above.

System Action: The system issued a ROLLBACK statement. Input will end. The system waits for you to enter the next statement.

User Response: Read the SQL error message and its action. Notify your system programmer. You are free to enter another ISQL command. You cannot enter another SQL statement during this session.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7322I The INPUT command completed.

Explanation: INPUT processing has completed.

System Action: The system waits for you to enter the next statement.

ARI7323I You ended the INPUT command by entering a CANCEL command.

Explanation: INPUT processing has completed as a result of entering the CANCEL command.

System Action: The system waits for you to enter the next statement.

ARI7324E The INPUT command rolled back all your data.

Explanation: ISQL tried to insert data into the table you specified. An error occurred as described in the SQL error message above.

System Action: INPUT processing cannot continue. ISQL has issued a ROLLBACK WORK command. INPUT processing has ended. The system waits for you to enter the next statement.

User Response: Read the SQL error message and its action. If the problem is still unclear, notify your system programmer. Otherwise, you are free to enter another command.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7325I The system completed the INPUT command.

Explanation: INPUT processing completed as a result of system problems encountered during INPUT command processing.

System Action: The system waits for you to enter the next statement.

User Response: Read the SQL error message and its action. If the problem is still unclear, notify your system programmer. Otherwise, you are free to enter another command.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7326A No data was entered. Enter data or an INPUT command.

Explanation: You pressed the ENTER key with nothing or a line of blanks on the input line.

System Action: No action is taken. The system waits for you to enter data or another INPUT command.

User Response: Enter either data or an INPUT command.

ARI7327E The INPUT command failed. INPUT was entered from a routine, but was not complete. Insert an END command after the last DATA or INPUT command.

Explanation: Because INPUT was started from a routine, all data or INPUT commands must be entered from the routine also. INPUT is trying to process more data or an INPUT command, but there were no more records in the routine.

System Action: The system has issued a CANCEL command and a ROLLBACK WORK command. INPUT processing has ended. The system waits for you to enter the next statement.

User Response: Examine the routine just executed. For INPUT to finish properly, an END command should be inserted where you want the INPUT command to end. You are also free to enter another command.

ARI7328E Command failed. You specified the column names incorrectly. Column names must be specified between parentheses.

Explanation: During checking of the INPUT command, it was discovered that parentheses were coded after the table name. However, no column names were written between the parentheses.

System Action: The INPUT command has ended. The system waits for you to enter the next statement.

User Response: Resubmit the INPUT command. If you want all columns, type just the table name. If you want specific columns, type them between the parentheses.

ARI7329E Command failed. You specified the column names incorrectly. They must begin with a letter, $, #, or @ unless they are enclosed in double quotation marks.

Explanation: During checking of the INPUT command, it was discovered that one or more of the column names began with a character other than a letter, $, #, or @. This is not valid.

System Action: The INPUT command has ended. The system waits for you to enter the next statement.

User Response: Resubmit the INPUT command. If all column name or names in error begin with a character other than a letter, $, #, or @, you must enclose the name in double quotes ("). Otherwise, find the correct column name or names. The column name or names in error do not exist.

ARI7330E The INPUT command failed. A routine or an EXEC issued an incomplete INPUT command. Insert an END command after the last DATA or INPUT command.

Explanation: Since input was initiated from a routine or an EXEC, all data or INPUT commands must also be entered from the routine or EXEC. Input is trying to process more data or an INPUT command, but there were no more records in the routine or EXEC.

System Action: The system has issued a CANCEL command and a ROLLBACK WORK RELEASE command. INPUT processing has ended. The system waits for you to enter the next statement.

User Response: Examine the routine or EXEC just executed. For input to finish properly, an END command should be inserted where you want the INPUT command to end. You are also free to enter another command.

ARI7350E RUN processing cancelled. You have a RUN command in your routine. Remove the RUN command from the routine.

Explanation: RUN is not a valid command in a routine.

System Action: An internal cancel is issued. The system waits for you to enter the next statement.

User Response: Take the RUN command out of the routine.

ARI7351E **RUN command failed. You did not specify a routine name. Reenter the RUN command with a routine name.**

Explanation: You must specify the name of the routine to be executed on the RUN command. The format is: RUN userid.routine name parameters.

System Action: Command processing failed. The system waits for you to enter the next statement.

User Response: Reenter the RUN command with a routine name.

ARI7354E **RUN command failed. The RUN command cannot find the** userid**.ROUTINE table. Reenter the RUN command using a valid user ID or create the routine table.**

Explanation: The specified ROUTINE table does not exist.

System Action: The RUN command failed. The system waits for you to enter the next statement.

User Response: Reenter the RUN command specifying the proper user ID; or, if the ROUTINE table does not exist, it must be created using the CREATE TABLE command. See the *DB2 Server for VSE & VM Interactive SQL Guide and Reference* manual for information on creating and using routines.

ARI7355E **RUN command failed. The RUN command cannot find the** named **routine. Reenter the RUN command with a valid routine name.**

Explanation: The named routine could not be found.

System Action: The RUN command failed. The system waits for you to enter the next statement.

User Response: Reenter the RUN command using a valid routine name.

ARI7357E **RUN command failed. The RUN command cannot finish because of a system error. Notify your system programmer.**

Explanation: While the system was trying to process the routine an internal error occurred. The messages preceding this message describe the internal error.

System Action: The RUN command failed. The system waits for you to enter the next statement.

User Response: Save the associated error messages and notify your system programmer. Enter another command.

System Programmer Response: Use the SQL error code specified in the previously issued error message to determine the proper course of action. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7358E **RUN command failed. The** {userid | name} **must be 8 characters or less. Reenter the RUN command.**

Explanation: The user ID or name specified on the RUN command was larger than 8 characters. The maximum size of the user ID or name field is 8.

System Action: RUN processing failed. The system waits for you to enter the next statement.

User Response: Reenter the RUN command specifying a user ID or name of not more than 8 characters.

ARI7359E **RUN command failed. Rebuild the routine table with the SEQNO column defined as SMALLINT or INTEGER, and NOT NULL.**

Explanation: The ROUTINE table must be created with a SEQNO column defined with a data type of SMALLINT or INTEGER, and NOT NULL.

System Action: The system waits for you to enter the next statement.

User Response: Enter a DROP TABLE command for the ROUTINE table, and a CREATE TABLE command to rebuild it. Specify SMALLINT or INTEGER as the data type for the SEQNO column. See the *DB2 Server for VSE & VM Interactive SQL Guide and Reference* manual for information on creating and using routines.

ARI7360E **RUN command failed. You cannot use the** userid**.ROUTINE table again in this session. Enter another command.**

Explanation: The system was not successful in closing the cursor for this table.

System Action: RUN processing failed. The system waits for you to enter the next statement.

User Response: Save the associated error messages and notify your system programmer.

System Programmer Response: Use the SQL error code specified in the previously issued error message to determine the proper course of action. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7361I **RUN command is processing normally. However, an empty command line was encountered in the routine. The empty line is ignored.**

Explanation: While the system was processing the commands in the specified routine a null line was encountered.

System Action: The line is ignored and normal processing continues.

User Response: Remove the null line from the routine if you do not want to get this message.

ARI7362E **RUN command ended abnormally because of a parameter substitution error. Correct the error and reenter the RUN command.**

Explanation: During RUN command parameter substitution, an error occurred which caused substitution to end. The message which precedes this message states the cause of the error.

System Action: The RUN command failed. The system waits for you to enter the next statement.

User Response: Use the message which precedes this message to correct the parameter substitution error. Reenter the RUN command.

ARI7363E **RUN command failed. Rebuild the routine table with the command column defined as VARCHAR.**

Explanation: The ROUTINE table must be created with the COMMAND column as VARCHAR.

System Action: The system waits for you to enter the next statement.

User Response: Enter a DROP TABLE command to delete the ROUTINE table and a CREATE TABLE command to rebuild it. See the *DB2 Server for VSE & VM Interactive SQL Guide and Reference* manual for information on creating and using routines.

ARI7364E **RUN processing cancelled. You have a RUN command in your routine or EXEC. Remove the RUN command from the routine or EXEC.**

Explanation: RUN is not a valid command in a routine or an EXEC.

System Action: An internal CANCEL is issued. This action involves a ROLLBACK WORK RELEASE. The system waits for you to enter the next statement.

The ROLLBACK RELEASE statement causes the database manager to treat any later SQL requests in this terminal session as coming from the default user ID for the virtual machine in which ISQL is running. Most ISQL users are not affected by this action. However, if you had issued an explicit CONNECT statement so that the database manager treated your SQL requests as coming from a user ID *other than* the user ID of the virtual machine you are using, and you want to enter more SQL requests in this terminal session, you must reenter that CONNECT statement to fully recover from the ROLLBACK RELEASE action. The SQL user ID and target database name will be reset to their defaults after the ROLLBACK RELEASE action.

User Response: Take the RUN command out of the routine or EXEC.

ARI7398I **Profile processing is starting.**

Explanation: ISQL initialization found either a master, user, or signon PROFILE routine and is processing it.

System Action: The appropriate routines are run.

ARI7399I **The ISQL default profile values are in effect.**

Explanation: A PROFILE routine did not exist in the SQLDBA.ROUTINE table or in the userid.ROUTINE table. A routine was not specified as a parameter on the signon.

System Action: ISQL signon processing continues using system values.

User Response: If you want to know what values are in effect, enter the LIST SET command.

ARI7500E **Command failed. You can only issue a FORMAT command during a query result. Enter an SQL SELECT before issuing a FORMAT command.**

Explanation: You tried to format the screen when there was no SELECT command active.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Enter a SELECT command or start a SELECT command before you enter the FORMAT command.

ARI7501E **Command processing stopped. You entered an incorrect keyword** token **on the FORMAT command. Reenter the FORMAT command with one or more of the following keywords: ON, OFF, SUBTOTAL, TOTAL, GROUP, OUTLINE, SEPARATOR, VARCHAR, NULL, TTITLE, BTITLE, COLUMN, EXCLUDE, or INCLUDE.**

Explanation: You entered an incorrect keyword on the FORMAT command. One or more of the following keywords must be specified after the FORMAT

command: ON, OFF, SUBTOTAL, TOTAL, GROUP, OUTLINE, SEPARATOR, VARCHAR, NULL, TTITLE, BTITLE, COLUMN, EXCLUDE, or INCLUDE.

token is the incorrect keyword you entered.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Enter FORMAT with one or more of the correct keywords.

ARI7502I **The** keyword **indicator was previously** old setting**. It is now** new setting**.**

Explanation: This message shows the setting of the FORMAT keyword that you have just changed.

keyword is the keyword you entered on the FORMAT command.

old setting is the previous setting of the FORMAT keyword.

new setting is the new setting of the FORMAT keyword.

System Action: The system sets the indicator as specified by the command.

ARI7503I **ISQL erased the bottom title for this query.**

Explanation: The bottom title for this query has been erased.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7504E **Command processing stopped. You entered an incorrect option** token **for the FORMAT** keyword **command. Reenter FORMAT** keyword **command with ON or OFF.**

Explanation: You entered an incorrect option on the FORMAT command.

token is the incorrect option you entered.

keyword is the keyword you entered after the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the command with a valid option. The valid options are ON and OFF.

ARI7505E **Command processing stopped. The number after the** keyword **keyword cannot be more than 8 digits long. Try again.**

Explanation: A number you entered was too large. The field allows 8 digits, including leading zeros.

keyword is the keyword you entered.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Reenter the command with a number from 1 to 254.

ARI7506I **The width you specified for column** column id **is too large. WIDTH has been changed to 254 bytes.**

Explanation: You entered a width greater than 254. The system has set the width of the column to the maximum of 254.

column id is the column id you entered.

System Action: The command is processed as if the command FORMAT COLUMN column id WIDTH 254 had been entered.

User Response: If you do not want a width of 254, reenter the command with a width less than 254.

ARI7507E **Command processing stopped. Enter a number after the** keyword **keyword. Try again.**

Explanation: A nonnumeric value was specified for a numeric option.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the command using a number as the option.

ARI7508E **Command processing stopped. You must enter an option after the** keyword **keyword. Try again.**

Explanation: You did not enter any option for the keyword specified in the message. No default action is taken.

keyword is the keyword you entered.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the command, specifying the desired option.

ARI7509I **You entered a null string longer than 20 characters. Only the first 20 characters were used.**

Explanation: You entered a null string specification that is too long. The first 20 characters are accepted as the string; the rest are ignored.

System Action: The command is processed with first 20 characters.

User Response: If you do not want the shortened string as your null string, reenter the command with a new string. The new string must not be longer than 20 characters.

ARI7510I **You have not defined a bottom title.**

Explanation: No bottom title (BTITLE) is defined now.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: If you want a bottom title, use the FORMAT BTITLE command to define one.

ARI7511I keyword value **is greater than the page width. Information will be truncated if printed.**

Explanation: WARNING: Because the keyword value is greater than the page width, it will be truncated when printed.

keyword value is the keyword value you entered.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: To avoid truncating the data, specify a number smaller than the page size.

ARI7512I **You entered a top or bottom title longer than 100 characters. Only the first 100 characters will be used.**

Explanation: A top or bottom title can be at most 100 characters. The first 100 characters are accepted; the rest are ignored.

System Action: Title is set to the first 100 characters you entered.

User Response: If you do not want the shortened title, reenter the FORMAT TTITLE or FORMAT BTITLE command with a different title. The new title must not be longer than 100 characters.

ARI7513E **Command processing stopped. You entered an incorrect column ID** token **after the FORMAT** keyword **command. Reenter the command with a correct column ID. Only the first 45 columns can be formatted.**

Explanation: You entered an incorrect column id after the FORMAT command.

token is the column id you entered.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the FORMAT command with one or more column ids. Column id must be either a column name or a column number. If you enter more than one column id, they must be enclosed in parentheses.

ARI7514E **Command processing stopped. You can only enter a FORMAT SUBTOTAL command after you enter a FORMAT GROUP command.**

Explanation: To enter a subtotal command, a FORMAT GROUP command must have been previously entered. The command is ignored.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Enter FORMAT GROUP; then reenter the FORMAT SUBTOTAL command.

ARI7515E **Command processing stopped. Column** column id **is not a numeric column. Reenter the FORMAT** keyword **command using a numeric column.**

Explanation: You tried to enter a FORMAT DECIMAL or FORMAT ZEROS command on a nonnumeric column.

column id is the column id you entered.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the FORMAT command using a numeric column.

ARI7516I **You entered a column name longer than 30 characters on the FORMAT** keyword **command. Only the first 30 characters were used.**

Explanation: A column name can be at most 30 characters. The first 30 characters are accepted; the rest are ignored.

keyword is the keyword you entered on the FORMAT command.

System Action: The column name is set to or used as the first 30 characters you entered.

User Response: If you do not want the shortened name, reenter the FORMAT command with a new name. The new name must not be longer than 30 characters.

ARI7517E **Command processing stopped. You entered an incorrect column width for column** column id**. Reenter the FORMAT command with a column width from 1 to 254 characters.**

Explanation: The column width must be a number from 1 to 254.

column id is the column id you entered.

System Action: Command processing has been stopped at the column id indicated in the message. Any keywords before the keyword associated with this column id have been processed; all other keywords are ignored.

User Response: Reenter the FORMAT command with a column width from 1 to 254.

ARI7518E **Command processing stopped. The DPLACES value for column** column id **is too large. The number of decimal places must be less than the column width. Try again.**

Explanation: The width of a column must be at least one larger than the number of decimal places specified for that column.

column id is the column id you entered.

System Action: The command is ignored. Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: You can either increase the column width so that it is larger than the number of decimal places, or decrease the number of decimal places so that a smaller column size can be specified.

ARI7519E **Command processing stopped. Specify a number for the number of decimal places. Try again.**

Explanation: The FORMAT COLUMN column id DPLACES command must be followed by a number. The number of decimal places must also be less than the width of the column.

System Action: Command processing has been stopped at the DPLACES keyword in error. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the FORMAT COLUMN column id DPLACES command with a valid number.

ARI7520E **Command processing stopped. You entered an incorrect option** token **on the FORMAT COLUMN** column id **command. Reenter the command with one or more of the following options: WIDTH, ZEROS, DPLACES, or NAME.**

Explanation: You entered an incorrect option on the FORMAT COLUMN command.

token is the incorrect option you entered.

column id is the column id you entered.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the FORMAT command with one or more of the options listed in the message.

ARI7521I **All columns are included.**

Explanation: If the FORMAT INCLUDE command is entered without any column ids being specified, all columns are included.

System Action: All columns are included in the display.

User Response: None if this is the desired result. Otherwise, exclude the columns you do not want included.

ARI7522I **Because you did not specify any column IDs, all columns will be displayed.**

Explanation: You entered a FORMAT EXCLUDE ALL BUT or FORMAT INCLUDE ONLY command without any column ids. Therefore, all columns are included in the display.

System Action: All columns are included in the display. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: You can enter a FORMAT INCLUDE command for the columns you want included.

ARI7523E **Command processing stopped. Column ID** column id **is not one of the first 45 columns. ISQL can only format the first 45 columns.**

Explanation: You tried to format a column that is not one of the first 45 columns.

column id is the column id you entered.

System Action: Command processing has been stopped at the column id indicated in the message. Any keywords before the keyword associated with this column id have been processed; all other keywords are ignored.

User Response: Enter a new SELECT command that will make this column one of the first 45 columns. Then reenter the FORMAT command.

ARI7524I **Bottom title is** btitle**.**

Explanation: The bottom title is now the value *btitle*.

btitle is the current bottom title.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7525I **Top title is** ttitle**.**

Explanation: The top title is now the value ttitle.

ttitle is the current top title.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7526E **Command processing stopped. Zero is not a valid value for** keyword**. Reenter the command with a value greater than zero.**

Explanation: You entered 0. Zero is not an acceptable value for this keyword.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the FORMAT command with a value greater than zero.

ARI7527I **{Top | Bottom} title is too long to print on one line. When you print your report, part of the title may not appear.**

Explanation: The title will not fit on one line. The last part of the title may be cut off when your report is printed.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: None, if you don't mind the shortened title. Otherwise, enter a shorter title, using the FORMAT command.

ARI7528I **You did not enter the keyword BLANKS on your FORMAT SEPARATOR command. The number itself is used as your column separator.**

Explanation: The number you entered will be used as the column separator, not as the number of blanks between your columns.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: None, unless you want blanks between your columns. In that case, you must reenter the FORMAT SEPARATOR command with the keyword BLANKS.

ARI7529I **The top title has been erased. The first 100 characters of the SQL SELECT statement will be used as the top title.**

Explanation: Your top title has been erased. The first 100 characters of the SELECT command will be used as the top title on your report.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7530E **Command processing stopped. A right parenthesis is missing on the FORMAT** keyword **command. Reenter the command with matching parentheses.**

Explanation: You entered a left parenthesis without a matching right parenthesis.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the FORMAT command with matching parentheses.

ARI7531I **The separator you entered was too long. The separator length has been shortened to 254 bytes.**

Explanation: The length of the separator cannot be greater than 254.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: None, if you do not mind the shortened separator. Otherwise, enter a shorter separator, using the FORMAT command.

ARI7532I **You cannot do the GROUP and SUBTOTAL on the same column. Column** column id **has been ignored in the FORMAT** keyword **command, but processing continues.**

Explanation: You tried to group and subtotal on the same column. Only one will be done. The message states which keyword will not be processed.

column id is the column id you entered.

keyword is the keyword you entered on the FORMAT command.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: If you don't want this result, reformat your display. See the *DB2 Server for VSE & VM Interactive SQL Guide and Reference* manual.

ARI7533I **Because there are no columns to group, subtotaling will no longer be done.**

Explanation: You entered a FORMAT GROUP ERASE command, so there are no more columns to group by. Therefore, subtotaling can no longer be done.

System Action: All subtotals are erased. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: If you want subtotals, enter a FORMAT GROUP command followed by a FORMAT SUBTOTAL command.

ARI7534E **Command processing stopped. You entered an incorrect column ID** token **on the FORMAT** keyword **command. All columns will be displayed. Reenter the command with a correct column ID.**

Explanation: You entered either FORMAT EXCLUDE ALL BUT or FORMAT INCLUDE ONLY with an incorrect column id. All columns will be included in the display.

token is the incorrect column id you entered.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the FORMAT command with one or more column ids. Column id must be either a column name or a column number. If you enter more than one column id, they must be enclosed in parentheses.

ARI7535E **Command processing stopped. You did not enter any column IDs between the parentheses on the FORMAT** keyword **command. All columns will be included in the display. Reenter the FORMAT** keyword **command with a list of column IDs.**

Explanation: You did not enter any column ids between the parentheses on your FORMAT command.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Reenter the FORMAT command with a list of column ids between the parentheses.

ARI7537E **Command processing stopped. You did not enter any column IDs between the parentheses on the FORMAT** keyword **command. Reenter the FORMAT** keyword **command with a list of column IDs.**

Explanation: You did not enter any column ids between the parentheses on the FORMAT command.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all

other keywords are ignored.

User Response: Reenter the FORMAT command with a list of column ids between the parentheses.

ARI7539I **The VARCHAR value you entered was too large. The VARCHAR size has been changed to 254 bytes.**

Explanation: You entered a FORMAT VARCHAR command with a value greater than 254.

System Action: The command was processed with a VARCHAR size of 254.

ARI7543I **Your display data exceeds 45 columns. When using ISQL, no more than 45 columns can be displayed or formatted. Only the first 45 columns you selected will be displayed.**

Explanation: ISQL allows a maximum of 45 columns to be displayed or formatted. Any additional columns will not be displayed.

System Action: The command is processed normally. Only the first 45 columns will be displayed.

User Response: If you want to display columns beyond the first 45, enter a SELECT command specifically requesting those columns.

ARI7544I **More than one data row was affected by the previous statement. If you do not want the work committed, issue CANCEL or ROLLBACK. If you want the work committed, enter any other statement.**

Explanation: An INSERT, DELETE, or UPDATE command affected more than one row and the AUTOCOMMIT is delayed.

If referential constraints exist with a rule of SET NULL or CASCADE then some of the affected rows may be in dependent tables.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: You can either continue with work (thereby committing the previous work), or enter CANCEL or ROLLBACK.

ARI7545E **Command processing stopped. Your FORMAT command needs a keyword. Enter FORMAT with one or more of the following keywords: ON, OFF, SUBTOTAL, TOTAL, GROUP, OUTLINE, SEPARATOR, VARCHAR, NULL, TTITLE, BTITLE, COLUMN, EXCLUDE, or INCLUDE.**

Explanation: You entered a FORMAT command without any keywords.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Enter the command with one or more valid keywords.

ARI7546E **Command processing stopped. Your FORMAT** keyword **command needs an option. Reenter the FORMAT** keyword **command with ON or OFF.**

Explanation: You entered an incomplete FORMAT command. The keyword must be followed by a value of ON or OFF.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Enter the command with a valid option, either ON or OFF.

ARI7547E **Command processing stopped. Your FORMAT** keyword **command needs a column ID. Reenter FORMAT** keyword **command with a column ID. Only the first 45 columns can be formatted.**

Explanation: You entered an incomplete FORMAT command. The keyword must be followed with a column id.

keyword is the keyword you entered on the FORMAT command.

System Action: Command processing has been stopped at the keyword indicated in the message. Any keywords before this keyword have been processed; all other keywords are ignored.

User Response: Enter the command with a column id.

ARI7550D **Warning: A logical unit of work is in progress. If an SQL error occurs, a ROLLBACK may result that will undo all the work you have done since the start of the logical unit of work. Do you want to process the statement anyway? Enter 0(No) or 1(Yes).**

Explanation: AUTOCOMMIT is OFF. Therefore, if an SQL error occurs while processing your statement, everything in this logical unit of work will be rolled back.

System Action: The system waits for you to enter `0` (NO) or 1 (YES).

User Response: If you are not worried about losing any work you have done, enter 1. To save the work you have done before processing this statement, enter `0`. Then you can enter COMMIT or SET AUTOCOMMIT ON. Then you can reenter your statement.

ARI7551I **Statement cancelled as requested.**

Explanation: You entered CANCEL.

System Action: The statement in process is cancelled, and a ROLLBACK RELEASE is done. The system waits for you to enter the next statement.

The ROLLBACK RELEASE statement causes the database manager to treat any later SQL requests in this terminal session as coming from the default user ID for the virtual machine in which ISQL is running. Most ISQL users are not affected by this action. However, if you had issued an explicit CONNECT statement so that the database manager treated your SQL requests as coming from a user ID *other than* the user ID of the virtual machine you are using, and you want to enter more SQL requests in this terminal session, you must reenter that CONNECT statement to fully recover from the ROLLBACK RELEASE action. The SQL user ID and target database name will be reset to their defaults after the ROLLBACK RELEASE action.

ARI7552E **Command failed. You did not enter the name on the** cmd **command.**

Explanation: You did not enter the name parameter required by the command specified in the message.

cmd is the command you entered.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Reenter the command with the name of the stored SQL statement you want.

ARI7553E **Command failed. Name entered is longer than 8 characters. Reenter the command with a name that is 8 or less characters long.**

Explanation: You entered a stored query name that was longer than 8 characters.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Reenter the command with an alphanumeric name of up to 8 characters.

ARI7554I **There is no previous SQL statement to recall.**

Explanation: You tried to do a RECALL PREVIOUS, but there is no previous SQL statement to recall.

System Action: The system waits for you to enter the next statement.

ARI7555I **There is no current SQL statement to recall.**

Explanation: You tried to do a RECALL, but there is no current SQL statement to recall.

System Action: The system waits for you to enter the next statement.

ARI7556E **Command failed. There is no current SQL statement to store.**

Explanation: You tried to store an SQL statement, but there is no SQL statement to store.

System Action: The system waits for you to enter the next statement.

ARI7557I **Excess keywords after the** ISQL command **command are ignored.**

Explanation: There are extra keywords on the ISQL command. They are ignored.

ISQL command is the name of the command you entered.

System Action: The command is processed. The extra data is ignored.

ARI7558E **Command failed. Stored query table not found. Notify your system programmer.**

Explanation: SQLDBA.STORED QUERIES table was either not created at installation, or was dropped.

System Action: The command is not processed.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred,

make a record of what went wrong and contact the designated support group for your installation.

ARI7559I Command ended as requested.

Explanation: You asked for the command to be ended.

System Action: The system waits for you to enter the next statement.

ARI7560E Command failed. The SQL statement you wanted to ISQL command **was not found. Try again.**

Explanation: You entered an ISQL command with an SQL statement name that was not found.

ISQL command is the ISQL command you entered.

System Action: The system waits for you to enter the next statement.

User Response: Check the spelling of the SQL statement name you entered, or enter LIST SQL * to see the names of your stored SQL statements. Then reenter your ISQL command.

ARI7561E Command failed. You entered a RENAME command without a new name for your stored SQL statement. Reenter the RENAME command with the old name followed by the new name.

Explanation: You tried to rename a stored SQL statement, but you did not enter a new name for the command.

System Action: The system waits for you to enter the next statement.

User Response: To rename your stored SQL statement, reenter the RENAME command with the old name followed by the new name.

ARI7563E Command failed. A stored SQL statement named name **already exists. Choose another name and try again.**

Explanation: You tried to rename a stored SQL statement. The new name you chose already exists, so you must choose another name.

name is the name you entered as the new name for the SQL statement.

System Action: The system waits for you to enter the next statement.

User Response: Reenter the RENAME command with a different new name. To see the names of the SQL statements you have already stored, enter LIST SQL *.

ARI7564E Command failed because of an SQL error. Notify your system programmer.

Explanation: AUTOCOMMIT is ON. An SQL error has occurred while trying to commit work for your ISQL command.

System Action: The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7568E Command failed because of an SQL error. A ROLLBACK statement has been issued.

Explanation: An SQL error has occurred that forced a ROLLBACK to be issued. All work in this logical unit of work has been rolled back.

System Action: A ROLLBACK has been issued. The system waits for you to enter the next statement.

User Response: Read the previous SQL error message and, if necessary, notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7569E Command failed because of an SQL error.

Explanation: An SQL error occurred which caused the system to stop processing your command.

System Action: The system waits for you to enter the next statement.

User Response: Read the previous SQL error message and, if necessary, notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7570I Processing for the ISQL command **command is complete.**

Explanation: Processing is done for the indicated command.

ISQL command is the name of the command you entered.

System Action: The system waits for you to enter the next statement.

ARI7572E **Command failed. You cannot use the word PREVIOUS. Reenter the command with a different name.**

Explanation: PREVIOUS is a reserved word used to refer to the previous SQL statement. You cannot STORE, ERASE, or RENAME an SQL statement with a name of PREVIOUS. You can RECALL PREVIOUS.

System Action: The command is ignored; no processing is done. The system waits for you to enter the next statement.

User Response: Reenter the command with a different name.

ARI7574E **Command failed. No SQL statement found on the HOLD command.**

Explanation: You entered a HOLD command without including an SQL statement.

System Action: The command is ignored; no processing is done. The system waits for you to enter the next statement.

User Response: Reenter the HOLD command, including the SQL statement.

ARI7575E **Command failed. An invalid SQL statement was found on the HOLD command.**

Explanation: You entered a HOLD command with an invalid SQL statement.

System Action: The command is ignored; no processing is done. The system waits for you to enter the next statement.

User Response: Reenter the HOLD command with a valid SQL statement.

ARI7576D **Enter END, REPLACE, or a new name which is 1 to 8 characters long. You cannot choose the word PREVIOUS as a name.**

Explanation: Your reply to message ARI7577D could not be used as a stored SQL statement name.

System Action: The system waits for your reply. If the reply is REPLACE, the stored SQL statement is replaced. If the reply is END, processing of the store command ends. If a name other than PREVIOUS is entered, the SQL statement is stored under the new name.

User Response: Enter REPLACE, END, or a new name for the SQL statement.

ARI7577D **A stored SQL statement named** name **already exists. Enter a new name to store the SQL statement, or enter one of the following keywords: REPLACE - to replace the existing stored SQL statement, or END - to end the store command processing.**

Explanation: The system cannot store the SQL statement under the name you have given it because that name is already in use.

name is the name you entered for the SQL statement.

System Action: The system waits for your reply. If the reply is REPLACE, the stored SQL statement is replaced. If the reply is END, processing of the store command ends. If a name other than PREVIOUS is entered, the SQL statement is stored under the new name.

User Response: Enter REPLACE, END, or a new name for the SQL statement to be stored.

ARI7578I **The SQL statement named** name **has been stored.**

Explanation: This message indicates that the SQL statement has been stored.

name is the SQL statement name you entered.

System Action: The system waits for you to enter the next statement.

ARI7579I **The stored SQL statement named** name **has been replaced.**

Explanation: This messages indicates that the previous copy of the stored SQL statement has been replaced.

name is the SQL statement name you entered.

System Action: The system waits for you to enter the next statement.

ARI7580I **The processing to store the SQL statement named** name **has been stopped by your request. The SQL statement has not been stored.**

Explanation: This message indicates that, at your request, the storing of the SQL statement has not been done. This message is displayed when you reply END to message ARI7576D or message ARI7577D.

name is the SQL statement name you entered.

System Action: The system waits for you to enter the next statement.

ARI7581I **There was no stored SQL statement named** name**. Your SQL statement has been stored as a new command.**

Explanation: This message is displayed when the STORE command is entered with the REPLACE option, but there is no stored SQL statement by that name to be replaced.

name is the SQL statement name you entered.

System Action: The system stores your SQL statement as a new command. The system waits for you to enter the next statement.

ARI7582I **REPLACE is the only optional keyword on the STORE command.** token **has been ignored.**

Explanation: This message indicates that excess data was found on the STORE command, instead of the optional keyword REPLACE.

token is the excess data on the STORE command.

System Action: The STORE command processing continues.

ARI7583E **Command failed because of an error in STORE command processing. Notify your system programmer.**

Explanation: An error condition occurred in the processing of the STORE command.

System Action: The command is not processed.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7585I **The stored SQL statement named** name **has been erased.**

Explanation: This message indicates that a stored SQL statement has been erased.

name is the SQL statement name you entered.

System Action: Command processing continues if there are more SQL statement names on the command. Otherwise, the system waits for you to enter the next command.

ARI7590I **An SQL statement name must be no more than 8 characters long.**

Explanation: The name under which an SQL statement is stored cannot exceed 8 characters. You must choose a shorter name.

System Action: Command processing continues if there are more SQL statement names on the command. Otherwise, the system waits for you to enter the next command.

ARI7591I **PREVIOUS cannot be used as a name for a stored SQL statement.**

Explanation: The name PREVIOUS is reserved for use by the system. You must choose a different name.

System Action: Command processing continues if there are more SQL statement names on the command. Otherwise, the system waits for you to enter the next command.

ARI7592I **The stored SQL statement named** name **was not found.**

Explanation: A stored SQL statement specified on the LIST SQL statement or ERASE command was not found.

name is the SQL statement name you entered.

System Action: Command processing continues if there are more stored SQL statement names on the command. Otherwise, the system waits for you to enter the next command.

ARI7593I **Excess data** token **is ignored.**

Explanation: You entered more than a new stored SQL statement name in response to message ARI7576D or message ARI7577D.

token is the excess data that was ignored.

System Action: The STORE command processing continues.

ARI7601I **ISQL ended normally on your request.**

Explanation: ISQL ending message. This message is displayed when you enter the EXIT command and autocommit processing was on.

System Action: ISQL ends.

ARI7602D **You are in a logical unit of work. Enter COMMIT to have a COMMIT issued for you, or ROLLBACK to have a ROLLBACK issued for you.**

Explanation: This message is displayed when you enter the EXIT command and autocommit processing is off, or when you enter SET AUTOCOMMIT ON and autocommit processing is off.

System Action: The system waits for your reply. The action taken by the system for each reply is as follows:

COMMIT A COMMIT is issued and a message is generated stating whether the commit work was successful or not. If

the command entered was EXIT and the commit work was successful ISQL ends. If the command entered was EXIT and the commit work was not successful, ISQL does not end. If the command entered was SET AUTOCOMMIT ON, autocommit processing is turned on only if the commit work was successful.

ROLLBACK A ROLLBACK is issued and a message is generated stating whether the rollback work was successful or not. If the command entered was EXIT and the rollback work was successful, ISQL ends. If the command entered was EXIT and the rollback work was not successful, ISQL does not end. If the command entered was SET AUTOCOMMIT ON, autocommit processing is turned on only if the rollback work was successful.

User Response: Give one of the specified replies.

ARI7603I {COMMIT | ROLLBACK} successful. ISQL now ended.

Explanation: Message displayed when you exit from ISQL, and the commit work or rollback work that is performed is successful.

System Action: ISQL ends.

ARI7604E ISQL-command **command failed. {COMMIT | ROLLBACK} not successful. See the previous SQL messages for a complete explanation. Enter the** ISQL-command **command again or enter another command.**

Explanation: You replied COMMIT or ROLLBACK to message ARI7602D, and the COMMIT or ROLLBACK was not successful.

ISQL-command is the ISQL command you entered.

System Action: The system waits for you to enter the next statement.

User Response: Try the command again or enter another command. Notify your system programmer if the problem continues.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7607I There is nothing to ignore.

Explanation: You entered the ISQL IGNORE command when you were not continuing a command or data.

System Action: The system waits for you to enter the next statement.

ARI7611E The CHANGE command nulled out the SQL statement buffer. Enter your next SQL statement.

Explanation: You entered a CHANGE command which nulled out the current SQL buffer.

System Action: The system waits for you to enter the next statement.

User Response: Enter the desired SQL statement.

ARI7612E CHANGE command failed. character string **was not found in your current SQL statement. Check the character string you want to change and try again.**

Explanation: The character string you entered on the CHANGE command was not found in your current SQL statement.

character string is the character string you wanted to change. If the string is more than 20 characters long, only the first 20 characters are shown in the message.

System Action: The system waits for you to enter the next statement.

User Response: Enter RECALL to see the current SQL statement. Then enter the CHANGE command with the character string you want to change.

ARI7613I CHANGE completed.

Explanation: The change was completed by the CHANGE command.

System Action: The changed SQL statement is displayed. The system waits for you to enter the next statement.

ARI7614E ISQL command **command failed. There is nothing in the SQL statement buffer. Enter your next SQL statement.**

Explanation: You entered a START or CHANGE command but there was no SQL statement in the SQL statement buffer.

ISQL command is the ISQL command you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the desired SQL statement.

ARI7615E ISQL command **command failed. The SQL statement name** name **on the** ISQL command **command is too long. Use an SQL statement name which is 8 characters long or fewer.**

Explanation: An SQL statement name more than eight characters long was entered with an ISQL command.

name is the SQL statement name you entered.

ISQL command is the ISQL command you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the command with a stored SQL statement name which is not more than eight characters long.

ARI7616I ISQL command **command cancelled at your request.**

Explanation: CANCEL was entered during the processing of a command.

ISQL command is the ISQL command that was being processed.

System Action: Command processing stops. A ROLLBACK RELEASE is issued. The system waits for you to enter the next statement.

The ROLLBACK RELEASE statement causes the database manager to treat any later SQL requests in this terminal session as coming from the default user ID for the virtual machine in which ISQL is running. Most ISQL users are not affected by this action. However, if you had issued an explicit CONNECT statement so that the database manager treated your SQL requests as coming from a user ID *other than* the user ID of the virtual machine you are using, and you want to enter more SQL requests in this terminal session, you must reenter that CONNECT statement to fully recover from the ROLLBACK RELEASE action. The SQL user ID and target database name will be reset to their defaults after the ROLLBACK RELEASE action.

ARI7617E ISQL command **command failed. A nonzero code was returned during the processing of the** ISQL command **command. See the previous message for the explanation and the action to take.**

Explanation: A DB2 Server for VM error was encountered while processing the ISQL command.

ISQL command is the ISQL command that was being processed.

System Action: Command processing stops. The system waits for you to enter the next statement.

User Response: Try the command again or enter another command. Notify your system programmer if the problem continues.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7618E ISQL command **command failed. SQL statement** name **was not found. Enter the** ISQL command **command with the name of one of your stored SQL statements.**

Explanation: You tried to list or start an SQL statement that was not previously stored.

ISQL command is the ISQL command you entered.

name is the SQL statement name you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the name of a stored SQL statement on the ISQL command, or enter LIST SQL * to see all your stored SQL statements.

ARI7619E ISQL command **command failed. There was a missing parenthesis or no parentheses on the parameter list of the** ISQL command **command. Enter the command with parentheses around the parameter list.**

Explanation: There was no ending parenthesis on the parameter list, an uneven number of parentheses in the parameter list, or no parentheses around the parameter list of the command.

ISQL command is the ISQL command you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the command with the corrected parameter list.

ARI7620I **You have** number **stored SQL statements.**

Explanation: This is a count of your stored SQL statements displayed when LIST SQL * is entered.

number is the actual count.

System Action: The system waits for you to enter the next statement.

ARI7621I **The counter for the LIST SQL statement is full. Counting of stored SQL statements will stop, but listing will continue.**

Explanation: You have more than 2 million stored SQL statements.

System Action: Listing of the stored SQL statements continues.

ARI7622E ISQL command **command failed. ISQL command** ISQL command **not allowed at this time. Enter any command except display commands or input commands.**

Explanation: You entered an ISQL command which could not be processed now. For example, this message is generated if you enter a display command when you are not displaying the results of a SELECT.

ISQL command is the ISQL command you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter an allowable command.

ARI7623E token **is not an SQL or an ISQL command. Enter any command except display commands or input commands.**

Explanation: You did not enter a valid SQL or ISQL command.

token is what was entered.

System Action: The system waits for you to enter the next statement.

User Response: Enter a command.

ARI7624E **HELP command failed.** topic **is not a HELP topic. Enter HELP or HELP CONTENTS for a list of topics.**

Explanation: Your entry for the topic on the HELP command is not part of the HELP documentation.

topic is the topic you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter an ISQL HELP command in one of the following forms:

```
HELP
HELP CONTENTS
HELP topic
```

ARI7625E **HELP command failed. The topic on the HELP command is more than 20 characters long. Enter HELP or HELP CONTENTS for a list of topics.**

Explanation: Your entry for the topic on the HELP command could not be processed because it was more than 20 characters long.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter an ISQL HELP command in one of the following forms:

```
HELP
HELP CONTENTS
HELP topic
```

ARI7626E ISQL command **command failed. There is not enough main storage available. Try again later.**

Explanation: The virtual storage needed for processing the command was not available.

ISQL command is the ISQL command you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Try the command again. If the condition continues, notify your system programmer.

System Programmer Response: If this condition persists, increase the size of the user's virtual machine.

ARI7627E ISQL command **command failed. The previous SQL statement cannot be used on the** ISQL command **command. Enter the** ISQL command **command with another stored SQL statement name.**

Explanation: The previous SQL statement cannot be processed by this ISQL command.

ISQL command is the ISQL command you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Use another name for the SQL statement.

ARI7628I **The formatting information for your SQL statement has been erased because you changed the SELECT list or the table name. The default formatting information will be used for displaying the results of your SQL statement.**

Explanation: You entered a CHANGE command which could make the format information for the SQL statement incorrect. The format information is dropped.

System Action: Your SQL statement result will be

formatted with the default formatting information that is in effect when the SQL statement is started. The system waits for you to enter the next statement.

ARI7629E **CHANGE command failed. The format of the CHANGE command is CHANGE /string1/string2/ or CHANGE /string1//. Enter the CHANGE command by using one of these formats.**

Explanation: The CHANGE command you entered was not in the right format.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the command with a correct format.

ARI7630E **CHANGE command failed. The change you requested cannot be made because it would make the SQL statement longer than 2048 characters. Enter the CHANGE command with string2 the same size or smaller than string1.**

Explanation: You entered a CHANGE command which would make the SQL statement in the SQL statement buffer longer than 2 048 characters.

System Action: Command processing stops. No change is made to the SQL statement in the current buffer. The system waits for you to enter the next statement.

User Response: Enter the CHANGE command with string2 the same size or smaller than string1.

ARI7631E **Enter the START command with the corrected parameter list or correct the parameter in your SQL statement.**

Explanation: This message is displayed when the parameter substitutions could not be performed on the SQL statement in the current buffer. Refer to the previous message to determine the cause of the problem.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Refer to the previous message for the action to take.

ARI7632E oper command **operator command failed. A system error occurred during the** oper command **operator command. Notify your system programmer.**

Explanation: An incorrect parameter was passed to ARIICMD from ARIIOCI (interface module to operator command processing).

oper command is the DB2 Server for VM operator command being executed.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7634E oper command **operator command failed. A system error occurred during the** oper command **operator command. See the previous message for a complete explanation. Notify your system programmer.**

Explanation: There was a problem in the DB2 Server for VM operator command processing modules. The message before this one is an SQL message describing the problem.

oper command is the DB2 Server for VM operator command being executed.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7635I oper command **operator command ended.**

Explanation: The indicated operator command has ended.

oper command is the DB2 Server for VM operator command executed.

System Action: The system waits for you to enter the next statement.

ARI7636D **Enter END to end the** oper command **command or press the Enter key to see more information.**

Explanation: Message asking if you want to end the operator command (reply END) or continue (press the ENTER key).

oper command is the DB2 Server for VM operator command being executed.

System Action: The system waits for your reply. Command processing stops if you reply END. The next data is displayed if you just press the ENTER key. The question is asked again if your reply is not END or ENTER.

User Response: Reply END or just press the ENTER key.

ARI7637E oper-command **operator command failed. Operator commands cannot be issued while you are in a logical unit of work. Enter COMMIT or ROLLBACK and then enter the** oper-command **operator command.**

Explanation: You tried to execute an operator command while in a logical unit of work.

oper-command is the DB2 Server for VM operator command you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter a COMMIT or ROLLBACK command, and then enter the operator command again.

ARI7638I **A parameter is in the SELECT list or table name in your SQL statement. Therefore, any formatting you do while viewing the query result will be erased when you end the display.**

Explanation: You entered one or more parameters in the SELECT-list or table name portion of your SQL statement. Therefore, you cannot save any of the formatting information you enter while viewing the query result.

System Action: Any formatting information you enter while viewing the query result is erased when you end the display. The system waits for you to enter the next statement.

User Response: If you want to save the formatting information, use the CHANGE command to change the parameters to values, and start the SQL statement with no parameter list.

ARI7639D **Last of the** oper command **operator command information. Enter END to end the display.**

Explanation: This is the last screen of the operator command output. You must reply END when finished with the display. Command cleanup is then done and the command ends.

oper command is the DB2 Server for VM operator command you entered.

System Action: The system waits for your reply. The display ends if your reply is END. You are asked for another reply if your reply is not END.

User Response: Reply with END when finished looking at the last screen of output.

ARI7640E **SQL statement ignored. SQL statements cannot be processed while using the INPUT command. To process your SQL statement, enter END and then reenter your SQL statement.**

Explanation: You entered an SQL statement while using the INPUT command. Your SQL statement was not processed.

System Action: The system waits for you to enter the next statement.

User Response: If you want to process your SQL statement now, end the INPUT command and then reenter your SQL statement. Otherwise, continue to input your data.

ARI7641D char **will be used as the delimiter for the CHANGE command. Do you want to continue? Enter 0(No) or 1(Yes).**

Explanation: The / was not used as the delimiter on the CHANGE command. If you forgot the slash delimiters, you can now end the CHANGE command.

char is the delimiter used if the CHANGE command continues.

System Action: The system waits for your reply. If your reply is 0, CHANGE command processing stops. If your reply is 1, the CHANGE command is processed.

User Response: Reply 0 to cancel the CHANGE command or 1 to continue the command.

ARI7642I **CHANGE command cancelled. No change was made to the SQL statement buffer.**

Explanation: Informative message generated when you reply NO to message ARI7641D. The CHANGE command is ended. The SQL statement in the current buffer is not changed.

System Action: The SQL statement buffer and formatting information remain the same. ROLLBACK WORK is not performed. The system waits for you to enter the next statement.

ARI7643E **HELP command failed. HELP documentation table** table name **was not found. Notify your system programmer.**

Explanation: One of the HELP documentation tables was not there.

table name is the name of the table that was not there (either SQLDBA.SYSTEXT1 or SQLDBA.SYSTEXT2).

System Action: The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Recreate the tables using the HELP text installation procedures provided in the *DB2 Server for VM Program Directory*.

ARI7644E **HELP command failed. No documentation is in the table SQLDBA.SYSTEXT2 for the topic that was entered. Notify your system programmer.**

Explanation: The HELP text table SQLDBA.SYSTEXT2 did not contain documentation for the topic specified, but the HELP text table SQLDBA.SYSTEXT1 listed the topic.

System Action: The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Recreate the tables using the HELP text installation procedures provided in the *DB2 Server for VM Program Directory*. If the problem persists, make a record of what went wrong and contact the designated support group for your installation.

ARI7645E **EXIT command failed. Extra information is on the EXIT command. Enter the EXIT command with no keywords or values.**

Explanation: You entered something extra after the word EXIT.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Reenter the EXIT command with no extra data on it.

ARI7646E **HELP command failed. The HELP text is not installed. If you want to use this facility, notify your system programmer.**

Explanation: You entered the HELP command, but the HELP text table SQLDBA.SYSTEXT1 is empty.

System Action: The system waits for you to enter the next statement.

User Response: Continue your ISQL session. Notify your system programmer if you would like to use the HELP facility.

System Programmer Response: Procedures for installing the HELP text tables are provided in the *DB2 Server for VM Program Directory*.

ARI7650E **The EXTRACT command** command **has been ignored because the Extract Facility is no longer supported. Enter an SQL or an ISQL command.**

Explanation: You entered an extract facility command, but the database manager no longer supports the extract facility.

command is the extract facility command you entered.

System Action: The system waits for you to enter the next statement.

User Response: Continue your ISQL session.

ARI7651I **Press ENTER to continue or type END to end display.**

Explanation: This message is displayed on the last line of the operator command display on VM.

System Action: The system waits for you to press the ENTER key or enter END.

User Response: Press the ENTER key if you wish to see more of the operator command display, or enter END if you do not.

ARI7652I **Last of information. Enter END to end display.**

Explanation: This message is displayed at the bottom of the last screen of the operator command display on VM.

System Action: The system waits for you to enter END.

User Response: Enter END when you are done looking at the operator command display.

ARI7700I **Extra information was on the** ISQL command **command.** token **is ignored but processing will continue.**

Explanation: Extra data was found on the command you entered.

ISQL command is the ISQL command you entered.

token is the extra information that was ignored.

System Action: Normal command processing continues.

ARI7701E ISQL command **command processing stopped. The keyword** keyword **on the** ISQL command **command is not correct. The valid keywords are:** keyword list**.**

Explanation: The keyword you entered on the command was not valid.

keyword is the keyword you entered.

ISQL command is the ISQL command you entered.

keyword list is the list of the valid keywords.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the command with a valid keyword.

ARI7702E **The keyword** token **on the LIST SET command is not a valid choice. The valid keywords are:** keyword list**.**

Explanation: You entered an incorrect keyword on the LIST SET command.

token is the incorrect keyword you entered.

keyword list is the list of keywords allowed on the command.

System Action: Processing of the keyword is bypassed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7709I **AUTOCOMMIT processing is {ON|OFF}.**

Explanation: This message shows the current AUTOCOMMIT setting.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7710I **The continuation character is** char**.**

Explanation: This message shows what the continuation character is.

char is the continuation character now in effect.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7711I **The system default print class will be used for your reports.**

Explanation: Your print class is now set to the system default.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7712I **The** keyword **value is** value**.**

Explanation: This message shows the value of a keyword you requested.

keyword is the keyword you entered on the LIST SET command.

value is the value of that keyword.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7713I number **copies of your reports will be printed.**

Explanation: This message shows how many copies of each report will be printed.

number is the number of copies that will be printed.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7714I **The page size of your reports is: WIDTH:** width **LENGTH:** length**.**

Explanation: This message shows the page size of your reports.

width is the width of the printed page.

length is the length of the printed page.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7715I **The separator value is** number **blanks.**

Explanation: This message indicates how many blanks will separate the columns when a query result is displayed.

System Action: Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7716I **User** *userid* **connected to server** *server-name***.**

Explanation: This message shows the current user and the current application server that are accessed as a result of the last implicit or explicit CONNECT command.

System Action: The system waits for you to enter the next command.

ARI7717I **One or more arithmetic exception errors occurred. Exception error** &1**. has occurred during** &2**. operation on** &3**. data, position** &4**..**

Explanation: This message is issued whenever an arithmetic operation on numeric data causes an arithmetic exception error. An example of this error may be a division by zero or an overflow (exceeds maximum) during the processing of a SELECT command. The data from this error is denoted by number signs (# # #) across the width of columns of the queried table:

&1. is the type of exception error.
&2. is the arithmetic operation on data.
&3. is the data type.
&4. is position N, where N is an item in select list where applicable. Position `0` is returned if an exception error occurs in the WHERE clause of the SELECT statement.

Note: The message describes the first arithmetic exception error in the queried table.

System Action: Command processing continues.

ARI7725E **The** ISQL command **command processing stopped. The command is not complete. You must specify** information**.**

Explanation: You entered a command that did not have all the required keywords or values.

ISQL command is the ISQL command you entered.

information is a list of what you did not specify on the command.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the command with all needed keywords and values.

ARI7726E **The** ISQL command **command processing stopped. The value you entered could not be processed. The values that can be processed are** valid values**.**

Explanation: Something was wrong with one of the values on the command.

ISQL command is the ISQL command you entered.

valid values is the description of the valid values.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the command with valid values.

ARI7727I **The old** keyword **value was** old**. The new** keyword **value is** new**.**

Explanation: The keyword value has been changed to the value you entered on the SET command.

keyword is the keyword you entered on the SET command.

old is the old value of the keyword.

new is the new value of the keyword.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7729I **The** keyword **string is more than** number **characters long. The first** number **characters will be used.**

Explanation: You entered a separator string or null string with the SET command that was longer than the maximum allowed.

keyword is the type of string you entered.

number is the number of characters that will be used.

System Action: The indicated number of characters of the string is saved.

ARI7731I **The SET AUTOCOMMIT command has been cancelled. The AUTOCOMMIT setting is not changed.**

Explanation: You entered SET AUTOCOMMIT ON, and then cancelled that action with a CANCEL command.

System Action: Command processing stops; the AUTOCOMMIT value is not changed. The system waits for you to enter the next statement.

ARI7732I **To commit work, you must enter the COMMIT statement yourself.**

Explanation: You entered SET AUTOCOMMIT OFF. If you want to commit work, you must enter the COMMIT statement.

System Action: The ISQL performs COMMIT statements or ROLLBACK statements only if you request them. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

User Response: You must enter COMMIT to make the changes that you make to tables permanent, or enter ROLLBACK to back out the changes.

ARI7733E **SET command processing stopped. The** keyword **value** data **could not be processed. The** keyword **value must be** valid data**.**

Explanation: You entered a SET command with a value that is not valid.

keyword is the keyword you entered.

data is the data you entered.

valid data is a list of acceptable values for the keyword.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the command with a correct value.

ARI7734E **SET command processing stopped. The** keyword **value** data **is too long. Enter the command with a value that is 1 character long.**

Explanation: You entered a value that was too long to be a class or continuation character value.

keyword is the keyword you entered on the SET command.

data is the value you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the command with a correct value.

ARI7735I **The** keyword **value on the SET** keyword **command is too large. It has been set to the maximum value** number**.**

Explanation: You entered a SET command with a keyword value greater than the maximum for that keyword.

keyword is the keyword you entered on the SET command.

number is the maximum value for that keyword.

System Action: The value has been set to the maximum value for that keyword. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7736E **SET command processing stopped.** char **cannot be a continuation character. The continuation character can be any character except a blank, a quote, or a semi-colon.**

Explanation: You entered a continuation character that was not valid.

char is the continuation character you entered.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter any continuation character except a semicolon, a single or double quote, or a blank.

ARI7737I action **is successful. All COMMIT statements and ROLLBACK statements will now be issued for you automatically.**

Explanation: You replied COMMIT or ROLLBACK to message ARI7602D during the SET AUTOCOMMIT command.

action is either COMMIT or ROLLBACK.

System Action: AUTOCOMMIT has been set on. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7739E **SET command processing stopped. No WIDTH or LENGTH keyword was specified with the PAGESIZE keyword. Reenter the command with the WIDTH and/or LENGTH keywords.**

Explanation: You entered a SET PAGESIZE command that was not correct.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the command with the correct keywords and values.

ARI7740E **SET command processing stopped. The** keyword **value cannot be zero. The** keyword **value can be** valid values**.**

Explanation: You entered a SET command with a value of zero. This is not processed because zero is not a meaningful value for the keyword.

keyword is the keyword you entered.

valid values is the valid range for the keyword.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Reenter the command with a value in the given range.

ARI7742E **SET command processing stopped. The** keyword **on the SET PAGESIZE command must be** valid data**.**

Explanation: You entered a SET PAGESIZE command and the width or length was not valid.

keyword is the keyword that had a value that could not be processed.

valid data is the range of valid values for the keyword.

System Action: Processing of the command stops. The system waits for you to enter the next statement.

User Response: Enter the SET PAGESIZE command with correct values for the width and length.

ARI7743I **The old column separator was** separator**.**

Explanation: You changed the column separator. This message shows what the old separator was.

separator is the old column separator value.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7744I **The new column separator is** separator**.**

Explanation: You changed the column separator. This message shows what the new separator is.

separator is the new column separator value.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7745I **The** keyword **value on the SET PAGESIZE command is too small. It has been set to the minimum value** number**.**

Explanation: You entered a SET command with a keyword value that was less than the allowable minimum for that keyword.

keyword is the keyword you entered on the SET PAGESIZE command.

number is the minimum value for the keyword.

System Action: The page width or page length has been set to the minimum value for that keyword. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7746I **The** keyword **value on the SET PAGESIZE command is too large. It has been set to the maximum value** number**.**

Explanation: You entered a SET command with a keyword value that was greater than the allowable maximum for that keyword.

keyword is the keyword you entered on the SET PAGESIZE command.

number is the maximum value for the keyword.

System Action: The page width or page length has been set to the maximum value for that keyword. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

ARI7755I **The** ISQL command **command does not accept the** keyword **keyword on CMS. Use CP commands to set and list defaults.**

Explanation: You entered a keyword that is not supported in the CMS environment.

ISQL command is the ISQL command you entered.

keyword is a keyword that is not supported on CMS.

System Action: The system waits for you to enter the next statement.

ARI7757I **The old page size width was:** old width**. The new page size width is:** new width**.**

Explanation: The SET PAGESIZE command was entered with a WIDTH parameter only. Therefore, this message states only new and old WIDTH values.

old width is the previous page width value.

new width is the new page width value.

System Action: The system waits for you to enter the next statement.

ARI7758I **The old page size length was:** old length**. The new page size length is:** new length**.**

Explanation: The SET PAGESIZE command was entered with a LENGTH parameter only. Therefore, this message states only the new and old LENGTH values.

old length is the previous page length value.

new length is the new page length value.

System Action: The system waits for you to enter the next statement.

ARI7759I **Your reports will be routed to the system printer.**

Explanation: This message indicates that your printed output will be routed to the system printer.

System Action: Command processing continues if there are more keywords. Otherwise, the system waits for you to enter the next command.

ARI7885E **The PRINT command failed. CLASS must be followed by a value from 0 to 9 or A to Z. Try again.**

Explanation: You entered the CLASS keyword without a value.

System Action: The system waits for you to enter the next statement.

User Response: Reenter the PRINT command with a CLASS value from 0 to 9 or A to Z.

ARI7886E **The PRINT command failed. You entered a CLASS value more than 1 character long. Reenter the PRINT command with a CLASS value from 0 to 9 or A to Z.**

Explanation: You entered the CLASS keyword with a value more than 1 character long.

System Action: The system waits for you to enter the next statement.

User Response: Reenter the PRINT command with a CLASS value from 0 to 9 or A to Z.

ARI7887E **The PRINT command failed. The CLASS value you entered is not correct. CLASS value must be either a letter or a number. Try again.**

Explanation: You entered an incorrect CLASS value. It must be either a letter or a number.

System Action: The system waits for you to enter the next statement.

User Response: Reenter the PRINT command with a CLASS value from 0 to 9 or A to Z.

ARI7888E **The PRINT command failed. Follow the COPIES keyword with a value. Reenter the PRINT command with a COPIES value of 1 to 255.**

Explanation: You entered the COPIES keyword without a value.

System Action: The system waits for you to enter the next statement.

User Response: Reenter the PRINT command with a COPIES value from 1 to 255.

ARI7889E **The PRINT command failed. Follow COPIES with a number from 1 to 255. Try again.**

Explanation: You entered the COPIES keyword with a value greater than 255.

System Action: The system waits for you to enter the next statement.

User Response: Reenter the PRINT command with a COPIES value from 1 to 255.

ARI7890E **The PRINT command failed. The COPIES value you entered was not a number. Reenter the PRINT command with a COPIES value of 1 to 255.**

Explanation: You entered the COPIES keyword with a value that is not a number.

System Action: The system waits for you to enter the next statement.

User Response: Reenter the PRINT command with a COPIES value from 1 to 255.

ARI7891E **The PRINT command failed. The system does not accept a COPIES value of 0. Follow the COPIES keyword by a value from 1 to 255. Try again.**

Explanation: You entered the COPIES keyword with a value of 0. The system does not accept a COPIES value of 0.

System Action: The system waits for you to enter the next statement.

User Response: Reenter the PRINT command with a COPIES value from 1 to 255.

ARI7892E **The PRINT command failed. The PRINT command does not accept the keywords TERMID, DESTID, and SYSTEM on CMS. Reenter the PRINT command without any of these keywords.**

Explanation: You entered TERMID, DESTID, or SYSTEM on the PRINT command. These keywords are not supported in the CMS environment.

System Action: The system waits for you to enter the next statement.

User Response: You must use the CP SPOOL and TAG commands to route your report to the printer you want. Enter these commands before you reenter the PRINT command.

ARI7894I **The width of the data to be printed is greater than the printer width. The data to be printed has been shortened to fit within the printer width.**

Explanation: The width of the data to be printed is greater than the system printer width.

System Action: The data will be printed up to a width of 132 characters.

User Response: There is nothing you need to do if you are satisfied with your report. To print further

data, use the COLUMN command to see the columns you want to print, and enter the PRINT command again.

ARI7895E The PRINT command failed. The printer is not ready. See your system programmer.

Explanation: The virtual printer 00E requires intervention.

System Action: The system will return to displaying the query result.

User Response: Inform your system programmer that system printer 00E requires intervention. You can continue your ISQL session.

System Programmer Response: Check the state of virtual printer 00E.

ARI7896E The PRINT command failed. A system problem exists. See your system programmer.

Explanation: An error of unknown cause has occurred while using the CMS PRINTL macro.

System Action: The system will return to displaying the query result.

User Response: Inform your system programmer that a system error has occurred. You can continue your ISQL session.

System Programmer Response: Check the state of virtual printer 00E.

ARI7897E The PRINT command failed. The printer is not attached. See your system programmer.

Explanation: System printer 00E is not attached to the system.

System Action: The system will return to displaying the query result.

User Response: Inform your system programmer that the system printer is not attached to the system. You can continue your ISQL session.

System Programmer Response: Check the state of virtual printer 00E.

ARI7898E The CP CLOSE command failed. See your system programmer.

Explanation: The system returned a nonzero completion code in executing the CP DIAGNOSE command to close the print file just created by the PRINT command.

System Action: The system will return to displaying the query result.

User Response: Inform your system programmer that the DIAGNOSE command to close the printer did not work correctly. You can continue your ISQL session.

System Programmer Response: Perform CP problem determination to determine why the DIAGNOSE command returned a nonzero return code.

ARI7901E The PRINT command failed. You cannot issue a PRINT command because you were not viewing a query result. Issue a SELECT statement.

Explanation: You entered a PRINT command while you were not viewing a query result. The PRINT command is valid only while a query result is being displayed.

System Action: The system waits for you to enter the next statement.

User Response: Enter a SELECT statement before entering the PRINT command.

ARI7902I You entered the PRINT command, but there is no data to be printed. Your report will contain only titles and column headings.

Explanation: You entered a PRINT command, but there is no data in your query result to be printed.

System Action: The system will print your report. It will contain only titles and column headings. The system waits for you to enter the next statement.

ARI7903E The PRINT command failed. You entered an incorrect keyword token **on the PRINT command. Try again.**

Explanation: You entered an incorrect keyword on the PRINT command.

token is the incorrect keyword you entered.

System Action: The system waits for you to enter the next statement.

User Response: Correct the keyword in error and reenter the PRINT command. The correct keywords for PRINT are CLASS and COPIES.

ARI7905I The PRINT function is in progress.

Explanation: The PRINT command is being processed.

System Action: The system fetches the data and processes the PRINT command.

ARI7908I **You are trying to print a report that is wider than the page width. Only the first** number **characters of your report will be printed.**

Explanation: You tries to print a report that is wider than the page width you set.

number is the number of characters that will be printed.

System Action: Your report will be printed, but some of the columns will be cut off.

User Response: Enter the SET PAGESIZE command if you want to make your page wider. Reenter or start your SELECT command. Then reenter the PRINT command.

ARI7920E **A system problem exists. An SQL error was encountered. The PRINT is terminated. Notify your system programmer.**

Explanation: An unexpected indication was received when closing or opening the cursor.

System Action: The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. Check the correctness of the table. Reenter the PRINT command. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7921E **An SQL error was encountered. The PRINT will continue, but you may not have your full output.**

Explanation: An unexpected indication was received when reading a row of data.

System Action: Printing continues.

User Response: Check the output to see if the full table was printed. If the full output was not obtained, notify your system programmer.

System Programmer Response: Perform problem determination. Check the output to see if the full table was printed. Check the correctness of the table. Reenter the PRINT command. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7922I **The PRINT command has been cancelled.**

Explanation: The CANCEL command was entered or you indicated you did not want to continue in response to message ARI7923D.

System Action: The system waits for you to enter the next statement.

ARI7923D **This PRINT begins in column** number1 **position** number2**. Do you want to continue? Press Enter to continue. If you do not want to continue, enter 0(No).**

Explanation: The data as displayed does not start in column 1 position 1 of the table.

number1 is the numeric value of the leftmost column in the display.

number2 is the numeric value of the left most position within the leftmost column in the display.

System Action: If you press the ENTER key, the table will be printed as displayed, starting from the top of the table. If you enter NO, the print will be ended.

User Response: Enter a response to indicate whether to print the table from its current column (or tab) position or end this print.

ARI7924I **The PRINT function is completed.**

Explanation: The print function has been completed.

System Action: The system waits for you to enter the next statement.

ARI7932I **The COMMIT process is complete.**

Explanation: Your COMMIT statement completed normally.

System Action: Any changes made are committed to the database. The system waits for you to enter the next statement.

ARI7933I **The ROLLBACK process is complete.**

Explanation: Your ROLLBACK statement completed normally.

System Action: The database manager backs out any changes made since the last COMMIT or ROLLBACK. The system waits for you to enter the next statement.

ARI7934E **The** *action* **statement failed. A portion of the statement is incorrect following or involving** *token***.**

Explanation: The portion of the *action* statement following or involving *token* is incorrect.

System Action: The system waits for you to enter the next statement.

User Response: Correct the error and reenter the statement.

ARI7937E **You cannot use the SQL statement** sql command **in ISQL.**

Explanation: You entered an SQL statement that cannot be processed by ISQL.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: For details on how to use this command, see the *DB2 Server for VSE & VM Application Programming* manual.

ARI7939E token **is invalid. It must be either numeric or MAX for forward or backward.**

Explanation: Either:

- You entered a nonnumeric value where only a numeric value can be used.
- You entered a FORWARD or BACKWARD command with a nonnumeric value other than the keyword MAX.

token is the value you entered.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Reenter the command with a proper value.

ARI7940E **Not enough storage is available for processing the DISPLAY command you entered. Try the command at a later time or enter a different command.**

Explanation: You entered a command which requires more virtual storage than the amount now available.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Retry the command. If this condition persists, notify your system programmer.

System Programmer Response: If this condition persists, increase the size of the user's virtual machine.

ARI7942E **The command** cmd **is not a valid ISQL command. Please enter another command.**

Explanation: You entered a command that is not recognized by ISQL.

cmd is the command you entered.

System Action: The command is ignored. The system waits for you to enter the next statement.

ARI7943E **The field** token **is not supported by ISQL DISPLAY processing. Reenter the SELECT statement without specifying field** token**.**

Explanation: Your SQL statement explicitly or implicitly requests that a long field column is to be used. Long field data types are not supported by ISQL.

token is the name of the column.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Enter a different SQL statement or reenter the SELECT statement without retrieving the field specified in the message.

ARI7944E **System error. The field** token **is of a type unknown to ISQL. Notify your system programmer.**

Explanation: A column identified by *token* was retrieved from the database manager, but the data type of this column is not known to ISQL. Data types known to ISQL are: INTEGER, SMALLINT, DECIMAL, FLOAT, CHAR, VARCHAR, LONG VARCHAR, GRAPHIC, VARGRAPHIC, LONG VARGRAPHIC, DATE, TIME, and TIMESTAMP. However, long fields are not supported by ISQL display processing. A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(*n*) where *n* is greater than 254 but less than or equal to 32767, or VARGRAPHIC(*n*) where *n* is greater than 127 but less than or equal to 16383.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7945I **The CONNECT statement completed successfully.**

Explanation: Your CONNECT statement completed normally.

System Action: The specified user is now connected to the database manager. The system waits for you to enter the next statement.

ARI7946E **The CONNECT statement failed. If this message is followed by an SQL error message, follow its directions. If it is not followed by an SQL error message, check the format of the CONNECT statement and reenter it.**

Explanation: Your CONNECT statement failed.

System Action: The CONNECT is not processed. The system waits for you to enter the next statement.

User Response: Reenter the CONNECT statement after checking the USER ID and PASSWORD for validity. If the problem persists, notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7947E **An internal processing error has occurred. An invalid call occurred to the SQL interface module. Notify your system programmer.**

Explanation: An internal call was made to the SQL processing routine with a parameter list that was not valid.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7949I **Too many parameters were on the command. They will be ignored.**

Explanation: You entered an SQL command with too many fields specified.

System Action: The extra information on the command is ignored. The system waits for you to enter the next statement.

ARI7951I **The value specified would cause the DISPLAY to be positioned before the first or after the last column. The DISPLAY will be positioned as far as possible in the direction specified.**

Explanation: You specified a value on an ISQL display command that, if used, would have resulted in an incorrect display.

System Action: The system processes as much as possible of the command. The system waits for you to enter the next statement.

ARI7952E **You cannot use zero as a value for the COLUMN or TAB command. Reenter the command with a value greater than zero.**

Explanation: You entered a COLUMN or TAB command with a value of zero. Zero is not an acceptable value.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Reenter the COLUMN or TAB command with a value greater than zero.

ARI7953E **An internal processing error has occurred within the DISPLAY processor. Notify your system programmer.**

Explanation: An internal error was encountered in the ISQL display processor, ARIIDQY.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7954E **Query failed. Full query result row length exceeded** nnnn **positions. Retry query by selecting fewer columns or use the SET command to decrease the VARCHAR value.**

Explanation: The query result row length exceeds the maximum length of nnnn characters allowed by ISQL for the screen size of your terminal: The maximum row length is determined by taking the screen height, subtracting 4, multiplying this by 2, because SQL retrieves 2 screens worth of data, adding 1 and dividing this into the size of the buffer, 65535. The formula is:

```
65535 / (((SCREENHT - 4) * 2) + 1) = MAX ROW LENGTH.
```

This is the maximum size that a row can be for your terminal size.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Retry your query, selecting fewer columns, or use the SET command to decrease the VARCHAR value.

ARI7955I **The system ended your query result to process your command.**

Explanation: You entered a command that cannot be processed while you are viewing a query result.

System Action: The system ends the query result and processes the command.

ARI7957E The TAB command failed. The TAB command is valid for the CHAR and VARCHAR columns only. Tabbing into DECIMAL, FLOAT, INTEGER, and SMALLINT columns is not allowed.

Explanation: You can use the TAB command on CHAR and VARCHAR columns only.

System Action: The command is ignored. The system waits for you to enter the next statement.

User Response: Enter next command.

ARI7958I The COMMIT RELEASE process is complete.

Explanation: The COMMIT RELEASE statement you entered has completed normally.

System Action: Any changes made to the database are committed. The system waits for you to enter the next statement.

The COMMIT RELEASE statement causes the database manager to treat any later SQL requests in this terminal session as coming from the default user ID for the virtual machine in which ISQL is running. Most ISQL users are not affected by this action. However, if you had issued an explicit CONNECT statement so that the database manager treated your SQL requests as coming from a user ID *other than* the user ID of the virtual machine you are using, and you want to enter more SQL requests in this terminal session, you must reenter that CONNECT statement to fully recover from the COMMIT RELEASE action.

ARI7959I The ROLLBACK RELEASE process is complete.

Explanation: The ROLLBACK RELEASE statement you entered has completed normally.

System Action: Any changes made to the database are rolled back. The system waits for you to enter the next statement. The COMMIT RELEASE statement causes the database manager to treat any later SQL requests in this terminal session as coming from the default user ID for the virtual machine in which ISQL is running. Most ISQL users are not affected by this action. However, if you had issued an explicit ROLLBACK statement so that the database manager treated your SQL requests as coming from a user ID *other than* the user ID of the virtual machine you are using, and you want to enter more SQL requests in this terminal session, you must reenter that ROLLBACK statement to fully recover from the COMMIT RELEASE action.

ARI7960I The query cost estimate for this SELECT statement is number**.**

Explanation: You entered a SELECT statement. The database manager has calculated a cost estimate for this query. The larger this number is, the longer you will have to wait to see your query result.

number is the query cost estimate, ranging from 1 to 9999. If the query cost estimate is greater than 9999, the value ">9999" is presented in the message. (The query cost estimate is calculated by adding 1 to the result of dividing the DB2 Server for VM optimizer cost estimate by 1000.)

System Action: Command processing continues unless you enter the CANCEL command.

User Response: If you do not want to wait for the result, enter the CANCEL command.

ARI7962E Command failed. A severe SQL error has occurred. A CONNECT statement is required in order to continue this ISQL session. Report this message and the previous SQL error message to your system programmer.

Explanation: A severe SQL error occurred while your command was being processed.

System Action: If the command that failed was not a CONNECT command, then ISQL will prevent you from issuing further commands against the currently established database.

User Response: If the command being executed was not a CONNECT, then you must terminate the ISQL session, or you may switch to another database by using the CONNECT command. For more information about the CONNECT command, refer to the *DB2 Server for VSE & VM Database Administration* manual.

If this error occurs while switching databases using the CONNECT command, you will remain connected to the database in effect prior to attempting the switch.

Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI7963E Command failed. This command is not valid while you view a query result from a routine or an EXEC.

Explanation: You put a command in a routine that cannot be processed while viewing a query result. The command failed.

System Action: The next command in the routine will be processed, unless you have requested that routine

processing be stopped or canceled when an error is detected.

User Response: Remove the failing command from the routine. Run the routine again.

ARI7964I The display command is valid only from a routine or an EXEC. It was ignored.

Explanation: You entered a DISPLAY command from the terminal while viewing a query result. The DISPLAY command is valid only from a routine.

System Action: The command is ignored. The system waits for you to enter the next statement.

ARI7966I The value specified would cause the display to be positioned after the last character in the column. The display will be positioned at the rightmost character in the column.

Explanation: You specified a value on an ISQL display command that, if used, would have resulted in an incorrect display.

System Action: The system processes as much as possible of the command. The system waits for you to enter the next statement.

ARI7967E HELP text for language id **could not be found.**

Explanation: Either the topic name has been misspelled, or the HELP text for this topic cannot be found.

System Action: The system waits for you to enter the next command.

User Response: Verify the topic name and retry the command. If the topic cannot be found, contact your system programmer.

System Programmer Response: Verify the existence of HELP text for this topic in SYSTEXT2.

ARI7968E Invalid string of double-byte character set (DBCS) characters.

Explanation: A string of DBCS characters is either is too long, contains bytes that are not properly paired, or improperly terminated, as in having an opening so delimiter (X'0E') without a terminating si delimiter (X'0F').

System Action: Command processing is ended.

User Response: Correct the input command and re-enter it.

ARI7969I DISPLAY command has been processed successfully.

Explanation: This message indicates that the display command was processed successfully. In the case of a FORMAT TTITLE title or FORMAT BTITLE title, the top title or bottom title respectively have the value of title.

System Action: The keyword is processed. Command processing continues if more keywords are on the command. Otherwise, the system waits for you to enter the next command.

ARI7970I This DISPLAY command cannot be processed because of graphic data present.

Explanation: This message indicates that the display command entered was correct but because the data to be displayed has graphic data, the command cannot be processed. This is a restriction of FORMAT TTITLE and FORMAT BTITLE.

System Action: The keyword is processed. Command processing continues if there are more keywords on the command. Otherwise, the system waits for you to enter the next command.

Chapter 14. ARI0800–ARI0899, ARI8000–ARI8999 DBS Utility Messages

Common Error Action

- If the error was not a serious DB2 Server for VM error, and the processing mode established by a SET ERRORMODE CONTINUE command is in effect, DBS Utility command processing continues.
- If the error was not a serious DB2 Server for VM error, and the processing mode established by a SET ERRORMODE CONTINUE command is not in effect:
  1. SQL ROLLBACK statement is executed.
  2. The DBS Utility enters error mode processing (ERRORMODE=ON).
- If the error was a serious DB2 Server for VM error, the DBS Utility enters error mode processing (ERRORMODE=ON).
- During DBS Utility error mode processing, only a DBS Utility SET ERRORMODE OFF or a SET ERRORMODE CONTINUE command will be executed.

  Other SQL and DBS Utility commands are not executed. They are read from the command file and displayed in the DBS Utility message listing. The DBS Utility only performs command identification and DBS Utility command syntax checking for the commands that are ignored during error mode processing. Errors that would be detected during the execution of a command or a data file input/output operation are not detected.

Rerunning DBS Utility

If no COMMIT of database changes occurred during this execution of the DBS Utility, all commands that altered the database should be reprocessed through the DBS Utility.

If a COMMIT of database changes occurred during this execution of the DBS Utility, rerun the DBS Utility but omit all the commands from the control statement input that altered the database prior to the last COMMIT of database changes.

A COMMIT of database changes may have occurred as a result of processing a user-supplied SQL COMMIT statement or as a result of the AUTOCOMMIT=ON processing initiated by a user-supplied DBS SET AUTOCOMMIT ON command.

If AUTOCOMMIT=ON, message ARI0811I is generated after each successful COMMIT of database changes by DBS Utility processing.

ARI0800I **...Begin COMMIT. Input Record Count = *count-value*.**

Explanation: This message is issued before DATALOAD or RELOAD COMMITCOUNT processing issues a COMMIT statement. The successful completion of the COMMIT will be identified by the subsequent message:

```
ARI0811I ...COMMIT of any database
           changes successful.
```

count-value is the current input record count value. After the COMMIT statement successfully completes, this value should be used as the RESTARTCOUNT value (**rcount**) if DATALOAD or RELOAD processing fails before the next COMMIT statement successfully completes.

System Action: DBS utility processing continues.

ARI0801I **DBS Utility started: *&1. &2.*.**

AUTOCOMMIT=OFF
ERRORMODE=OFF
ISOLATION LEVEL=REPEATABLE READ.

Explanation: Identifies the start of DBS Utility processing and that AUTOCOMMIT=OFF and ERRORMODE=OFF, and ISOLATION LEVEL=REPEATABLE READ are in effect for command processing.

&1. and &2. are a 17-position date and time (mm/dd/yy hh:mm:ss) identifying when the message was issued.

System Action: Processing continues.

ARI0802I **End of command file input.**

Explanation: The last record has been read from the command file or an error has occurred that prevents further command file read processing. If a command file read error occurred, message ARI0814E will precede this message.

© Copyright IBM Corp. 1987, 2000

System Action: DBS Utility end of program processing is initiated.

ARI0803E Invalid command identifier.

Explanation: The last command displayed in the DBS Utility message listing cannot be identified. The command is not supported by DBS Utility processing or it does not begin with valid DBS Utility or SQL command identifier keywords.

All command file records are displayed as continuations of the invalid command up to but not including the command file record beginning the next recognizable command.

System Action: The command is not processed. See "Common Error Action" on page 223 for more information.

Programmer Response: The command is not processed. See "Common Error Action" on page 223 for more information.

ARI0804E Internal DBS error occurred: Module=*&1*. Reason Code=*&2A*.

Explanation: An internal DBS Utility error has occurred.

&1. is the name of the DBS Utility module detecting the error.

&2A. is a value that identifies the specific error detected by the module. The possible values and associated descriptions vary for each module and are described below.

After the message is generated, a storage dump is taken. Register 15 is set to a hexadecimal value unique to each module. These dump identification values (DUMP ID) are also described below.

DBS Module	Reason Code	DESCRIPTION	R15 Contents
ARIDALC	4	Invalid CALL to module. The call type parameter was not ALLOCATE (X'00') or FREE (X'08').	800
ARIDALI	4	Invalid CALL to module. The call type parameter was not ALLOCATE (X'00') or FREE (X'08').	801
ARIDALT	4	Invalid CALL to module. The call type parameter was not ALLOCATE (X'00') or FREE (X'08').	802
ARIDBS	4	Invalid pointer was passed to the module that loads the character translation table.	803
ARIDCFI	8	Invalid CALL to module. A CALL to the module was made to read the command file: • Before the command file was opened, • After the command file was closed, or • After a command file I/O error occurred.	804
ARIDCSP	4	An entry to the module, the command file status, and the ARIDCSP processing control fields were in an inconsistent state.	805
ARIDCSP	8	Command build logic error. During command build processing, the last nonblank position of the command file record was less than the first nonblank position.	805
ARIDDFI	8	Invalid CALL to module. A CALL to the module was made to read the command file: • Before the command file was opened • After the command file was closed, or • After a command file I/O error occurred.	806
ARIDDLO	4	The current command was not a DBS DATALOAD TABLE command when the module ARIDDLO was entered.	807

DBS Module	Reason Code	DESCRIPTION	R15 Contents
ARIDDLO	8	Command processing logic error. An ARIDCSP return code (R15) value > 8 was received while reading commands for actual processing (no previous errors have occurred).	807
ARIDDLO	12	A command sequence check logic error occurred during normal command sequence processing.	807
ARIDDLO	16	A logic error occurred during DATALOAD input OPEN processing. The last command processed was not an INFILE or INMOD command.	807
ARIDDLO	20	Command processing logic error. An ARIDCSP return code (R15) value > 8 was received while reading commands after an error occurred.	807
ARIDDLO	24	An input data field was not processed for a column during INSERT processing based on column nodes already matched to data fields.	807
ARIDDUL	4	The current command was not a DATAUNLOAD command when module ARIDDUL was entered.	813
ARIDDUL	8	Command processing logic error. An ARIDCSP return code (R15) value > 8 was received while reading commands for actual processing. (That is, no previous commands occurred.)	813
ARIDDUL	12	A command sequence check logic error occurred during normal command sequence processing.	813
ARIDDUL	20	Command processing logic error. An ARIDCSP return code (R15) value > 8 was received while reading commands after an error occurred.	813
ARIDEXI	4	Invalid CALL to module. The call type parameter was not OPEN (X'12'), FETCH (X'02'), or CLOSE (X'04').	808
ARIDSFA	4	Invalid CALL to module. The call type parameter was not ALLOCATE (X'00'), COMPUTE (X'02'), or FREE (X'04').	809
ARIDSQL	4	Invalid CALL to module. The call type parameter was not valid.	810
ARIDUNL	12	Internal logic error. The SQL I/O area is too small.	812

System Action: Processing of the command is terminated. A storage dump request is issued. See "Common Error Action" on page 223 for more information.

Programmer Response: Contact system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred,

make a record of what went wrong and contact the designated support group for your installation.

ARI0805E Command is incomplete.

Explanation: The last command displayed in the DBS Utility message listing is not complete.

All command file records were read or a command file read error occurred before the end of the command was identified.

The command file records comprising the command and any command file processing error messages appear before the ARI0805E message in the DBS Utility message listing.

Note: Unbalanced quotes in a command or command information appearing in positions 73 through 80 of a command file record may cause an erroneous end of file condition to occur.

System Action: The command is not processed. See "Common Error Action" on page 223 for more information.

Programmer Response: Refer to the previous messages in the DBS Utility message listing to identify the type of command being built and a description of any abnormal command file processing conditions.

Correct the command error or the command file problem identified and rerun the DBS Utility. (See "Rerunning DBS Utility" on page 223.)

ARI0806E The command length already is greater than 8192 positions.

Explanation: The last command displayed in the message listing is longer than 8192 positions. The maximum command length has already been exceeded before this control statement was read. The contents of this control statement cannot be appended to the command being built.

This message will only appear after the condition described by message ARI0810E has occurred.

System Action: Refer to message ARI0810E.

Programmer Response: Refer to message ARI0810E.

ARI0807E ...Errors occurred during command processing.

Explanation: Indicates that all user commands have not processed successfully during execution of the DBS Utility.

System Action: DBS Utility end of program processing continues.

Programmer Response: Refer to the previous messages in the DBS Utility message listing to identify the errors encountered.

ARI0808I DBS processing completed: *&1.&2.*.

Explanation: Identifies the end of DBS Utility command processing.

&1.&2. is a 17-position date and time (mm/dd/yy hh:mm:ss) identifying when the message was issued.

System Action: DBS Utility end of program processing continues. The message file is closed and processor storage areas acquired at the start of DBS Utility processing are released before DBS Utility processing terminates.

ARI0809I ...No errors occurred during command processing.

Explanation: All commands supplied were processed successfully, or all commands since the last SET ERRORMODE OFF command were processed successfully.

System Action: DBS Utility end of program processing continues.

ARI0810E Command file record truncated after position *&1A*..

Explanation: The last command displayed in the DBS Utility message listing exceeds the maximum command length of 8192 positions.

&1. is the position (column in the control statement record) of last control statement displayed that was identified as position 8192 of the command.

Note: If no commands approaching the maximum command length were present in the DBS Utility command file input, reasons for the error condition prompting this message are:

- Unbalanced quotes in a command.
- Command information appearing in positions 73 through 80 of a control statement record.

System Action:

- The command is not executed.
- DBS Utility continues to read command file records until the end of the command is encountered.
- Subsequent command file records comprising the command will cause the message ARI0806E to be issued.
- Command processing is terminated.
- Also see "Common Error Action" on page 223.

Programmer Response: Analyze the previous command file records displayed in the DBS Utility message listing.

Correct the command or if necessary, reduce the length of the command by eliminating unnecessary blanks between command parameters in each control statement, then reprocess the command. (See "Rerunning DBS Utility" on page 223.)

ARI0811I **...*&1*. of any database changes successful.**

Explanation: The DBS Utility initiated execution of an SQL COMMIT or SQL ROLLBACK command was successful. The command was not supplied as user input, but was executed automatically by DBS Utility processing. Any database changes made since the beginning of DBS Utility processing, or since the last SQL COMMIT or SQL ROLLBACK command was executed, have been committed or rolled back.

&1. is COMMIT or ROLLBACK.

System Action: DBS Utility processing continues.

Programmer Response: If a ROLLBACK of database changes is indicated, reference the previous error messages to identify the errors that caused the ROLLBACK and the action to be taken.

If a COMMIT of database changes is indicated, DBS Utility processing committed the database changes automatically because AUTOCOMMIT=ON or because ERRORMODE=OFF was in effect when the DBS Utility end of program processing was initiated.

ARI0812I **...Resume command execution: AUTOCOMMIT=OFF ERRORMODE=OFF.**

Explanation: A DBS SET ERRORMODE OFF command was successfully processed. DBS Utility will resume execution of commands at this point with AUTOCOMMIT=OFF and ERRORMODE=OFF.

System Action: Proceeds to process and execute commands with AUTOCOMMIT=OFF and ERRORMODE=OFF.

ARI0813I **...Suspend command execution: AUTOCOMMIT=OFF ERRORMODE=ON.**

Explanation: This message identifies that DBS Utility error mode processing is being initiated (AUTOCOMMIT=OFF and ERRORMODE=ON).

System Action:

- Proceeds with DBS Utility error mode processing.
- Command identification and DBS Utility command syntax checking will occur but no further commands will be executed.
- Errors that would be detected during execution of a command or any associated data file input/output operations will not be detected.

ARI0814E **File *&1.* error: DNAME=*&2*. Reason Code = *&3A*.. Return Code = *&4A*..**

Explanation: An input/output file processing error occurred.

&1. is either OPEN, CLOSE, READ (GET), or WRITE (PUT).

&2. is DNAME identifying the file.

&3A. is the Primary Error Code describing the error.

&4A. is the Secondary Error Code describing the error.

Primary Error Code: The Reason Code = &3A. value reflects an error detected by either a DBS Utility module or the DB2 Server for VM Operating System Dependent Service Module Generalized I/O Routine (ARISYSD5). The code values in the following list are provided by the DBS Utility:

91 Module ARIDMGE print message call, but the message number in the CPA field CPAMSGNO is 0.

92 Module ARIDMGE print line call (not a message), but the CPA field CPAMSGNO is not 0.

93 Module ARIDMGE print SQL message call, but the CPA field CPAMSGNO is not 0.

94 Module ARIDDFI detected an unsupported or incorrect record format (RECFM) after the file was successfully opened.

95 Module ARIDDUL detected an incorrect record format (RECFM) after the file was successfully opened. The DATAUNLOAD processing requires an output record length greater than 32760 bytes, but the output file identified is not defined with spanned records (RECFM = VS or VBS).

96 Module ARIDRLP detected an empty file. RELOAD PROGRAM processing requires a non-empty file and may be created by UNLOAD PROGRAM processing.

97 Module ARIDDFI detected a record format of variable-length spanned records (RECFM = VS or VBS) is used, but the output file identified does not have a filemode number equal to 4, after the file was successfully opened.

99 ARIDMGE detected an unknown error. The message processing status fields are in an inconsistent state.

All other Primary Error Code values are provided by ARISYSD5. For further explanation, look up the displayed error codes under either “Chapter 23. Sequential File Errors” on page 407, or “SEQ FILE ERRORS” in the HELP text.

Secondary Error Codes: The Return Code =&4A; value is an error code provided by the DB2 Server for VM Operating System Dependent Service Module Generalized I/O Routine (ARISYSD5).

If the Primary Error Code is provided by a DBS Utility module, the Secondary Error Code is zero. For further explanation, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Determine problem with the file identified by analyzing any reason code displayed in this message, and any DBS Utility messages displayed previous to this message or by operating system messages displayed on the operator console or terminal.

If possible, correct the error and reprocess the last command displayed in the DBS Utility message listing. (See "Rerunning DBS Utility" on page 223.)

If the error cannot be corrected, notify the system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0815I ...*&1*. Processing Mode = *&2*..

Explanation: This message is displayed as a result of processing a Database Services Utility SET AUTOCOMMIT or SET UPDATE STATISTICS command. It identifies whether DBS AUTOCOMMIT or UPDATE STATISTICS processing will be ON or OFF for subsequent command processing.

&1. is AUTOCOMMIT or UPDATE STATISTICS.

&2. is ON or OFF.

System Action: Processing proceeds.

ARI0816E DBS COMMENT command does not end with a single quote.

Explanation: An error has occurred in the last DBS COMMENT command displayed in the DBS Utility message listing. The terminating single quote for the comment text was not found.

System Action: See "Common Error Action" on page 223.

Programmer Response: Ensure that there are no unbalanced quotes in the command and that command information is not supplied in positions 73 through 80 of the command file records comprising the command. Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0817I ...Command file processing was interrupted.

Explanation:

- The command file was empty.
- A command file read error occurred before the first command was completed.
- An error occurred in attempting to start the DBS Utility in single user mode.
- A severe DB2 Server for VM error occurred while running the DBS Utility in single user mode.

System Action: DBS Utility end of program processing is initiated.

Programmer Response: If the command file contains no valid commands, then correct the command file. Rerun the DBS Utility.

If message ARI0814E precedes this message in the DBS Utility message listing, correct the command file error indicated. Rerun the DBS Utility.

If an error occurred in starting the DBS Utility in single user mode or if a severe DB2 Server for VM error occurred while running the DBS Utility in single user mode, examine the error messages preceding this message to determine the problem.

ARI0819I ...Command not processed: ERRORMODE=ON.

Explanation: The command displayed preceding this message in the DBS Utility message listing was not processed because of a previous error.

System Action: DBS Utility error mode processing continues.

ARI0821E Module *module-name* interface error. Reason Code = *reason-code*

Explanation: An error has been detected by module *module-name*. The INMOD command is displayed preceding this message in the DBS Utility message listing.

The *reason-code* is the return code issued by *module-name*. The values are:

```
8          Processing error.
```

System Action: See "Common Error Action" on page 223 for more information.

Programmer Response: Refer to the previous DB2 Server for VM messages in the message output listing to identify the error encountered and the action to be taken.

ARI0822E Commands are out of sequence or a subcommand is missing.

Explanation: The last command displayed is out of sequence. Either required DBS DATALOAD or DATAUNLOAD subcommands have been omitted, or the command file records are out of sequence.

System Action: The DATALOAD or DATAUNLOAD command is not executed. See "Common Error Action" on page 223 for more information.

Programmer Response: Supply all required commands and subcommands in the correct sequence and reprocess the commands. (See "Rerunning DBS Utility" on page 223.)

ARI0823E SQL message formatting error. SQLCODE=*&1A*..

Explanation: An internal DBS Utility error has occurred. The DBS Utility module ARIDSQL received a return code = 20 or 24 from the module ARIMSMF while attempting to format the descriptive text for the identified SQLCODE.

System Action: Processing continues.

Programmer Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0824E *&1*. storage unavailable. Bytes Requested = *&2A*. Reason Code = *&3A*.

Explanation: The identified DBS Utility module could not acquire the required amount of processor storage.

&1. is a module name.

&2A. is the amount of storage requested in bytes.

&3A. is a code describing the error (see below).

The code value (&3A.) is the return code supplied by the DB2 Server for VM Common Service Module (ARISYSD) in response to the acquire storage request made through the entry point ARISYSD1.

The code value is the same as the return code supplied by the CMS free storage macro. The free storage macro return codes are described in the *VM/ESA: CMS Application Development Reference for Assembler* manual.

System Action: See "Common Error Action" on page 223.

Programmer Response: Correct the problem identified by the reason code (&3A.) displayed in the message text.

If the problem cannot be corrected, contact the system programmer.

System Programmer Response: If the problem identified by the reason code (&3A.) cannot be corrected, perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0825E *&1. &2*. storage not released: Reason Code = *&3A*..

Explanation: The identified DBS Utility module could not release processor storage.

&1. is a module name.

&2. is an area name.

&3A. is a code describing the error (see below).

The code value (&3A.) is the return code supplied by the DB2 Server for VM Common Service Module (ARISYSD) in response to the acquire storage request made through the entry point ARISYSD2.

The code value is the same as the return code supplied by the CMS DMSFRET macro.

System Action: Processing of the last command displayed in the DBS Utility message listing is terminated. Also see "Common Error Action" on page 223 for more information.

Programmer Response: Correct the problem identified by the reason code (&3A.) supplied in the message text.

If the problem cannot be corrected, contact the system programmer.

System Programmer Response: If the problem cannot be corrected, perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0827I ...Begin command execution: ERRORMODE=CONTINUE.

Explanation: This message is generated after a DBS Utility SET ERRORMODE CONTINUE command was successfully processed. It is also generated during the DBS Utility program initialization if the control file is assigned to the terminal.

The DBS Utility will execute subsequent commands with the normal DBS Utility error processing suppressed. If a command processing error is detected, the Database Services Utility will neither execute an SQL ROLLBACK statement nor enter DBS Utility error mode processing. Subsequent commands will be executed by the DBS Utility. For a further explanation of this processing, refer to the description of the DBS Utility SET ERRORMODE command CONTINUE parameter.

System Action:

- Command processing continues.

- If other than a serious DB2 Server for VM error is detected, the appropriate error message or messages will be generated, but the DBS Utility will not execute an SQL ROLLBACK statement or enter DBS Utility error mode processing.
- If a serious DB2 Server for VM error is detected, DBS Utility command processing will be suspended.

ARI0828I ...*&1*. reset to *&2A*..

Explanation: This message is generated after a DBS Utility SET LINECOUNT/LINEWIDTH command was successfully processed. This message is also generated during the DBS Utility program initialization if the logical record length of the DBS Utility message file is not equal to 121 or the DBS Utility message file has been assigned to the terminal.

&1. LINECOUNT or LINEWIDTH.

LINECOUNT identifies the number of lines per print page to be used for the DBS Utility message file output.

LINEWIDTH identifies the number of print positions per line to be used for the DBS Utility message file output resulting from SQL SELECT command processing.

&2A. LINECOUNT or LINEWIDTH value to be used for the DBS Utility message file output.

System Action: Command processing continues.

ARI0829E Invalid message file LRECL value = *&1A*.

Explanation: This message is generated during the DBS Utility program initialization if the logical record length (LRECL) of the DBS Utility message file (ddname=SYSPRINT) is less than 81.

&1A. is an invalid DBS Utility message file logical record length (LRECL) value.

System Action: This message is written to the virtual machine console. DBS Utility processing ends with a return code = 16.

Operator Response: Contact the VM machine user.

Programmer Response: Correct the CMS FILEDEF command defining the DBS Utility message file (ddname=SYSPRINT). Supply a logical record length (LRECL) value equal to or greater than 121.

ARI0830E User-supplied SELECT statement error encountered.

Explanation: An error was detected during SQL processing of the SQL SELECT statement supplied by the user for DATAUNLOAD processing.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Refer to the error messages in the DBS Utility message listing prior to this message for a description of the SQL error encountered. Correct the SQL SELECT statement error and rerun the command.

ARI0831I Column *&1*. data will not be unloaded.

Explanation: This message identifies a column in the select-list of the DATAUNLOAD command that is not referenced by the user-supplied Data-Field-Identification subcommands. The data for the column will not be unloaded.

&1. is the column name of the select-list column that will not be unloaded.

System Action: DBS Utility processing continues.

ARI0832I Column *&1*. data truncation or data overflow may occur.

Explanation: This message identifies one of the following conditions:

- A select-list column defined with a nonnumeric data type and a length greater than the length of the associated output record CHAR, GRAPHIC, or ZONED data field. Nonblank, trailing (low-order) positions of column values will be truncated by DATAUNLOAD processing.
- A select-list column defined with a numeric data type that may result in a value with more significant digits than the length of the associated output record CHAR, GRAPHIC, or ZONED data field allows. Asterisks will be placed in the output record data field if this numeric data value overflow condition occurs.

&1. is a column name or integer value specified in the Data-Field-Identification subcommand that identifies a select-list column in the SQL SELECT statement supplied after the DATAUNLOAD command.

System Action: DBS Utility processing continues.

ARI0833E Column *&1*. data overflow. Rowcount = *&2A*..

Explanation: This message identifies an occurrence of a numeric select-list column value that exceeds the capacity of the output record numeric data field. The significant digits of the column value cannot be contained in the output record data field.

&1. is a column name or integer value specified in the Data-Field-Identification subcommand.

&2A. is an SQL SELECT row count at the time the condition was detected. For example, if this condition was detected in the tenth row selected from the

database by the SQL SELECT statement supplied for DATAUNLOAD processing, this value will be 10.

System Action: The row cannot be written to the output file. If the row is written to the output file, the data may be truncated or replaced by asterisks. DBS Utility processing will be stopped when end-of-input, SQLCODE 100 is detected.

Programmer Response: Verify that the output file contains the expected data. If it does not, correct the applicable dataunload Data-Field-Identification subcommand and rerun the command.

ARI0834E **Invalid data field position for column *&1*..**

Explanation: This message identifies that a data field for a column referenced by a DATAUNLOAD Data-Field-Identification subcommand starts or ends in an output record position exceeding the specified length of the output record. The length of the output record is identified in the previous ARI0868I message that appears in the DBS Utility message listing.

&1. is a column name or integer value specified in the Data-Field-Identification subcommand that identifies a select-list column in the SQL SELECT statement supplied after the DATAUNLOAD command.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the applicable DATAUNLOAD Data-Field-Identification subcommand or increase the length of the output record and rerun the command.

ARI0835I ***&1A*. records were written to the output data file.**

Explanation: Identifies the number of records written to the output data file by DATAUNLOAD processing.

&1A. is the number of records.

System Action: DBS Utility processing continues.

ARI0836I **Default output record data field positions:**

Explanation: This message precedes the ARI0837I messages that identify the default output record data field positions for a DATAUNLOAD command select-list.

System Action: DBS Utility processing continues. One or more ARI0837I messages will be generated after this message.

ARI0837I *&1. &2A.–&3A*..

Explanation: Identifies the default output record data field positions for a DATAUNLOAD command select-list column. This message follows message ARI0836I or another occurrence of message ARI0837I in the DBS Utility message listing.

&1. is the column name or integer value that identifies the source of the data.

&2A. is the default start position for the column data in the output data record.

&3A. is the default end position for the column data in the output data record.

System Action: DBS Utility processing continues.

ARI0838E **Invalid SQL CONNECT statement syntax.**

Explanation: The syntax of an SQL CONNECT statement is not valid. For example:

- An SQL CONNECT statement keyword or parameter is missing or misspelled.
- Extra information is specified.
- The statement is not terminated by a semicolon.

System Action: The command is not processed. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0839E **LINEWIDTH is not less than message record length = *&1A*..**

Explanation: An invalid value is specified for the LINEWIDTH parameter in the last command displayed in the DBS Utility message file listing. The value is equal to or greater than the DBS Utility message file record length.

&1A. is a DBS Utility message file record length.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command or define the DBS Utility message file with a larger record length and rerun the job. (See "Rerunning DBS Utility" on page 223.)

ARI0840E **Invalid data type = *&1A*. encountered in the SQLDA.**

Explanation: While processing the last command displayed in the DBS Utility message listing, an invalid SQLDA data type value was detected after the execution of a DBS-initiated SQL DESCRIBE command.

This is a DBS Utility program error or a DB2 Server for VM system error.

&1A. is the invalid data type detected.

System Action: Processing of the current command is terminated. Also see "Common Error Action" on page 223.

Programmer Response: Notify your system programmer.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0841E End of file or read error before data record completed.

Explanation: During DATALOAD continued record processing, the last data record read indicated that the data was continued in the following data record. An end of file condition or read error occurred before the continuation record was read.

System Action: DATALOAD command processing is ended. Also see "Common Error Action" on page 223.

Programmer Response: If a command file read error is indicated by the error message ARI0814E displayed previous to this message in the DBS Utility message listing, correct the file problem identified.

If a command file read error did not occur, correct the continuation indicator in the input data record.

Reprocess the command. (See "Rerunning DBS Utility" on page 223.)

ARI0842E COMMITCOUNT specified and DBS Utility AUTOCOMMIT = OFF.

Explanation: This message is issued when the COMMITCOUNT parameter is specified on the DATALOAD INFILE or RELOAD INFILE subcommand and Database Services Utility AUTOCOMMIT processing is OFF. A Database Services Utility SET AUTOCOMMIT command must be issued prior to running a DATALOAD or RELOAD command with COMMITCOUNT processing in effect.

System Action: DATALOAD or RELOAD command processing is ended.

Programmer Response: Rerun after supplying a SET AUTOCOMMIT ON command before the DATALOAD or RELOAD command or removing the COMMITCOUNT parameter from the DATALOAD INFILE or RELOAD INFILE subcommand.

ARI0843I Restart processing at input data record = *&1A*..

Explanation: This message identifies the input record count at which DATALOAD or RELOAD processing will begin after the number of records specified by the RESTARTCOUNT value (rcount) are read and skipped. The value (&1A.) is one more than RESTARTCOUNT value.

System Action: Processing continues.

ARI0844E End of data or read error occurred during RESTART processing.

Explanation: This message is issued when an input file read error or end-of-file condition occurs before the number of records specified in the RESTARTCOUNT value (rcount) are read.

System Action: Command processing is ended.

Programmer Response: Either a bad input file was used or an incorrect RESTARTCOUNT value (rcount) was specified. Rerun using a good input file or with the correct RESTARTCOUNT value.

ARI0845I Process input table data: *&1..&2.*

Explanation: The input table identified will be processed by the RELOAD DBSPACE or the RELOAD TABLE command.

- If NEW was specified in the RELOAD DBSPACE or the RELOAD TABLE command:
  - The input table definition was used to create a new table in the database.
  - The input table data will be loaded into the database table created.
- If PURGE was specified in the RELOAD DBSPACE command:
  - The input table exists in the DBSPACE specified in the command.
  - All rows will be deleted from the database table.
  - The input table data will be loaded into the database table.
- If PURGE was specified in the RELOAD TABLE command:
  - The output table exists in the database.
  - All rows will be deleted from the database table.
  - The input table data will be loaded into the database table.

&1. is the input table owner.

&2. is the input table name.

System Action: DBS Utility processing continues.

ARI0846I Ignore input table data: *&1..&2.*

Explanation: The input table identified will not be processed by the RELOAD DBSPACE command.

- If NEW was specified in the RELOAD DBSPACE command, a table cannot be created in the database using the input table definition, because that table already exists in the database.
- If PURGE was specified in the RELOAD DBSPACE command, the input table does not exist in the DBSPACE specified in the command.

&1. is the input table owner.

&2. is the input table name.

System Action: DBS Utility processing continues.

ARI0847I ...No input tables processed.

Explanation: No tables were loaded as a result of processing the last RELOAD DBSPACE command displayed in the DBS Utility message listing.

If the last command displayed is a RELOAD DBSPACE command with the NEW option, none of the tables in the input file could be created and loaded because they already exist in the database.

If the last command displayed is a RELOAD DBSPACE command with the PURGE option, none of the tables in the input file could be loaded because they do not exist in the specified DBSPACE.

System Action: Processing continues.

Programmer Response: None if the desired results were produced.

If the desired results were not produced, review the commands displayed in the DBS Utility message listing, supply the correct command, and reprocess.

ARI0848E Reload of system tables not allowed.

Explanation: The last command displayed in the DBS Utility message listing is a Database Services Utility RELOAD command that attempted to load one of the DB2 Server for VM catalog tables. Any table with an owner = SYSTEM cannot be loaded with a Database Services Utility RELOAD command.

System Action: Processing of the command is terminated. Also see "Common Error Action" on page 223.

Programmer Response: Do not attempt to load a catalog table with a Database Services Utility RELOAD command.

ARI0849E Definition of the input table *&1..&2.* **is not identical to the definition of the current table** *&3..&4..&5***.**

Explanation: The last command displayed in the DBS Utility message listing requested that an existing application server table be loaded with erroneous input file data.

The definition of the table from which the input file table data originated is not identical to the definition of the target table in the application server.

System Action: Processing of the last command displayed in the DBS Utility message listing is terminated. Also see "Common Error Action" on page 223.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0850I SQL SELECT processing successful: Rowcount = *&1A.***.**

Explanation: Identifies that the DBS Utility processing of an SQL SELECT statement has completed without error.

&1A. is the number of rows selected.

System Action: Processing continues.

ARI0851E SQL SELECT processing unsuccessful: Rowcount = *&1A.***.**

Explanation: Identifies that the processing of an SQL SELECT statement by the DBS Utility was not successful.

&1A. is the number of rows retrieved by the SQL SELECT statement before the error was encountered.

System Action: See "Common Error Action" on page 223.

Programmer Response: Refer to the previous messages in the DBS Utility message listing for a description of the errors encountered.

ARI0852I *parm1 parm2* **processing started.**

Explanation: This message identifies the start of actual execution of the indicated command.

- *parm1* is the primary command name.
- *parm2* is the DBSPACE/TABLE/INDEX/blank/PACKAGE.

If no subsequent messages appear in the DBS Utility message listing, VM or the DB2 Server for VM system may have terminated the DBS Utility package. Check the virtual machine console messages to determine the reason for termination.

System Action: Processing continues.

ARI0853I *parm1 parm2.parm3.parm4* **successful.**

Explanation: The indicated DBS Utility processing for a database table or DBSPACE has completed successfully.

- *parm1* is the UNLOAD or RELOAD.
- *parm2* is the application server.
- *parm3* is the owner of table/blank/DBSPACE/program.
- *parm4* is the table name/DBSPACE name/program name.

System Action: Processing continues.

ARI0854E *parm1 parm2.parm3.parm4* **unsuccessful.**

Explanation: The indicated DBS Utility processing for a database table or DBSPACE has not completed successfully:

- *parm1* is UNLOAD or RELOAD.
- *parm2* is the application server.
- *parm3* is the owner of table/blank/DBSPACE/program.
- *parm4* is the table name/DBSPACE name/program name.

System Action: See "Common Error Action" on page 223.

Programmer Response: Correct the command errors identified by previous messages in the DBS Utility message listing and reprocess the command. (See "Rerunning DBS Utility" on page 223.)

ARI0855I *&1. &2.* **processing successful.**

Explanation: The DBS Utility processing of the identified command has completed successfully.

&1. is the primary command name.

&2. is TABLE, INDEX or blank.

System Action: Processing continues.

ARI0856E The following SQL error and/or warning conditions occurred for a DBS-initiated SQL *&1*. statement.

Explanation: While processing the last command displayed in the DBS Utility message listing, an SQL warning or error condition occurred for the DBS Utility initiated execution of an SQL statement.

The SQL messages that immediately follow this message describe the SQL conditions encountered.

&1. identifies the SQL statement executed by the DBS Utility that resulted in the SQL error or warning conditions.

System Action: Processing of the last command displayed in the DBS Utility message listing is terminated. Also see "Common Error Action" on page 223.

Programmer Response: Analyze the SQL problem described in the following SQL messages. If possible, correct the error and reprocess the command. (See "Rerunning DBS Utility" on page 223.)

If the SQL error indicated cannot be corrected, contact your system programmer.

System Programmer Response: If the SQL error cannot be corrected, it is a probable database manager program error. Gather all EXEC and DBS Utility output (include the DBS Utility generated messages, any storage dumps generated, and the virtual machine console output related to the EXEC) then contact the designated support group for your installation.

ARI0857E *&1. &2.* **processing unsuccessful.**

Explanation: The DBS Utility processing of the identified command has not completed successfully.

&1. is the primary command name.

&2. is TABLE, INDEX or blank.

System Action: See "Common Error Action" on page 223.

Programmer Response: Refer to previous error messages in the DBS Utility message listing for a description of the errors encountered and the action to take.

ARI0858E DATALOAD of more than 16 tables is not allowed.

Explanation: More than 16 DATALOAD TABLE commands were entered before an INFILE subcommand. The concurrent load of more than 16 tables through a single sequence of Database Services Utility DATALOAD TABLE commands is not allowed.

System Action: DATALOAD TABLE processing is ended. Also see "Common Error Action" on page 223.

Programmer Response: Reduce the number of DATALOAD TABLE commands preceding the INFILE subcommand to 16 and reprocess the commands. (See "Rerunning DBS Utility" on page 223.)

ARI0859E Invalid column reference.

Explanation: The last DATALOAD TABLE Table-Column-Identification (TCI) or DATAUNLOAD Data-Field-Identification (DFI) subcommand displayed in the DBS Utility message listing contains an invalid column name or column reference. The column is not in the table specified in the last DATALOAD TABLE command or in the select-list of the SQL SELECT statement for the last DATAUNLOAD command displayed in the DBS Utility message listing.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Determine the correct column name or column reference, alter the command and rerun.

ARI0860E ...ENDDATA command missing.

Explanation: An end of command file input condition occurred while reading the DATALOAD TABLE command input data records embedded within the command file. The ENDDATA command identifying the end of embedded data records was not encountered before the end of command file input condition occurred.

The end of command file input condition may have occurred because of a command file read error or because all command records were read.

System Action: DATALOAD TABLE processing is terminated unsuccessfully. Also see "Common Error Action" on page 223.

Programmer Response: If a command file read error is indicated by the error message ARI0814E displayed previous to this message in the DBS Utility message listing, correct the file problem identified.

If a command file error did not occur, supply an ENDDATA command after the last embedded input data record in the command file input.

Reprocess the command. (See "Rerunning DBS Utility" on page 223.)

ARI0861E NULL/CURRENT clause specified for column defined as NOT NULL or invalid data type.

Explanation: The last DATALOAD TABLE Table-Column-Identification (TCI) or DATAUNLOAD Data-Field-Identification (DFI) subcommand displayed in the DBS Utility message listing contains:

- A null-clause or set-null-clause, but the associated column does not allow null values.
- A current-clause, but the associated column is not defined as a (correct) DATETIME or CHAR data type.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0862E SQL INSERT error (table name is *&1..&2..&3.*).

Current input record count is *&4A.*.

Explanation: An SQL error or warning condition occurred while processing the last or last group of DBS Utility commands displayed in the DBS Utility message listing. The input record count when the error occurred is also identified. If *insert blocking is not being used* for the table identified in the message, the record count displayed will be the record in error. If *insert blocking is being used* for the table, the record count is the count up to the end of the current block. The record in error may be a previous record in the block. Setting COMMITCOUNT to 1 will turn off blocking.

The actual table being processed when the SQL condition occurred is identified in the message:

&1. is the server-name.

&2. is the table owner.

&3. is the table name.

&4A. is the input record count.

The SQL error or warning condition is identified by the preceding SQL messages in the DBS Utility message listing.

Hexadecimal Display:

A hexadecimal display of the data record being processed when the SQL error occurred will follow the message text. If DB2 Server for VM *insert blocking is not being used* for the table identified in the message, the data record containing the bad data value will be displayed. If *insert blocking is being used* for the table, the data record displayed will be the current data record being processed when the DBS Utility processing detected the SQL error. The bad data value may be in the record displayed or in a previous data record.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0863E *&1..&2..&3. &4.* data field positions exceed input record length.

Explanation: A table column data field is not complete in an input data record. The start position or end position of the table column data field exceeds the length of an input data record.

&1. is the server-name.

&2. is the table owner.

&3. is the table name.

&4. is the column name.

System Action: DATALOAD TABLE processing is ended. Also see "Common Error Action" on page 223.

Programmer Response: Correct the DATALOAD TABLE Table-column-identification subcommand. Rerun the job. (See "Rerunning DBS Utility" on page 223.)

ARI0864E Data field is longer than the column.

Explanation: The last DATALOAD TABLE Table-Column-Identification (TCI) displayed in the DBS Utility message listing specifies:

- An input record CHAR data field length greater than the length of the target CHAR column, or
- An input record DECIMAL data field with a precision or scale greater than the precision or scale of the target DECIMAL column, or
- An input record ZONED data field with a scale greater than the scale of the target DECIMAL column.
- An input record 8-byte FLOAT data field with a length greater than the 4-byte FLOAT target column.

System Action: DATALOAD TABLE processing is ended. Also see "Common Error Action" on page 223.

Programmer Response: Correct the command or correct the input data. Rerun the DBS Utility. (See "Rerunning DBS Utility" on page 223.)

ARI0865E Invalid data field data type for column.

Explanation: The last DATALOAD TABLE Table-Column-Identification (TCI) or DATAUNLOAD Data-Field-Identification (DFI) subcommand displayed in the DBS Utility message listing specifies an invalid input or output record field data type for the associated table or select-list column.

System Action: DATALOAD TABLE processing is ended. Also see "Common Error Action" on page 223.

Programmer Response: Correct the command or correct the input data. Reprocess the command. (See "Rerunning DBS Utility" on page 223.)

ARI0866E Input record data field error: Column = *&1*.. Input Record Count = *&2A*..

Explanation: A data conversion error occurred.

The input data could not be converted to the table column numeric data type because the input data field contains one of the following errors:

- Nonnumeric character data.
- Blanks embedded within the character data.
- A data value that is too small or too large for the target column.

&1. is the table column name.

&2A. is the input record count at the time the error occurred.

Hexadecimal Display:

A hexadecimal display of the data record containing the incorrect data value will follow the text.

System Action: DATALOAD TABLE processing is ended. Also see "Common Error Action" on page 223.

Programmer Response: Correct the input data. Rerun the DBS Utility. (See "Rerunning DBS Utility" on page 223.)

ARI0867E Invalid data field length.

Explanation: The last DATALOAD TABLE Table-Column-Identification (TCI) or DATAUNLOAD Data-Field-Identification (TFI) subcommand, displayed in the DBS Utility message listing, specifies an invalid length for a FIXED, FLOAT, REAL, DOUBLE PRECISION, GRAPHIC, TIME, DATE or TIMESTAMP data record data field:

- Data record data fields with the data type FIXED must be 1, 2, or 4 bytes in length.
- Data record data fields with the data type FLOAT must be 4 or 8 bytes in length.
- Data record data fields with the data type REAL must be 4 bytes in length.
- Data record data fields with the data type DOUBLE PRECISION must be 8 bytes in length.
- Data record data fields with the data type GRAPHIC must be an even number (2, 4, 6, .. 100 ...,) of bytes in length.
- Data record data fields with the data type DATE must be at least 10 bytes in length.
- Data record data fields with the data type TIME must be long enough to accommodate the full length of the time format being used.
- Data record data fields with data type TIMESTAMP must be at least 19 bytes in length.

System Action: DATALOAD TABLE processing is ended. Also see "Common Error Action" on page 223.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0868I DNAME=*&1*. RECFM=*&2*. RECSZ=*&3A*. BLKSIZE=*&4A*..

Explanation: This message contains the input or output file specifications being used by DBS Utility command processing to open the input or output data file.

The information was derived from the FILEDEF command specifications for the file or the DBS Utility default specification values.

This message will be generated after an input or output data file has been successfully initialized (opened) for processing.

&1. is the INFILE or OUTFILE dname parameter.

&2. is the record format specification:

S or VS	= Variable, spanned, unblocked
SB or VBS	= Variable, spanned, blocked
F	= Fixed, unblocked
FB	= Fixed, blocked
V	= Variable, unblocked
VB	= Variable, blocked

Note: If RECFM=VBS is specified in the CMS FILEDEF command for an output file, the record format is changed to RECFM=VS if the required logical record length is greater than the block size minus 4.

&3A. is the logical record length specification.

Note: For input or output data files with variable-length spanned records, the actual logical record length may be greater than the record length displayed in the message.

&4A. is the block size specification.

System Action: DBS Utility processing continues.

ARI0869I RECFM, RECSZ, BLKSZ, and/or PDEV specifications ignored.

Explanation: This message identifies that the RECFM, RECSZ, BLKSIZE, or PDEV command parameters specified in the preceding command displayed in the DBS Utility message listing are being ignored. This information will be derived from the specifications in the CMS FILEDEF command for the file.

System Action: DBS Utility processing continues.

ARI0870I Enter the command terminated by semicolon or enter EXIT to end.

Explanation: When the DBS Utility control file (SYSIN) is assigned to the terminal, this message is written to the terminal when the user should enter the next command or end DBS Utility processing.

If the user wishes to continue DBS Utility processing, the next valid command should be entered at the terminal.

If the user wishes to end DBS Utility processing, the EXIT command should be entered at the terminal. Alternatively, to end DBS Utility processing, the user can enter one of the following:

EXIT
EXIT;
QUIT
QUIT;
HX
SQLHX

System Action: Waits for the next terminal entry by the user.

User Response: Enter next command or enter EXIT to end DBS Utility processing.

ARI0871E Table *&1..&2..&3.* not found.

Explanation: The identified table was not found in SYSTEM.SYSCATALOGS.

&1. is the server-name.

&2. is the table owner.

&3. is the table name.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Ensure the table name specified in the command is correct.

If incorrect, correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

If correct, the table must be created before it can be processed by an UNLOAD or DATALOAD command or the NEW parameter must be specified in the RELOAD TABLE command.

ARI0872E DBSPACE *&1.* not found.

Explanation: The identified DBSPACE is not in the database. If owner.DBSPACE name was specified in the last command displayed in the DBS Utility message listing, that specific DBSPACE was not found. If only a DBSPACE name was specified in the last command, neither a private or public DBSPACE with this name was found.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Ensure that the DBSPACE name specified in the UNLOAD DBSPACE or RELOAD DBSPACE command is correct.

If incorrect, correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

If correct, the DBSPACE must be acquired and tables created in the DBSPACE before the DBSPACE can be unloaded.

ARI0873E DBSPACE *&1..&2..&3.* contains no tables.

Explanation: The identified DBSPACE is empty.

&1. is the server-name.

&2. is the DBSPACE owner.

&3. is the DBSPACE name.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Do not attempt to UNLOAD the DBSPACE until tables are created in the DBSPACE.

ARI0874I *&1A.* **rows unloaded from table** *&2..&3..&4..*

Explanation: Identifies the number of rows unloaded from a table and written to the output file during UNLOAD command processing.

&1A. is the number of rows.

&2. is the server-name.

&3. is the table owner.

&4. is the table name.

System Action: Processing continues.

ARI0875I *&1A.* **rows loaded into table** *&2..&3..&4..*

Explanation: Identifies the number of rows loaded into a table.

&1A. is the number of rows.

&2. is the server-name.

&3. is the table owner.

&4. is the table name.

System Action: Processing continues.

ARI0876E **Data file record error: Record Type =** *&1A.*. **Input record count =** *&2A.*. **Reason Code =** *&3A.*.

Explanation: An error has been detected in the RELOAD TABLE or RELOAD DBSPACE input data file. The format or content of the input file is incorrect.

RECORD TYPE (&1A.) identifies the type of input record expected, or the type of record being processed when the error occurred.

Possible types are:

10 Unload header record

20 SQLDA table description record

60 Table row data record.

Input record count (&2A.) identifies the number of input file records read before the error was detected.

Reason Code (&3A.) identifies the particular error that was encountered.

The definition of the REASON CODE values are:

04 A premature end of file occurred. &1A. identifies the record type value expected when the end of file condition occurred.

08 An unexpected record type was read or the format of a record read is invalid. &1A. identifies the expected record type.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: If the input file was created by DBS UNLOAD processing, contact the system programmer.

If the input file was not created by DBS UNLOAD processing, correct the error and rerun the job.

System Programmer Response: Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0877E **Table** *&1..&2..&3.* **not created. It already exists.**

Explanation: A table with the tablename specified in the last RELOAD TABLE command displayed in the DBS Utility message listing could not be created since a table with the same owner.tablename already exists in the database.

&1. is the name of the application server.

&2. is the owner of the table.

&3. is the name of the table.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command by specifying a different tablename or the PURGE parameter and reprocess the command. (See "Rerunning DBS Utility" on page 223.)

ARI0878E **Creation of** *&1. &2..&3..&4.* **failed.**

Explanation: While processing the last RELOAD command displayed in the DBS Utility message listing, an SQL CREATE TABLE or INDEX statement executed by DBS Utility resulted in an SQL error or warning condition.

&1. is TABLE or INDEX.

&2. is the server-name.

&3. is the owner of the table or index.

&4. is the name of the table or index.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Refer to the messages preceding this message in the DBS Utility message listing to identify the SQL error or warning conditions encountered. Perform the actions necessary to correct the SQL condition. Repreprocess the RELOAD command. (See "Rerunning DBS Utility" on page 223.)

ARI0879I **Table** *&1..&2..&3.* **created.**

Explanation: Identifies that a table has been successfully created while processing the last RELOAD command displayed in the DBS Utility message listing.

&1. is the server-name.

&2. is the table owner.

&3. is the table name.

System Action: Processing continues.

ARI0880I *&1A.* **rows deleted from table** *&2..&3..&4.***.**

Explanation: Identifies the number of table rows deleted while processing the last RELOAD command displayed in the DBS Utility message listing.

&1A. is the number of rows in the table that were deleted.

&2. is the name of the server-name.

&3. is the owner of the table.

&4. is the name of the table.

System Action: Processing continues.

ARI0881E *&1. &2..&3.* **was not found.**

Explanation: The table name specified by the INTABLE or RESTARTTABLE parameter in the last RELOAD TABLE command displayed in the DBS Utility message listing was not found on the input data file.

&1. is a RELOAD parameter value (either INTABLE or RESTARTTABLE).

&2. is the table owner (if specified in either INTABLE or RESTARTTABLE).

&3. is the table name specified in the INTABLE or RESTARTTABLE parameter.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0882E **Missing the** *&1. &2***. parameter.**

Explanation: A command parameter is missing from the last command displayed in the DBS Utility message listing.

&1. is the command or command parameter identifier.

&2. is the command parameter subidentifier (optional).

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0883E **Extra command parameters.**

Explanation: The last command displayed in the DBS Utility message listing contains additional information after all valid command parameters have been specified.

The message may also occur because the command file record after a DBS Utility command does not contain a valid command identifier, or information has been specified after the terminating semicolon in an SQL statement parsed by DBS Utility processing.

Note: Only the various forms of the SQL COMMIT, ROLLBACK, and CONNECT statements are parsed by DBS Utility processing.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0884I **...Command processed.**

Explanation: The last command displayed in the DBS Utility message listing was processed.

System Action: DBS Utility processing continues.

ARI0885E **Invalid** *&1.* **value length.**

Explanation: The length of a command parameter value is invalid in the last command displayed in the DBS Utility message listing.

&1. identifies the command parameter.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0886E Incomplete *&1.* parameter.

Explanation: A command parameter is incomplete in the last command displayed in the DBS Utility message listing.

&1. identifies the command parameter.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0887E Invalid *&1. &2.* value.

Explanation: An invalid command parameter value is contained in the last command displayed in the DBS Utility message listing.

&1. is the command parameter identifier.

&2. is the command parameter subidentifier (optional).

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0888E Invalid *&1.* parameter.

Explanation: An unknown keyword parameter or an invalid parameter value was specified in the last command displayed in the DBS Utility message listing.

&1. identifies the command parameter.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0889E Second keyword in the command must be TABLE.

Explanation: The command identifier keyword TABLE is missing in the last command displayed in the DBS Utility message listing.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0890E Second keyword in the command must be either TABLE, DBSPACE, PACKAGE, or PROGRAM.

Explanation: The command identifier keyword TABLE, DBSPACE, PACKAGE or PROGRAM is missing in the last command displayed in the DBS Utility message listing.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0891E *&1. &2.* parameter is missing a *&3.*.

Explanation: A command parameter is missing a required symbol or a quoted string value is empty (quoted string value length = 0) in the last command displayed in the DBS Utility message listing.

&1. is the primary command parameter identifier.

&2. is the secondary command parameter identifier (optional).

&3. Identifies the missing symbol.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0892E *&1.* parameter(s) not found.

Explanation: A required command parameter is missing or a keyword parameter has been specified without parameter values in the last command displayed in the DBS Utility message listing.

&1. identifies the command parameter.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0893E *&1. &2.* parameter is longer than *&3A.* positions.

Explanation: A command parameter is too long in the last command displayed in the DBS Utility message listing.

&1. is the command parameter identification.

&2. is the command subparameter identification (optional).

&3A. is the maximum allowable parameter length.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0894I Isolation level *level* in effect.

Explanation: This message identifies the isolation level in effect for subsequent processing performed by the DBS Utility (*level* = CURSOR STABILITY, REPEATABLE READ, or UNCOMMITTED READ). The message is issued after a user-supplied DBS ISOL(CS), ISOL(RR), or ISOL(UR) control parameter or SET ISOLATION command is processed. The identified isolation level remains in effect until the next SET ISOLATION command is processed or DBS Utility processing ends.

System Action: DBS Utility processing continues.

ARI0895E Invalid SQLCODE detected. See following message.

Explanation: An invalid SQLCODE has been received after the execution of an SQL statement. The SQLCODE value is not within the value range -999 through +999. This is a serious DB2 Server for VM error and subsequent Database Services Utility SET ERRORMODE OFF commands will be ignored.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Contact your system programmer.

System Programmer Response: Probable DB2 Server for VM system error. Perform problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0896E Duplicate or conflicting *&1*. parameter.

Explanation: A duplicate or conflicting command parameter is specified in the last command displayed in the DBS Utility message listing.

&1. is the command parameter identification.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0897E Unknown parameter starting at or near position *&1A*..

Explanation: An invalid parameter was detected in the last command displayed in the DBS Utility message listing.

&1A. is the approximate position in the command where the incorrect parameter begins.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI0898I ...Command ignored because of severe system error.

Explanation: A SET ERRORMODE OFF or SET ERRORMODE CONTINUE command has been encountered after a severe DB2 Server for VM error has occurred during DBS Utility processing.

A severe DB2 Server for VM error is an error that does not allow further SQL processing or an error that will normally result in the failure of all subsequent SQL statements executed by DBS Utility processing. If both SQLWARN0 and SQLWARN6 are set to S after an attempt to process an SQL statement has been made, the DBS Utility assumes that a severe DB2 Server for VM system error has occurred.

System Action: No action taken. DBS Utility error mode processing continues.

ARI0899I ...Command ignored.

Explanation: A SET ERRORMODE OFF, SET ERRORMODE ON, or SET ERRORMODE CONTINUE command has been encountered, but the error mode processing status identified in the command is already in progress.

System Action: No action taken. DBS Utility processing continues.

ARI8000I ...Ignore previous error. Processing is continuing.

Explanation: Because a SET ERRORMODE CONTINUE command was supplied before the DATALOAD command, and the COMMITCOUNT parameter was specified on the DATALOAD INFILE subcommand, the error identified by the preceding ARI0862E or ARI0866E message is ignored.

System Action: DATALOAD processing continues.

ARI8001E Invalid graphic data starting at or near position *&1A*..

Explanation: If extended graphic support is in effect for the database, this message is issued after DBS Utility processing detects an SO delimiter without a terminating SI delimiter or the graphic data is not represented in paired bytes. The message refers to the last command record entered. The last command record entered is displayed immediately preceding this error message in the DBS Utility message file.

&1A. is the approximate position in the command record where the invalid graphic data begins.

System Action: Command processing is ended.

Programmer Response: Correct input command and rerun.

ARI8002I Blocked INSERT processing was suppressed for *&1..&2..&3*.

Explanation: Insert blocking was suppressed during DATALOAD processing for the table or view identified.

&1. is the server-name.

&2. is the owner.

&3. is the name of the table or view.

System Action: Processing continues.

ARI8003I ...Extended DBCS (DBCS=YES) processing now in effect.

Explanation: This message is issued when extended DBCS processing is in effect for the database being accessed by DBS Utility processing. The DBS Utility determines when extended DBCS processing is in effect for a database by accessing the database SYSTEM.SYSOPTIONS catalog table. If a row with the SQLOPTION column value equal to DBCS and the VALUE column value equal to YES exists, extended DBCS processing is in effect for the database.

System Action: DBS Utility processing continues WITH EXTENDED DBCS processing in effect.

ARI8004I User *userid* connected to application server *server-name*.

Explanation: This message shows the current user and the current application server accessed as a result of the last implicit or explicit CONNECT command.

System Action: No action taken.

ARI8005I One or more arithmetic exception errors occurred. Exception error *&1*. occurred during *&2*. operation on *&3*. data, position *&4*..

Explanation: This message is issued whenever an arithmetic operation on numeric data causes an arithmetic exception error. The cause may be a division by zero or overflow (exceeds maximum) during the processing of a SELECT statement or a DATAUNLOAD command. The data from this error is denoted by number signs (# # #) across the width of columns of the queried table:

&1. is the type of exception error.
&2. is the arithmetic operation on data.
&3. is the data type.
&4. is position N, where N is an item in the select-list, if applicable. Position 0 is returned if an exception error occurs in the WHERE clause of a SELECT statement.

Note: The message describes the first arithmetic exception error in the queried table.

System Action: Processing continues.

ARI8006E An arithmetic error occurred during unloading of data into a numeric field. Exception error *&1*. occurred during *&2*. operation on *&3*. data, position *&4*..

Explanation: This message is issued whenever an arithmetic exception error (indicated by number signs # # #) is being written into a numeric datatype field of an output file defined for DATAUNLOAD processing.

The user is informed about the first occurrence of an arithmetic exception error caused by the SELECT statement while processing the DATAUNLOAD command:

&1. is the type of exception error.
&2. is the arithmetic operation on data.
&3. is the data type.
&4. is position N, where N is an item in a select-list, if applicable. Position 0 is returned if an exception error occurs in the WHERE clause of a SELECT statement.

System Action: DATAUNLOAD command processing is ended. Also see "Common Error Action" on page 223.

Programmer Response: Correct the arithmetic error or change the numeric data type to CHARacter data type in the output file.

ARI8007I Error(s) occurred during ERRORMODE CONTINUE processing.

Explanation: Some commands processed while SET ERRORMODE CONTINUE was in effect were not processed successfully.

System Action: DBS Utility end of program processing continues.

Programmer Response: Refer to previous messages during ERRORMODE CONTINUE processing in the DBS Utility message listing to identify any unexpected errors.

ARI8009E *&1.* **PACKAGE [***&2.&3.***]** *failed.*

Explanation:

- &1. = UNLOAD or RELOAD
- &2. = IN DATABASE
- &3. = dbname

The UNLOAD or RELOAD PACKAGE failed. If you specified an UNLOAD PACKAGE FROM clause or RELOAD PACKAGE TO clause, &2 and &3 indicate the database where the failure occurred.

System Action: Command processing ends. If you specified a RELOAD PACKAGE TO clause, then processing continues on subsequent databases.

Programmer Response: Refer to the messages preceding this message in the DBS Utility message listing to identify the SQL error conditions encountered. Reprocess the command for the failed cases. (See "Rerunning DBS Utility" on page 223.)

ARI8010E **Package name** *owner.package-name* **is not the same as the package name in the input file.**

Explanation: The package-name in the RELOAD PACKAGE command does not match the package-name in INFILE.

System Action: Command processing is ended. Also see "Common Error Action" on page 223 for more information.

Programmer Response: Determine the correct package name, alter the command, and rerun the package.

ARI8011I **Package** *&1..&2.* **was** *&3.* **[***&4. &5.***].**

Explanation:

- *&1.* = owner
- *&2.* = package name
- *&3.* = either **created** or **replaced**
- *&4.* = **in server**
- *&5.* = the application server name

Database Services Utility created or replaced a package. If an UNLOAD PACKAGE FROM clause or RELOAD PACKAGE TO clause is specified, &4. and &5. indicates the application server where this occurred.

System Action: Processing continues.

ARI8012E **Package name** *&1..&2.* **not found** *&3. &4..*

Explanation:

- &1. = owner
- &2. = package-name
- &3. = **in server**
- &4. = application server name

The identified package was not found in SYSTEM.SYSACCESS. If you specified an UNLOAD PACKAGE FROM clause, &3. and &4. indicate the application server where the package was not found.

Programmer Response: Ensure the package-name specified in the command is correct. If incorrect, correct the command, and reprocess it. (See "Rerunning DBS Utility" on page 223.) If correct, the package must be created before it can be processed by an UNLOAD PACKAGE or a REBIND PACKAGE command.

ARI8013I **Duplicate server name** *server name* **found in the TO clause.**

Explanation: The identified application server found in the TO clause of a RELOAD PACKAGE command occurred more than once.

System Action: Command processing continues. A package is reloaded to that application server only once.

ARI8014I **RELOAD PACKAGE processing is terminated. ERRORMODE CONTINUE is not in effect.**

Explanation: An error occurred during the reloading of the package into one of the application servers. The TO clause of the RELOAD PACKAGE command specifies the application server. Because ERRORMODE CONTINUE was not in effect, Database Services Utility ignores the remainder of the application server names in the list and processing of this command ends.

System Action: Command processing is ended. Also see "Common Error Action" on page 223 for more information.

Programmer Response: Refer to the messages preceding this message in the DBS Utility message listing to identify the SQL error conditions encountered. Reprocess the command for the failed cases. (See "Rerunning DBS Utility" on page 223.)

ARI8015E *command* **failed.**

Explanation: Indexes are re-created and PRIMARY KEYS, FOREIGN KEYS, and UNIQUE CONSTRAINTS are reactivated, after reloading the data. The attempt to re-establish entity or Referential Integrity has failed. The *command* indicates the first command that failed.

System Action: Processing of the DBS Utility command has ended.

User Response: Correct the data and reissue the RELOAD, or manually establish the missing indices and keys.

ARI8016I *command* **successful.**

Explanation: Indexes are re-created and PRIMARY KEYS, FOREIGN KEYS, and UNIQUE CONSTRAINTS reactivated, after the data is reloaded. A step in this process has succeeded.

System Action: Processing of the DBS Utility command continues.

ARI8017E Second keyword in command must be PACKAGE.

Explanation: The command identifier keyword PACKAGE is missing from the last command displayed in the Database Services Utility message listing.

System Action: Command processing ends. See "Common Error Action" on page 223 for more information.

System Programmer Response: Correct the command and reprocess it. See "Rerunning DBS Utility" on page 223 for more information.

ARI8018E Command is not supported for primary or unique keys.

Explanation: Using the Database Services Utility REORGANIZE INDEX command for an index that the database manager created to enforce the uniqueness of a primary or unique key is invalid. Use the following commands instead:

- ALTER TABLE ACTIVATE PRIMARY KEY for primary keys.
- ALTER TABLE ACTIVATE UNIQUE KEY for unique keys.

System Action: The Database Services Utility ignores the command.

ARI8020E Invalid CREATE SCHEMA statement.

Explanation: You must make the first (and only the first) statement within the schema sequential input file the CREATE SCHEMA statement.

System Action: Command processing ends.

User Response: Correct the command and reprocess it. If a COMMIT of database changes occurred during this execution of the SCHEMA command, rerun the Database Services Utility for this command, but omit all the statements within the sequential input file that altered the database prior to the last COMMIT of database changes.

ARI8021E Statement not supported within a Schema.

Explanation: The last statement displayed in the Database Services Utility message listing is not a valid schema statement.

System Action: Processing of this statement ends.

User Response: Correct the command and reprocess it. If a COMMIT of database changes occurred during this execution of the SCHEMA command, rerun the Database Services Utility for this command, but omit all the statements within the sequential input file that altered the database prior to the last COMMIT of database changes.

ARI8022E Second keyword in command must be INDEX.

Explanation: The command identifier keyword INDEX is missing in the last command displayed in the Database Services Utility message listing.

System Action: Command processing ends. See "Common Error Action" on page 223 for more information.

User Response: Correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.)

ARI8023E User *userid1* **is not authorized to create a Schema for another user** *userid2***.**

Explanation: The AUTHORIZATION user ID **userid2** on the CREATE SCHEMA statement must be the same as the currently connected user *userid1*.

System Action: Command processing ends.

User Response: Correct the command and reprocess it.

ARI8024E The application server does not support package creation.

Explanation: The package cannot be bound against an application server prior to SQL/DS Version 3 Release 1.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Obtain the portable access module for this application from an SQL/DS Version 2 Release 2 application server and try again.

ARI8025E The package cannot be bound with the DRDA protocol. Reason Code = *reason code***.**

Explanation: A RELOAD PACKAGE processing error occurred. The package input data file contains a

package which has the characteristic shown by the *reason code*.

The *reason code* can have a value of:

1 The package was created under SQL/DS Version 2 Release 2.
2 The package is a modifiable package created using extended dynamic statements.

System Action: Command processing ends. See "Common Error Action" on page 223 for more information.

Programmer Response: Ensure the package is not created under SQL/DS Version 2 Release 2 and that it is not a modifiable package created using extended dynamic statements.

ARI8026E Error binding the following SQL statements: *stmt-name*.

Explanation: Binding of the SQL statement beginning with *stmt-name* was not successful. This message is followed by other messages which indicate the condition that caused the error message.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Take action based on the condition that caused the error message.

Note: "Common Error Action" on page 223 refers to the section in the manual.

ARI8027E Infile contents invalid.

Explanation: An error has been detected in the RELOAD PACKAGE input data file. In the input file, the format or content or both are incorrect.

System Action: Command processing is ended. See "Common Error Action" on page 223 for more information.

Programmer Response: Contact the system programmer. The system programmer should perform problem determination. If a system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI8028I Schema processing terminated - ERRORMODE CONTINUE not in effect.

Explanation: An error occurred when the database manager processed the SCHEMA command. Because ERRORMODE CONTINUE was not in effect, the database manager ignores the rest of the statements in the sequential input file, and stops processing the SCHEMA command.

System Action: Command processing ends.

User Response: Correct the command and reprocess it. If you committed changes to the database during this execution of the SCHEMA command, run the Database Services Utility for this command again, but omit all the statements within the sequential input file that have already altered the database.

ARI8029E Command is not supported for *&1*.

Explanation: The DBSU command is not supported when DBSU is connected to a non-DB2 Server for VM application server or to an earlier release of an SQL/DS application server. Parameter *&1* is:
- A non-DB2 Server for VM application server
- An SQL/DS application server, Version 3 Release 1 or earlier.

System Action: The command is ignored.

ARI8030E The Extract Facility is no longer supported.

Explanation: The database manager no longer supports the Extract Facility INMOD subcommand of the DATALOAD command.

System Action: DATALOAD TABLE processing ends.

ARI8031E The package *&1.* cannot be reloaded to a back level release server.

Explanation: The package cannot be reloaded to the target application server because of package format differences. This message will be issued when reloading an SQL/DS Version 3 Release 1 or above package to a pre-Version 3 Release 1 database. The RELOAD PACKAGE command failed.

&1. is the name of the package.

System Action: The package is not reloaded.

User Response: Issue ROLLBACK to the logical unit of work.

ARI8032I The length of the SQL CREATE TABLE statement for table *&1..&2*. exceeds 8192 bytes. It will not be possible to reload with the parameter NEW.

Explanation: The CREATE TABLE statement of the identified table exceeds 8192 bytes. The CREATE TABLE statement that the DBS Utility places into the unload output file may be incomplete, because of truncation.

To reload the data from this unload output file, RELOAD with the PURGE parameter must be used, and the table or tables must already exist in the database.

&1. is the owner of the table.

&2. is the name of the table.

System Action: Processing continues.

ARI8033E **DBS-initiated SQL CREATE TABLE statement for table *&1..&2*. exceeds 8192 bytes. RELOAD with the PURGE parameter should be used.**

Explanation: During reload with the parameter NEW, the CREATE TABLE statement built by the DBS utility for use in creating the table exceeds the 8192 byte maximum SQL statement length allowed by the database manager, and the table cannot be created by the DBS utility.

&1. is the owner of the table.

&2. is the name of the table.

System Action: Command processing is ended.

Programmer Response: To reload the data, use RELOAD with the PURGE parameter. The table must already exist.

ARI8034I *command*

Explanation: The DBS Utility is processing a RELOAD command that includes the COMMITCOUNT parameter, but neither the RESTARTCOUNT or RESTARTTABLE parameters. RELOAD processing will DROP all indexes and DEACTIVATE all primary and unique keys before reloading data. The COMMITCOUNT parameter indicates that these DROP statements will be committed to the database before the RELOAD process has completed. If the RELOAD process does not complete successfully, the dropped objects can no longer be recovered. To aid the user in recovering these objects, the ARI8034I messages provide the CREATE statements required to recover the dropped objects. If no error occurs reloading the data, all dropped objects will be recreated during normal RELOAD processing. Refer to ARI8035I to identify the clustering index.

System Action: Processing continues.

Programmer Response: If an error occurs during RELOAD processing, refer to the ARI8034I messages to create or activate the dropped and deactivated objects.

ARI8035I **The clustering index for** *table-creator*.*table-name* **is** *index-type index-owner*.*index-name*

Explanation: This message identifies the INDEX, PRIMARY KEY or UNIQUE KEY that is currently the clustering index for the table.

System Action: Processing continues.

Programmer Response: When recovering from a failed RELOAD, this object should be the first to be created or activated so that it will be used as the clustering index.

ARI8700I **Enter: Command continuation, null line to run, or QUIT.**

Explanation: When the DBS Utility control file (SYSIN) is assigned to the terminal, this message is written to the terminal when the user enters a null line (presses the ENTER key) instead of entering the information necessary to complete the current command.

System Action: Waits for the next terminal entry by the user.

User Response: Enter the remainder of the current command or a null line (press ENTER key) to process the current command. To continue to the next command without processing the current command, enter QUIT or HX.

ARI8701I **Enter: Command string continuation followed by a(n) *&1.*.**

Explanation: When the DBS Utility control file (SYSIN) is assigned to the terminal, this message is written to the terminal after the user entered a null line instead of command continuation information. It identifies that a delimited string within the command is being continued and that the user must complete the delimited string before entering other command information or ending the command.

&1. is the delimiter required to terminate the string.

System Action: Waits for the next terminal entry by the user.

User Response: Enter the remainder of the delimited string. End the delimited string with the delimiter identified by &1. and complete the command entry.

ARI8702I **Enter: Command continuation or QUIT.**

Explanation: When the DBS Utility control file (SYSIN) is assigned to the terminal, this message is written to the terminal after the user entered a null line instead of the next DATAUNLOAD or DATALOAD subcommand.

System Action: Waits for the next terminal entry by the user.

User Response: Enter the next DATAUNLOAD or DATALOAD subcommand. To continue to the next command without processing the current command, enter QUIT or HX.

ARI8703I **Enter: Next data record or ENDDATA.**

Explanation: When the DBS Utility control file (SYSIN) is assigned to the terminal and the DBS Utility DATALOAD command is reading input data records from the control file, this message is written to the terminal after the user has entered a null line instead of the next data record. It identifies that the user can enter

the next data record or ENDDATA to indicate that there are no more data records.

System Action: Waits for the next terminal entry by the user.

User Response: Enter the next data record or ENDDATA from the terminal.

ARI8980I **Statistics were generated automatically for table** *&1..&2.*.

Explanation: The database manager generated table or DBSPACE statistics while the Database Services Utility loaded the rows. The Database Services Utility does not need an SQL UPDATE STATISTICS request, and one is not issued during Database Services Utility command processing.

&1. is the owner of the table or DBSPACE.

&2. is the name of the table or DBSPACE.

System Action: Database Services Utility processing continues.

ARI8981I **Dynamic statistic accumulation was disallowed for table** *&1. &2.* **Reason Code =** *&3.*.

Explanation: The database manager does not allow the automatic accumulation of table statistics while loading data for this DATALOAD command.

&1. is the owner of the table.

&2. is the name of the table.

The reason code identifies why the automatic accumulation of table statistics was disallowed:

01 One or more indexes exist on the table.

02 The ROWCOUNT in the SYSCATALOG entry for this table was greater than zero, or was -2 (indicating that a failed DATALOAD was restarted).

System Action: Database Services Utility processing continues. The Database Services Utility issues an SQL UPDATE STATISTICS request to generate the statistics.

ARI8982I **Statistics for loaded rows replace statistics for table** *&1..&2.*.

Explanation: The Database Services Utility processed a DATALOAD command, and the ROWCOUNT value in the SYSCATALOG entry for the loaded table was not greater than zero. As a result, the database manager automatically generated statistics for the loaded rows, and replaced the table's current statistics.

&1. is the owner of the table.

&2. is the name of the table.

If the table was not empty prior to the DATALOAD, the generated statistics do not accurately reflect the table's contents. This prevents the DB2 Server for VM Optimizer from selecting the most efficient access path when retrieving rows from the table.

System Action: Database Services Utility processing continues. The Database Services Utility issues an SQL UPDATE STATISTICS request to generate statistics for each table.

ARI8983I **Dynamic statistic accumulation was disallowed for all tables named in the DATALOAD command.**

Explanation: A DATALOAD command that loaded data into more than one table was processed. For this DATALOAD, the database manager cannot accumulate statistics while the Database Services Utility loads the rows. The database manager does not support this form of statistics collection for the concurrent loading of more than one table.

System Action: Database Services Utility processing continues. The Database Services Utility issues an SQL UPDATE STATISTICS request to generate the statistics for each table.

ARI8990I **The table** *&1..&2..&3.* **is in a nonrecoverable storage pool.**

Explanation: This message identifies each table in the nonrecoverable storage pool that will be loaded by DATALOAD processing.

&1. is the name of the application server.

&2. is the owner of the table.

&3. is the name of the table.

System Action: Processing continues.

ARI8991I **If the database manager is executing with LOGMODE=A|Y, any rows inserted into the database will not be deleted by ROLLBACK processing.**

Explanation: This message is issued before Database Services Utility RELOAD PURGE insert processing that will update a nonrecoverable storage pool is started. If a subsequent error occurs during DBS RELOAD PURGE processing, the table or tables may be partially loaded, because ROLLBACK processing will not remove the rows inserted into the database.

If running in single user mode with LOGMODE=N, ROLLBACK processing will remove the inserted rows.

If running in multiple user mode, or in single user mode with LOGMODE=A or Y, ROLLBACK processing will not remove the inserted rows; it will commit the rows to the database.

System Action: Processing continues.

Programmer Response: If RELOAD PURGE processing fails to complete successfully, rerun the DBS Utility to reload the tables.

ARI8992I **If the database manager is executing with LOGMODE=A|Y, any rows inserted into the table** *[server-name.]owner.table-name* **will not be deleted by ROLLBACK processing.**

Explanation: An error occurred during Database Services Utility DATALOAD processing, and rows were inserted into a table in a nonrecoverable storage pool. This message identifies that subsequent ROLLBACK processing may not remove the inserted rows.

If running in multiple user mode, or in single user mode with LOGMODE=A or Y, ROLLBACK processing will not remove the inserted rows. In fact, the rows will be committed to the database by a ROLLBACK.

If running in single user mode with LOGMODE=N, ROLLBACK processing will remove the inserted rows.

System Action: Processing continues.

Programmer Response: If running in multiple user mode, or in single user mode with LOGMODE=A or Y, perform the recovery actions required for a nonrecoverable storage pool before rerunning the DBS utility.

ARI8993I **If the database manager is executing with LOGMODE=A|Y, you should not use DATALOAD RESTART processing before the last** *&1A.* **rows inserted into the table** *&2..&3..&4.* **are deleted.**

Explanation: An error occurred during Database Services Utility DATALOAD processing, and rows were inserted into a table in a nonrecoverable storage pool. This message identifies that DATALOAD RESTART processing may not be applicable until special action is taken.

&1A. is the number of rows inserted into the table since the start of DATALOAD processing or since the last ARI0800I message was displayed.

&2. is the name of the application server.

&3. is the owner of the table.

&4. is the name of the table.

If running in multiple user mode, or in single user mode with LOGMODE=A or Y, ROLLBACK processing will not remove the rows inserted since the start of DATALOAD processing or since the last ARI800I message was generated. Special action is required before rerunning the DBS Utility to load the identified tables.

If running in single user mode with LOGMODE=N, ROLLBACK processing will remove the inserted rows.

System Action: Processing continues.

Programmer Response: If running in single user mode with LOGMODE=N, the DBS Utility can be rerun to load the identified tables without taking any special action.

If running in multiple user mode, or in single user mode with LOGMODE=A or Y, ROLLBACK processing will commit (not remove) the rows inserted since the last ARI0800I message was displayed. You must perform the recovery actions required for a nonrecoverable storage pool before rerunning the DBS Utility.

ARI8994I **Blocked FETCH or INSERT processing was suppressed.**

Explanation: This message identifies that blocked fetch or insert processing was suppressed by the database manager while processing the previous command displayed in the DBS Utility message file. Possible causes include:

- The DBS Utility is running in single user mode.
- A table or view with a LONG VARCHAR or LONG VARGRAPHIC column was referenced by the command.
- There was insufficient virtual storage to support blocked fetch or insert processing.
- Two or more rows of the fetched or inserted data will not fit into an 8K block.

System Action: DBS Utility processing continues.

ARI8995I *parm1* **continuing. Record Count =** *parm2***.**

Explanation: This message is written to the operator console or user terminal after every 15000 input or output data file records are processed by a DBS Utility command.

parm1 is either DATALOAD, DATAUNLOAD, RELOAD, or UNLOAD.

parm2 value is either:
- The current data file record count (*parm1* is DATALOAD or RELOAD)
- The current output data file record count (*parm1* is DATAUNLOAD or UNLOAD).

Note: During DATALOAD command processing, the message is written only if the DBS Utility message file is not assigned to:
- A console or terminal, and
- DATALOAD COMMITCOUNT processing is not in progress.

System Action: DBS Utility processing continues.

ARI8996I ...Begin UPDATE STATISTICS for *&1..&2..&3..*

Explanation: This message is written to the DBS Utility message file before an SQL UPDATE STATISTICS statement is issued for a table during DBS Utility command processing.

&1. is the name of the application server.

&2. is the owner of the table.

&3. is the name of the table.

System Action: DBS Utility processing continues.

ARI8997I ...Begin COMMIT [RELEASE] processing.

Explanation: This message is written to the DBS Utility message file before an SQL COMMIT or SQL COMMIT RELEASE statement is issued. The successful completion of the COMMIT will be identified by the subsequent message:

```
ARI0811I ...COMMIT of any database
           changes successful.
```

System Action: DBS Utility processing continues.

ARI8998I ...Begin ROLLBACK [RELEASE] processing.

Explanation: This message is written to the DBS Utility message file before a ROLLBACK or ROLLBACK RELEASE statement is issued by DBS Utility command processing. Successful completion of the ROLLBACK will be identified by the subsequent message:

```
 ARI0811I ...ROLLBACK of any database
            changes successful.
```

System Action: DBS Utility processing continues.

ARI8999E Invalid SQL statement or parameter: *&1.*.

Explanation: An invalid control parameter or SQL statement was encountered in the DBS Utility invocation parameter list.

&1. is the first 20 positions of the invalid parameter string; or the phrase INVALID LENGTH OR CALL TYPE.

System Action: DBS Utility processing completes.

Programmer Response: If the phrase INVALID LENGTH OR CALL TYPE is displayed for variable *&1.*, correct the user program. The program specified an invalid DBS Utility call type or an invalid parameter string length.

When the variable &1. contains a portion of the DBS Utility parameter string, correct the parameter string. After the appropriate corrections are made, rerun.

Chapter 15. ARI0900–ARI0949 Database Maintenance and Database I/O Messages

ARI0900E operation **error on DDNAME =** xxxxxxxx**. Return Code =** nn**. EREASON =** nn**.**

Explanation: An error occurred in a sequential input or output operation on the *xxxxxxxx* file during database generation, ADD DBSPACE processing, or ADD and DELETE DBEXTENT processing.

The Return Code value is the Primary Error Code value. The EREASON value is the Secondary Error Code value. For further explanation, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: For corrective action, look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

ARI0901E **End of file on SYSIN before DBGEN process is completed.**

Explanation: End of file was encountered on SYSIN before all required control information was received by the database generation process. Either:

- The CUREXTNT keyword was not specified, or
- The number of DBEXTENT control statements was not the same as specified by the CUREXTNT keyword control statement, or
- There were no DBSPACE control statements.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Make the appropriate correction and rerun the program.

ARI0903E **DBGEN keyword CUREXTNT was not specified.**

Explanation: The required CUREXTNT keyword control statement was not specified as input to the database generation process.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Supply the CUREXTNT keyword control statement and rerun the program.

ARI0904E **Value specified for DBGEN keyword** keyword **is invalid.**

Explanation: The value specified for keyword was either null or exceeded eight characters in length.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Correct the value specified for the keyword and rerun the program.

ARI0905E **Invalid keyword specified.**

Explanation: You did not specify a valid keyword for database generation or an ADD and DELETE DBEXTENT function.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Correct the unrecognizable keyword and rerun the program.

ARI0906E **Value for** keyword **exceeds maximum of** nnnnnnn**.**

Explanation: The value specified for *keyword* exceeded the maximum allowed value *nnnnnnn*.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Correct the value specified and rerun the program.

ARI0907E **Incorrect value specified on DBSPACE definition. Return Code =** n**.**

Explanation: A value specified on a DBSPACE control statement was considered incorrect (not a valid type or number). Values of *n* can be:

1 Unrecognized type of DBSPACE specified (not PUBLIC, PRIVATE, or INTERNAL).

2 Unrecognizable number of DBSPACEs specified on the INTERNAL DBSPACE control statement.

© Copyright IBM Corp. 1987, 2000

3 Unrecognizable number of pages specified for a PUBLIC, PRIVATE, or INTERNAL DBSPACE statement (nonnumeric value or value less than 1).

4 During database generation, one of the first two PUBLIC DBSPACEs defined was allocated to a nonrecoverable storage pool. These are the catalog and package DBSPACEs and cannot be assigned to a nonrecoverable storage pool. Nonrecoverable storage pools are defined by the POOL control statement.

5 During database generation, the INTERNAL DBSPACE control statement was not immediately followed by an END delimiter control statement. You may have misspelled the END delimiter control statement.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Make the appropriate change to the DBSPACE control statement and rerun the program.

ARI0908E Insufficient DBEXTENT information.

Explanation: An END delimiter control statement was encountered before the number of DBEXTENT control statements specified by the CUREXTNT keyword were processed.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Ensure that the number of DBEXTENT control statements is the same as was specified by the CUREXTNT keyword and rerun the program.

ARI0909E DBEXTENT number sequence error.

Explanation: If database generation is being performed, one of the following errors occurred:

- The DBEXTENT control statements were not ordered so that the DBEXTENT numbers were in the required sequence. The required sequence is 1, 2, ... CUREXTNT value. The CUREXTNT value is specified on the CUREXTNT control statement.
- The DBEXTENT number on a DBEXTENT control statement was entered incorrectly or was omitted.
- The POOL keyword on the POOL control statement was misspelled. The POOL control statement was assumed to be a DBEXTENT control statement (and the misspelled word was assumed to be the DBEXTENT number).

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Ensure that the DBEXTENT control statements are correctly specified and are in the proper sequence. Ensure that a POOL control statement was not misspelled. Rerun the program.

ARI0910E Incorrect storage pool specified. Return Code = n.

Explanation: A specified storage pool number or a DBEXTENT or DBSPACE control statement was not valid for the reason signified by the value of *n*:

1 A nonnumeric character was encountered.

2 A value less than one or greater than MAXPOOLS was specified.

3 A DBSPACE control statement specified a storage pool number for which no DBEXTENTs were assigned.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Correct the value in error and rerun the program.

ARI0911E Maximum number of DBSPACES exceeded.

Explanation: The number of DBSPACEs being defined exceeded the value specified by the MAXDBSPC keyword.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Increase the value specified on the MAXDBSPC keyword or reduce the number of DBSPACE control statements and rerun the program.

ARI0912E Insufficient space in the application server directory.

Explanation: There is not enough space in the application server directory to map the number of DBSPACEs defined. You can use the output of the SHOW DBCONFIG operator command to estimate the amount of directory space available for adding DBSPACEs.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Reduce either the number or the size of the DBSPACEs being defined and

rerun the program, or use the expand directory function to increase the size of the page map table to allow more DBSPACE pages to be added.

ARI0913E Maximum number of pages exceeded.

Explanation: When formatting the DBEXTENTs the available space defined across all DBEXTENTs exceeded the maximum number of pages.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Reduce the space defined across all DBEXTENTs and rerun the program, or use the expand directory function to increase the size of the allocation bitmaps and page map table to allow more DBEXTENT and DBSPACE pages to be added.

ARI0914E System checkpoint failed.

Explanation: System checkpoint failed while attempting to add DBSPACEs to database. This may be due to a cancelled database or LOG ARCHIVE or an error found in the directory.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Perform problem determination. Make a record of what went wrong and contact the designated support group for your installation.

ARI0915I One or more DBSPACES added to database.

Explanation: The DBSPACEs specified in the SYSIN file have been added to the database by the Database Storage System (DBSS) component. Any errors occurring after this message are in the catalog update phase of the ADD DBSPACE function.

System Action: The process continues with the catalog update.

ARI0916E Number of internal DBSPACES is zero, or number of pages specified is zero.

Explanation: One of the following conditions occurred:

- No INTERNAL DBSPACE control statement was supplied
- The number of INTERNAL DBSPACEs on the INTERNAL DBSPACE control statement was specified as zero during database generation or the ADD DBSPACE process.
- The number of pages specified on the INTERNAL DBSPACE control statement was zero.

In any case, a cancel request is issued and the database changes are not made.

System Action: Processing is terminated.

Operator Response: Return the console output with the output listing.

System Programmer Response: Rerun the program, specifying the INTERNAL DBSPACE control statement (with more than zero DBSPACEs) and a correct number of pages.

ARI0917E Maximum number of DBEXTENTS exceeded.

Explanation: You attempted to add more DBEXTENTs than the maximum number specified at database generation time. The MAXEXTNTS database generation control statement defines the maximum number of DBEXTENTs that can be added to the database. The SHOW DBCONFIG operator commands displays this value.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: If more DBEXTENTs are required, you will need to do another data base generation and specify a larger value for MAXEXTNTS.

ARI0918I No input to ADD/DELETE DBEXTENT.

Explanation: You invoked the ADD or DELETE DBEXTENT function, but the input file did not supply any ADD or DELETE control statements. POOL or ARCHIVE control statements might not be in the input file.

System Action: The program ends normally.

Operator Response: Return the console output with the output listing.

System Programmer Response: Provide the desired respective ADD and DELETE control statements (and optionally, POOL and ARCHIVE control statements) in the input file, and rerun the program.

ARI0919D Database generation invoked. The database will be formatted and the original database destroyed. Enter either: DBGEN to continue, or CANCEL to cancel.

Explanation: The application server was started with the SQLDBINS or SQLDBGEN EXEC. This invokes the database generation process, which will format the directory, DBEXTENTs, and logs of the database.

System Action: The system waits for the operator's reply.

If the reply is DBGEN, the database generation process continues.

If the reply is CANCEL, the application server is ended without modification to the database.

Any other reply causes the message to be reissued.

Operator Response: If database generation is to be performed, reply DBGEN.

If you do not want database generation to be performed, reply CANCEL.

System Programmer Response: Inform the operator of your intentions when running database generation.

ARI0920D Cold-log function invoked. Logs will be formatted and existing log data destroyed.

Enter either: COLDLOG to continue, or CANCEL to cancel.

Explanation: The application server was started with the SQLLOG EXEC. This invokes the COLDLOG function, which will format the database logs.

System Action: The system waits for the operator's reply.

If the reply is COLDLOG, message ARI0944D is issued to determine if the DB2 Server for VM logs are to be reformatted or reconfigured.

If the reply is CANCEL, the application server is ended without modification to the log or logs.

Any other reply causes this message to be reissued.

Operator Response:

If COLDLOG (log reformatting or reconfiguring) is to be performed, reply COLDLOG.

If you do not want to perform COLDLOG (log reformatting or reconfiguring), reply CANCEL to end the application server.

System Programmer Response: Inform the operator of your intentions when running COLDLOG.

ARI0921I No input to add DBSPACE.

Explanation: The ADD DBSPACE function was invoked, but no control information was provided in the SYSIN file.

System Action: The program ends normally.

Operator Response: Return the console output with the output listing.

System Programmer Response: If the intent was to add DBSPACEs, provide the necessary control information in the SYSIN file and rerun the program.

ARI0922I DBEXTENT(s) have been added to or deleted from the database.

Explanation: Adding or deleting DBEXTENTs to or from the database has been completed successfully.

System Action: The program continues.

ARI0923E Space allocation for filename **is less than minimum.**

Explanation: The space allocated for the DBEXTENT or log minidisk identified by *filename* is less than the minimum allowed by DB2 Server for VM storage management.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Increase the space allocation for the DBEXTENT or log minidisk identified by file name and rerun the program. For the minimum space allocation for the device type on which this DBEXTENT or log is defined, refer to the *DB2 Server for VM System Administration* manual.

ARI0924I It is recommended that a database archive be taken.

Explanation: The directory expansion process completed successfully. It is strongly recommended that a database archive be taken immediately. Any updates made to the database after the directory expansion may not be applied correctly during a restore if a database archive taken before the directory expansion is used as the starting point of the restore set.

System Action: Processing ends.

ARI0925E An error occurred during {read|write} I/O operation on DASD cuu**. DDNAME =** ddname**. Return Code =** nnn**.**

Explanation: An error occurred while attempting to read or write a block of an DB2 Server for VM database minidisk using the CP IUCV SEND function.

The cuu value is the virtual device address of the database minidisk.

The DDNAME value identifies the database minidisk:

BDISK Directory minidisk

LOGDSK1
First log minidisk

LOGDSK2
Second log minidisk

DDSKn
DBEXTENT n of the database; n is a 1- to 3-digit number.

The Return Code value defines the error condition, as follows:

- Return Code = 1: This value indicates that the database manager has attempted to access a minidisk block address that is not valid (beyond the end of the minidisk). The value is from field IPRMMSG1 in the IUCV REPLY External Interrupt Buffer. This error can occur because of a user error in the following cases:
  - The Directory minidisk (BDISK) was replaced with a smaller minidisk. When restoring the database from the archive file, this error will occur as a write error.
  - A DBEXTENT minidisk (DDSKn) was replaced with a smaller minidisk. When restoring the database from the archive file, this error may occur. It can also occur after a successful restore from archive as a write error (the time period may be short or long).
  - If you restore your database with an archive file created from another DB2 Server for VM database, this error can occur (as a write error) on the Directory minidisk (BDISK) or any DBEXTENT (DDSKn) during restore. It can also occur later on any DBEXTENT. It occurs because the actual minidisk is smaller than the archived minidisk.

  For all other cases (including all read error cases), this is a DB2 Server for VM system error.
- Return Code = 2 through Return Code = 7: These are values from field IPRMMSG1 in the IUCV REPLY External Interrupt Buffer. A Return Code other than 5 indicates that a DB2 Server for VM or VM system error occurred. Return Code = 5 indicates an uncorrectable I/O error. For this situation, a CP I/O error message (DMK prefix) describing the error is displayed on the system operator (CP) console. This message identifies the database virtual machine and the virtual device address.
- Return Code = 9: The device has been reset by a CP RESET command (from either the virtual machine operator or the system operator). The displayed value is the first byte of the field IPUSER in the IUCV SEVER External Interrupt Buffer.
- Return Code = 57 or Return Code = 58: A DB2 Server for VM or VM system error occurred. The displayed value is obtained from the first byte of the field IPUSER in the IUCV SEVER External Interrupt Buffer. The database manager has added 50 to the value. (For example, if IPUSER = 07, Return Code = 57.)
- Return Code = 101 through Return Code = 121: A DB2 Server for VM or VM system error occurred. The displayed value is obtained from the CP IUCV SEND function return code. The database manager has added 100 to the value. (For example, if SEND return code = 01, Return Code = 101.)

System Action: Unless the error is on a log minidisk and you are using dual logging, the application server terminates. For dual logging, the application server terminates only if errors occur on the same relative block on both log minidisks. For termination, a virtual machine dump occurs to assist in problem determination for all error conditions except uncorrectable I/O errors and device reset.

Operator Response: Save the error message and the virtual machine dump (if any), and notify your system programmer.

System Programmer Response: Take action based on the Return Code value:

- Return Code = 1: If the error was caused by a database minidisk that is too small, you must replace the minidisk. Refer to the *DB2 Server for VM System Administration* manual for information about how to replace a database minidisk.

  Otherwise, determine the service levels of VM and the DB2 Server for VM product. Make a record of what occurred and contact the designated support group for your installation.
- Return Code = 5: Contact the system operator to obtain the cause of the I/O error. Operator action or hardware service may be required to remove the cause of the error. If you have an unrecoverable DASD media error, refer to the *DB2 Server for VM System Administration* manual for information about how to replace a database minidisk.
- Return Code = 9: Request that the virtual machine operator and system operator not enter CP RESET commands for database minidisks.
- All other Return Code values: Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated support group for your installation.

Unless an unrecoverable DASD media error occurred, restart the application server (possibly after the DASD is repaired). If the error occurred during an Add DBEXTENT or Add DBSPACE operation, refer to the *DB2 Server for VM System Administration* manual for information on recovery from system failures. If the error occurred during a database generation, restart the process from the beginning (using the SQLDBINS EXEC or the SQLDBGEN EXEC).

ARI0926E The maximum number of minidisks specified for the database exceeds the limit for the Virtual Machine.

Explanation: The number of minidisks specified for the database is greater than the number of IUCV connections allowed for the virtual machine. Each database minidisk requires one IUCV connection. In addition, each user virtual machine which can be concurrently connected to the database virtual machine requires one IUCV connection. The MAXCONN parameter value of the OPTION control statement in the database virtual machine entry in the VM directory must be updated. Refer to the *DB2 Server for VM*

System Administration manual for information on how to define minidisks for the database.

System Action: The database manager terminates.

Operator Response: Refer this message to your system programmer.

System Programmer Response: Increase the MAXCONN value in the VM directory entry for the database virtual machine to allow for the necessary IUCV connections.

ARI0927E **An error occurred while the database manager was opening ({DISKID | CONNECT}) the database. DDNAME =** ddname**. DASD =** cuu**. Return Code =** nnn**.**

Explanation: An error occurred while attempting to open a DB2 Server for VM database minidisk.

If CONNECT is shown in the message, the database manager was using the CMS CMSIUCV CONNECT macro.

If DISKID is shown in the message, the database manager was using the CMS DISKID function. In this situation, the virtual device address (cuu) is not returned; the message displays a 0 for cuu.

The *cuu* value is the virtual device address of the database minidisk.

The *ddname* value identifies the database minidisk:

BDISK Directory minidisk

LOGDSK1
First log minidisk

LOGDSK2
Second log minidisk

DDSKn
DBEXTENT n of the database; n is a 1- to 3-digit number.

For the CMS DISKID function, the Return Code value (the DISKID return code) defines the error condition, as follows:

- Return Code = 4: Invalid call. A DB2 Server for VM or VM system error occurred.
- Return Code = 12: Minidisk was not properly formatted and reserved as a Block I/O minidisk. This could be caused by any of these conditions:
  - Minidisk not formatted (by CMS FORMAT command).
  - Minidisk formatted with a blocksize of 800.
  - Minidisk not reserved for Block I/O (by CMS RESERVE command).
  - Minidisk is an OS, DOS, or VSE disk.

  This is a DB2 Server for VM or VM system error if you successfully defined this minidisk and did not replace it. You could have done that by any of these methods:
  - Database generation (using the SQLDBINS EXEC or the SQLDBGEN EXEC).
  - Add DBEXTENT operation (using the SQLADBEX EXEC).
  - Log reconfiguration (using the SQLLOG EXEC).

  If you replaced the minidisk yourself, you may not have issued the required CMS FORMAT and RESERVE commands successfully. Refer to the *DB2 Server for VM System Administration* manual for information about how to replace a database minidisk.
- Return Code = 28: A CMS FILEDEF command for *ddname* has not been issued, or the FILEDEF command did not specify DISK and cuu. The **dbname** SQLFDEF file on the database manager production minidisk for this database does not contain a CMS FILEDEF ddname DISK cuu command for this minidisk. One of the following has occurred:
  - You attempted to generate a database without using the EXECs supplied with this product for this purpose.
  - You attempted to add a DBEXTENT or log minidisk to the database without using the EXECs supplied with this product for this purpose.
  - You used incorrect restart procedures for a failure during an Add DBEXTENT operation.
  - During database generation, you edited the control statement file and changed or added DBEXTENT numbers in that file.
  - During an Add DBEXTENT operation, you edited the control statement file and changed or added DBEXTENT numbers in that file.
  - You changed the **dbname** SQLFDEF file so that it no longer issues a FILEDEF command (with DISK and cuu) for the displayed *ddname*.
  - If DDNAME = LOGDSK2, you added a second log minidisk by using the SQLSTART EXEC with parameters STARTUP=L and DUALLOG=Y instead of using the SQLLOG EXEC.
  - The EXEC which last updated (or generated) your database minidisk configuration did not work correctly.
- Return Code = 100: The minidisk is not attached to your virtual machine. The **dbname** SQLFDEF file on the production minidisk for this data base does not contain a CP LINK command for this minidisk. (It does contain a CMS FILEDEF command for this minidisk.) One of the following has occurred:
  - You changed the **dbname** SQLFDEF file so that it no longer issues a LINK command for this minidisk.

  - You issued a CP DETACH command for this minidisk after starting the application server.
  - The EXEC which last updated (or generated) your database minidisk configuration did not work correctly.
- Return Code = 101: The CKD DASD minidisk is not attached to your virtual machine. You issued a CP DETACH command for this minidisk after starting the application server.
- Return Code = 102 through Return Code = 112: A VM system error occurred. The displayed value is obtained from the CP DIAGNOSE X'18' instruction return code. The database manager has added 100 to the value. (For example, if the return code = 02, Return Code = 102.)
- Return Code = 113: A VM system error occurred. The Return Code value displayed the return code from the CP IUCV CONNECT macro plus 100 added by the database manager. For example, if the return code from IUCV is 13, the database manager will display Return Code = 113. In this case, Return Code = 113, indicates that there are no more IUCV connections available.
- Return Code = 201: The FB-512 DASD minidisk is not attached to your virtual machine. You issued a CP DETACH command for this minidisk after starting the application server.
- Return Code = 202 through Return Code = 205: A VM system error occurred. The displayed value is obtained from the CP DIAGNOSE X'20' instruction return code. The database manager has added 200 to the value. (For example, if the return code = 02, Return Code = 202.)
- Return Code = 213: An uncorrectable I/O error occurred while attempting to read the FB-512 DASD minidisk. For this situation, a CP I/O error message (DMK prefix) describing the error is displayed on the system operator (CP) console. This message identifies the database virtual machine and the virtual device address.

The CMS DISKID function is described in the *VM/ESA: CMS Application Development Reference* manual. The CP DIAGNOSE instructions are described in the *VM/ESA: CP Programming Services* manual.

Note: For CMS DISKID errors the virtual device address is not displayed. You can obtain the virtual device address by examining the **dbname** SQLFDEF file on the production minidisk for a CMS FILEDEF command with the displayed *ddname*. This command contains the virtual device address; it has the format:

```
FILEDEF ddname DISK cuu
```

For the CMS CMSIUCV CONNECT macro, the Return Code value defines the error condition, as follows:

- Return Code = 8, 13, 16, 24, or 40: A DB2 Server for VM or VM system error occurred. For Return Code 8, 16, 24, or 40, the Return Code value displayed is the return code from the CMS CMSIUCV CONNECT macro. For Return Code 13, the Return Code value displayed is the return code from the CP IUCV CONNECT macro.
- Return Code = 11 or Return Code = 12: A VM system error occurred. The Return Code value displayed is the return code from the CP IUCV CONNECT macro.
- Return Code = 113 through Return Code = 118: A VM system error occurred. The Return Code value displayed the return code from the CP IUCV CONNECT macro plus 100 added by the database manager. For example, if the return code = 13, the database manager will display Return Code = 113. In this case, Return Code = 113, indicates that there are no more IUCV connections available.
- Return Code = 151 through Return Code = 156: A DB2 Server for VM or VM system error occurred. The displayed value is obtained from the first byte of the field IPUSER in the IUCV SEVER External Interrupt Buffer. The database manager has added 150 to the value. (For example, if IPUSER = 01, Return Code = 151.) The IUCV SEVER function for DASD Block I/O is described in the *VM/ESA: CP Programming Services* manual.
- Return Code = 175: Minidisk is attached to the virtual machine in read-only access mode. The database manager requires read and write access. This should not occur because the **dbname** SQLFDEF file always causes the minidisk to be attached in write mode (by a CP LINK command). This Return Code value is from the database manager, which detects the error from information obtained from CMS.
- Return Code = 200: Minidisk has been formatted (by a CMS FORMAT command) to an incorrect block size. This Return Code value is from the database manager, which detects the error from information obtained from CMS. Required block sizes for database minidisks are:

  ```
  Logs      - 4096 bytes
  DBEXTENTs - 4096 bytes
  Directory -  512 bytes
  ```

  This is a DB2 Server for VM or VM system error if you had successfully defined this minidisk and did not replace it. You could have done that by any of these methods:

  - Database generation (using the SQLDBINS EXEC or the SQLDBGEN EXEC).
  - Add DBEXTENT operation (using the SQLADBEX EXEC).
  - Log reconfiguration (using the SQLLOG EXEC).

  If you replaced the minidisk yourself, you may not have issued the required CMS FORMAT and RESERVE commands successfully. Refer to the *DB2*

Server for VM System Administration manual for information about how to replace a database minidisk.

System Action: The application server terminates. For conditions other than uncorrectable I/O errors, the application server terminates with a virtual machine dump to assist in problem determination.

Operator Response: Save the error message and the virtual machine dump (if any). Notify your system programmer.

System Programmer Response: Take action based on the function (DISKID or CONNECT) and the Return Code value, as follows:

- For DISKID Return Codes:

  04 + Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated support group for your installation.

  12 If you have made an error in reconfiguring the database, perform that procedure again. Otherwise, determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated support group for your installation.

  28 If you have made an error in generating the database (using the SQLDBINS EXEC or SQLDBGEN EXEC), perform that procedure again. Otherwise, obtain a copy of the **dbname** SQLFDEF file from the production minidisk. Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated support group for your installation.

  100 If the error was caused by a CP DETACH command, restart the application server. Otherwise, obtain a copy of the **dbname** SQLFDEF file from the production minidisk. Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated support group for your installation.

  101 or 201
  Restart the application server.

  102 through 112 and 202 through 205
  + Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated VM support group for your installation.

  113 or 213 -
  Contact the system operator to obtain the cause of the I/O error. Operator action or hardware service may be required to remove the cause of the error. If you have an unrecoverable DASD media error, refer to the *DB2 Server for VM System Administration* manual for information about how to replace a database minidisk.

- For CONNECT Return Codes:

  08, 13, 16, 24, 40, or 151 through 156
  + Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated support group for your installation.

  11, 12, or 114 through 118
  + Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated VM support group for your installation.

  175 If you have made an error in generating the database (using the SQLDBINS EXEC or SQLDBGEN EXEC), perform that procedure again. Otherwise, obtain a copy of the **dbname** SQLFDEF file from the production minidisk. Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated support group for your installation.

  200 If you have made an error in reconfiguring the database, perform that procedure again. Otherwise, determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated support group for your installation.

- All Return Code values: Unless an unrecoverable DASD media error occurred, restart the application server (possibly after the DASD is repaired). If the error occurred during an Add DBEXTENT or Add DBSPACE operation, refer to the *DB2 Server for VM System Administration* manual for information on recovery from system failures. If the error occurred during a database generation, restart the process from the beginning (using the SQLDBINS EXEC or the SQLDBGEN EXEC).

ARI0928E **An error occurred while the database manager was closing the database. DDNAME =** ddname**. DASD =** cuu**. Return Code =** nnn**.**

Explanation: An error occurred while attempting to close a database minidisk using the CP IUCV SEVER macro or the CMS CMSIUCV SEVER macro.

The cuu value is the virtual device address of the database minidisk.

The DDNAME value identifies the database minidisk:

BDISK Directory minidisk

LOGDSK1
First log minidisk

LOGDSK2
Second log minidisk

DDSKn
DBEXTENT n of the database; n is a 1- to 3-digit number.

The Return Code value defines the error condition, as follows:

- Return Code = 1 through Return Code = 99: These are values from the CMS CMSIUCV SEVER macro return code. A DB2 Server for VM or VM system error occurred.
- Return Code = 100 or more: A VM system error occurred. The displayed value is obtained from the CP IUCV SEVER macro return code. The database manager has added 100 to the value. (For example, if SEVER return code = 01, Return Code = 101.)

The CMS CMSIUCV SEVER macro is described in the *VM/ESA: CMS Application Development Reference for Assembler* manual, and the CP IUCV SEVER macro is described in the *VM/ESA: CP Programming Services* manual.

Database close failures do not affect the database. No data is written during a close operation (or after a close operation).

System Action: The application server terminates with a virtual machine dump to assist in problem determination.

Operator Response: Save the error message (and any other error messages) and the virtual machine dump. Notify your system programmer.

Note: The DB2 Server for VM run has completed successfully unless:

- You were running the SQLDBINS EXEC.
- You were running an EXEC that starts the application server more than once.
- You received other error messages indicating specific error conditions.

System Programmer Response: Take action based on the Return Code value.

- Return Code = 1 through Return Code = 99: Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated support group for your installation.
- Return Code = 100 or more: Determine the service levels of VM and the DB2 Server for VM product. Make a record of what went wrong and contact the designated VM support group for your installation.

Also see note under Operator Response; if you were running the SQLDBINS EXEC to generate a database, you must rerun the SQLDBINS EXEC.

ARI0929E **Incorrect pool number {BLANK | value) on POOL statement.**

Explanation: During database generation or the ADD DBEXTENT function, a POOL control statement with an incorrect storage pool number was encountered. The incorrect storage pool number is identified by **BLANK** or *value* in the message text. The storage pool number was incorrect for one of the following reasons:

- It was omitted (**BLANK** in the message text).
- It was not numeric.
- Its value was less than 2 (storage pool 1 cannot be specified, and numbers less than 1 are not valid storage pool numbers).
- It was greater than the maximum number of storage pools defined for the database (by the database generation MAXPOOLS control statement). The SHOW DBCONFIG operator command will tell you how many storage pools are defined in an existing database.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Supply the correct POOL control statement information and rerun the program.

ARI0930E **Incorrect keyword** keyword **on POOL statement.**

Explanation: During database generation or ADD DBEXTENT processing, an error was detected. The second operand (*keyword* in message text) of the POOL control statement was incorrectly specified. The second operand of the POOL control statement can be either LOG or NOLOG (or can be omitted).

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Supply the correct POOL control statement information and rerun the program.

ARI0931E **Pool number** n **is not an empty storage pool.**

Explanation: During ADD DBEXTENT processing, a POOL control statement specified a storage pool number (*n* in message text) that already has either one or more DBEXTENTs, or one or more DBSPACEs assigned to it. You can specify only a storage pool that does not yet have any DBEXTENTs or DBSPACEs

assigned to it. The SHOW DBEXTENT, or SHOW POOL operator command will tell you which storage pools currently have DBEXTENTs assigned to them. Query the SYSDBSPACES catalog table to determine which storage pools currently have DBSPACEs assigned to them.

Note: You can specify an empty storage pool and then follow the POOL control statement with a DBEXTENT control statement that assigns a DBEXTENT to the storage pool.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Supply the correct POOL control statement information and rerun the program.

ARI0932E **Number of DBEXTENTS exceeds CUREXTNT value** value**.**

Explanation: During database generation one of the following errors occurred:

- The number of DBEXTENT control statements exceeded the value (*value* in the message text) specified in the CUREXTNT control statement. The CUREXTNT value specifies exactly how many DBEXTENTs you must define during database generation.
- The END delimiter control statement, which must follow the last DBEXTENT control statement (and precede the DBSPACE control statements), was omitted.

System Action: The program ends.

Operator Response: Return the console output with the output listing.

System Programmer Response: Verify that the CUREXTNT control statement was specified with the correct value. Ensure that the correct number of DBEXTENT control statements are specified and that they are followed by an END delimiter control statement. Correct the error and rerun the program.

ARI0933E **DBEXTENT number** extno **specified as input to** action**. DBEXTENT is invalid. Reason Code =** n**.**

Explanation: You have specified an invalid DBEXTENT number *extno* to add or delete. The type of action is signified by *action*. The reason is signified by the value *n*. For ADD DBEXTENT, *n* is one of the following:

1. You did not enter a value.
2. The value encountered a nonnumeric character.
3. You entered a value less than one.
4. You did not enter a value that is exactly one greater than the number of DBEXTENTs currently defined nor one of the previously deleted DBEXTENT numbers.
5. The deleted area is not large enough to hold the new DBEXTENT. Use the SHOW POOL command to determine the size of the deleted area.
6. The free area at the end of the DBEXTENT control area in the directory is not large enough to hold the new DBEXTENT. Use the SHOW POOL command to determine the size of this free area.

For DELETE DBEXTENT, *n* is one of the following:

1. You did not enter a value.
2. The database manager encountered a nonnumeric character.
3. You entered a value less than one.
4. You entered a value that is greater that the maximum number of DBEXTENTs allowed in the database.
5. The DBEXTENT is in the database but not in the storage pool that you specified. Use the SHOW POOL command to determine the pool number or use the default value by not inputting the pool number.

System Action: The program ends.

System Programmer Response: Rerun the program with the correct values.

ARI0934E **Error occurred while deleting DBEXTENT number** extno**. Reason Code =** n**.**

Explanation: The ADD/DELETE DBEXTENT processing encountered an error. It occurred while the function was deleting DBEXTENT number *extno*. The function has added or deleted all DBEXTENT numbers in the input file prior to *extno*. The function will not process any DBEXTENT numbers in the input file after *extno*. The value of *n* signifies the error.

1. The storage pool has insufficient space to relocate occupied pages in the DBEXTENT to be deleted. The function must copy all the occupied pages in the DBEXTENT to another location before deleting the DBEXTENT. The storage pool does not have enough space to accommodate them. (Ensure you have not exceeded the SOSLEVEL and there is enough space in the storage pool.) Use the operator command SHOW POOL to determine the number of allocated pages on the DBEXTENT to be deleted.
2. The error occurred while the function was reading the allocated pages.
3. The error occurred while the function was locating a new page for relocation. This might be an I/O error if the database manager was accessing the directory blocks.

System Action: The program ends. The DBEXTENT *extno* is not changed.

System Programmer Response: Determine the cause of the error from the reason code. If the storage pool does not have enough space, add DBEXTENTS to the same pool before invoking the deletion. (Ensure you have not exceeded the SOSLEVEL and there is enough space in the storage pool. You may have to increase the SOSLEVEL to 95%.) For other errors, follow your installation's problem determination procedure.

ARI0935E Archopt **failed. Correct the cause of error. Do a normal SQLSTART to do the archive (or user archive).**

Explanation: Processing failed during the archive. *Archopt* is the type of archive the program is taking.

System Action: The program ends. All the DBEXTENTs specified have been added or deleted successfully.

System Programmer Response: Determine and correct the cause of the error. Do a normal SQLSTART of the application server and take the archive (or user archive).

ARI0936I **Warning: DBEXTENT number** extno **is already in place. Program continues.**

Explanation: The DBEXTENT number *extno* that you wanted to add is already in the database.

System Action: The program continues.

System Programmer Response: Ensure that the DBEXTENT number is correct. If the DBEXTENT was added in a previous run, no action is required. If you want to move the DBEXTENT from one storage pool to another, delete the DBEXTENT in a separate run and rerun the program.

ARI0937I **Warning: DBEXTENT number** extno **that you want to delete is not found. Program continues.**

Explanation: The DBEXTENT to be deleted was not found in the database. A previous run of a program may have deleted it.

System Action: The program continues.

System Programmer Response: Ensure that the DBEXTENT number is correct. If the DBEXTENT was deleted in a previous run, no action is required. Otherwise, rerun the program with the correct values.

ARI0938E **You have specified DBEXTENT number** extno **for deletion more than once. Processing terminates.**

Explanation: You cannot delete a DBEXTENT more than once in a single execution of the ADD or DELETE DBEXTENT function, even though the DBEXTENT may have been added back to the database in the interim.

System Action: Processing ends.

System Programmer Response: Rerun the program with proper input.

ARI0939E **Storage pool** poolno **contains the INTERNAL DBSPACE and the DBEXTENT** extno **specified to be deleted is the only DBEXTENT in this pool. DBEXTENT** extno **cannot be deleted. Processing terminates.**

Explanation: Storage pool *poolno* contains the INTERNAL DBSPACE, and the DBEXTENT *extno* that you specified for deletion is the only DBEXTENT left in this pool. You cannot delete DBEXTENT *extno*.

System Action: Processing ends.

System Programmer Response: Move the INTERNAL DBSPACE to another pool using the ADD DBSPACE function or add more DBEXTENTs to the storage pool before you delete the DBEXTENT.

ARI0940I **Additional 512-byte directory blocks required:** *nnnnnnnnn*

Explanation: The directory has been expanded previously to allow more DBSPACE pages to be added. To expand the directory again to allow both DBSPACE and DBEXTENT pages to be added, the new directory must be large enough to hold both the previously expanded DBSPACE pages as well as the DBEXTENT pages to be added. The new directory is not large enough. The message displays the additional number of directory blocks that are required to expand the directory.

System Action: Processing ends.

System Programmer Response: This message indicates how much additional space is required. Specify a larger VSAM Extent and re-run the job. Refer to the *DB2 Server for VM System Administration* manual to determine the size of the new minidisk.

ARI0944D **Do you wish to reformat or reconfigure? Enter 0 to REFORMAT (erase only log data). Enter 1 to RECONFIGURE (erase log data and log history).**

Explanation: You should reply `0` if you are reformatting the log or logs. The following events require log reformatting:

- Switching from LOGMODE=A or L to LOGMODE=N
- Resetting the logs because the application server cannot be warm started because of a logical error in the log.

- Avoiding log recovery when restoring a database from a back-level database archive.
- Switching from dual to single logging. This may be considered a case of reconfiguring the logs, but does not require the CMS FORMAT and RESERVE commands.

If you reply 1 to reconfigure the logs, CMS FORMAT and RESERVE commands will be issued for the log minidisks identified by message ARI0688D for single logging or message ARI6129D for dual logging.

The following events require log reconfiguring:

- Changing the size of your logs
- Changing the location of one or both of your logs (possibly due to a log I/O error)
- Switching from single to dual logging.

System Action: CMS FORMAT and RESERVE commands are invoked on the log disks if the reply is 1.

In either case, the logs are reformatted to indicate no prior logging activity and initialization continues. (Any logging of database updates is lost.)

Operator Response: Decide whether to reply 0 or 1. Enter your reply.

ARI0945I Expanded space will be used for DBSPACE pages.

Explanation: You have requested to expand the directory and to use the additional space to hold more DBSPACE pages.

System Action: Processing continues.

ARI0946I Expanded space will be used for DBSPACE and DBEXTENT pages.

Explanation: EXPAND = ALL was supplied in the parameter list. The directory expansion process will expand the directory to allow more DBEXTENT and DBSPACE pages to be added.

System Action: Processing continues.

ARI0947I Current|New maximum DBEXTENT|DBSPACE pages: *nnnnnnnnn*

Explanation: This message shows the (Current|New) maximum number of (DBEXTENT|DBSPACE) pages that the directory can hold.

System Action: Processing continues.

ARI0948I DBEXTENT|DBSPACE pages added: *nnnnnnnnn*

Explanation: This message shows the number of additional (DBEXTENT|DBSPACE) pages that will be available in the new directory.

System Action: Processing continues.

ARI0949I (Current|New) directory size (blocks): *nnnnnnnnn*

Explanation: This message shows the size of the (Current|New) directory in 512-byte pages.

System Action: Processing continues.

Chapter 16. ARI0950–ARI0987 Trace Formatter Messages

ARI0950E INPUT keyword keyword **is not valid.**

Explanation: The submitted control statement keyword is unrecognized. The recognized trace formatter control statement keywords are: AGENTNO, COMP, DATE, DUMPALL, HEADER, RETCODE, SUBCOMP, TIME, TRACENO, and USERID.

System Action: The program ends.

Programmer Response: Select the correct control statement keyword. Each control statement can contain only one keyword.

ARI0951E Number of parameters in user ID exceeds six.

Explanation: The maximum number of parameters used with the control statement keyword USER ID is six.

System Action: The program ends.

Programmer Response: Reduce the number of parameters to six or less. For more information about USER ID parameters, refer to the *DB2 Server for VSE & VM Operation* manual.

ARI0952E Number of parameters in AGENTNO exceeds six.

Explanation: The maximum number of parameters used with the control statement keyword AGENTNO is six.

System Action: The program ends.

Programmer Response: Reduce the number of parameters to six or less. For more information about AGENTNO parameters, refer to the *DB2 Server for VSE & VM Operation* manual.

ARI0953E Nondigit character in one of the AGENTNO parameters.

Explanation: Syntax error. At least one of the parameters used with the control statement keyword AGENTNO contains nonnumeric characters.

System Action: The program ends.

Programmer Response: Submit the parameters with the correct syntax. For more information about AGENTNO parameters, refer to the *DB2 Server for VSE & VM Operation* manual.

ARI0954E Number of parameters in TRACENO exceeds six.

Explanation: The maximum number of parameters used with the control statement keyword TRACENO is six.

System Action: The program ends.

Programmer Response: Reduce the number of parameters to six or less. For more information about TRACENO parameters, refer to the *DB2 Server for VSE & VM Operation* manual.

ARI0955E Nondigit character in one of the TRACENO parameters.

Explanation: Syntax error. At least one of the parameters used with the control statement keyword TRACENO contains nonnumeric characters.

System Action: The program ends.

Programmer Response: Submit the parameters with the correct syntax. For more information about TRACENO parameters, refer to the *DB2 Server for VSE & VM Operation* manual.

ARI0956E Number of parameters in COMP exceeds one.

Explanation: The maximum number of parameters used with the control statement keyword COMP is one.

System Action: The program ends.

Programmer Response: Enter only one of the allowed parameters: CONV, DBSS, DRRM DSC, RA, RDS, WUM, or STG.

ARI0957E Parameter in COMP not valid.

Explanation: The parameter specified with the control statement keyword COMP can be CONV, DBSS, DRRM DSC, RA, RDS, WUM, or STG.

System Action: The program ends.

Programmer Response: Submit the correct parameter: CONV, DBSS, DRRM, DSC, RA, RDS, WUM, or STG.

ARI0958E Number of parameters in SUBCOMP exceeds eight.

Explanation: The maximum number of parameters used with the control statement keyword SUBCOMP is eight.

System Action: The program ends.

© Copyright IBM Corp. 1987, 2000

Programmer Response: Reduce the number of parameters to eight or less. For more information about SUBCOMP parameters, refer to the *DB2 Server for VSE & VM Operation* manual.

ARI0959E Parameter in SUBCOMP not valid.

Explanation: At least one of the parameters used with the control statement keyword SUBCOMP is not valid. The allowable parameters for the subcomponents DBSS, DRRM, DSC, RA and RDS are:

For RDS:	For DBSS:	For DSC:	For RA:
AU	DC	AGENT	RA
AG	DM	COM	COM
EXEC	ENTRY		
INT	EXIT		
OPT	INDEX	For DRRM:	
PA	LOCK		
SG	LOG	DICT	
	LUW	GEN	
	SORT	PARSE	
	STAT	RDIIN	
	STOR		

System Action: The program ends.

Programmer Response: Submit the correct parameters. For more information about SUBCOMP parameters, refer to the *DB2 Server for VSE & VM Operation* manual.

ARI0960E Number of parameters in DATE exceeds two.

Explanation: The maximum number of parameters used with the control statement keyword DATE is two.

System Action: The program ends.

Programmer Response: Input the two parameters 'start-date' and 'end-date' or the parameter 'target-date'.

ARI0961E Syntax error in one of the DATE parameters.

Explanation: Syntax error. Each portion of the date must be specified as two digits; that is, a leading zero must be entered where needed. The delimiter must be /. The second date, if used, must be greater than the first date. An example of correctly specified dates is: DATE 01/23/85 02/08/85.

System Action: The program ends.

Programmer Response: Submit the parameters with the correct format.

ARI0962E Nondigit character in one of the DATE parameters.

Explanation: Syntax error. At least one of the parameters used with the control statement keyword DATE contains nonnumeric characters. Each portion of the date must be specified as two digits; that is, a leading zero must be entered where needed. The second date, if used, must be greater than the first date. An example of correctly specified dates is: DATE 01/23/85 02/08/85.

System Action: The program ends.

Programmer Response: Submit the parameters with the correct syntax.

ARI0963E Number of parameters in TIME exceeds two.

Explanation: The maximum number of parameters used with the control statement keyword TIME is two.

System Action: The program ends.

Programmer Response: Input the two parameters 'start-time' and 'end-time' or the parameter 'target-time'.

ARI0964E Syntax error in one of the TIME parameters.

Explanation: Syntax error. Each portion of the time must be specified as two digits; that is, a leading zero must be entered where needed. The delimiter must be a colon (:).

Note: The second time specified should be greater than the first time. For example, specifying TIME 23:45:00 00:10:00 will cause no trace output to be selected (no warning message will be issued). Consequently, specifying an interval that passes through midnight must be done in two different runs of the trace formatter. An example of correctly specified times is: TIME 09:05:00 14:04:59. It is possible (but not very useful) to specify only one time with the TIME keyword. For example, TIME 12:00:00 specifies that only the trace records created during that second of time be formatted.

System Action: The program ends.

Programmer Response: Submit the parameters with the correct syntax.

ARI0965E Nondigit character in one of the TIME parameters.

Explanation: Syntax error. At least one of the parameters used with the control statement keyword TIME contains nonnumeric characters. Each portion of the date must be specified as two digits; that is, a leading zero must be entered where needed. The delimiter must be ':'.

Note: The second time specified should be greater than the first time. For example, specifying TIME 23:45:00 00:10:00 will cause no trace output to be selected (no warning message will be issued). Consequently, specifying an interval that passes through midnight

must be done in two different runs of the trace formatter. An example of correctly specified times is: TIME 09:05:00 14:04:59. It is possible (but not very useful) to specify only one time with the TIME keyword. For example, TIME 12:00:00 specifies that only the trace records created during that second of time be formatted.

System Action: The program ends.

Programmer Response: Submit the parameters with the correct syntax.

ARI0966E Number of parameters in RETCODE exceeds one.

Explanation: The maximum number of parameters used with the control statement keyword RETCODE is one.

System Action: The program ends.

Programmer Response: Enter only one of the allowed parameters (P, N, or *).

ARI0967E Parameter in RETCODE not valid.

Explanation: The submitted parameter used with the control statement keyword RETCODE is not valid. The parameter must be P, N, or *.

System Action: The program ends.

Programmer Response: Enter one of the following as the RETCODE parameter:

P Positive (nonzero) return codes only

N Negative return codes only

* All nonzero return codes

ARI0968E The trace file does not contain trace data.

Explanation: The trace file (tape or CMS file) does not contain the DB2 Server for VM trace data. (The wrong tape may be mounted.)

System Action: The program ends.

Programmer Response: Ensure that the operator mounted the correct tape and that the correct CMS FILEDEF command was entered for ddname ARITRAC. If the correct file is being accessed, it does not contain the trace data. (The tape is probably empty or rewritten.)

ARI0969I Trace file is empty.

Explanation: The DB2 Server for VM trace formatter program has detected that the DB2 Server for VM trace file is empty. (It contains no records.) When database manager activity was being traced, no trace output data was created.

System Action: The program ends.

Programmer Response: None required. You may want to verify that the correct file was submitted as input to the trace formatter program.

ARI0972E Storage allocation error. Return Code = nn.

Explanation: In allocating virtual storage during trace formatting, a CMS free storage macro failure occurred. The cause of the error is described by the Return Code value *nn* (which is the return code from the free storage macro).

System Action: The program ends.

Programmer Response: Report the problem to your system programmer.

System Programmer Response:

- Return Code = 1: The DB2 Server for VM trace formatter program has insufficient virtual storage. At least 65K is required to run the program.
- For any other Return Code value: Make a record of what went wrong and contact the designated support group for your installation.

ARI0973E Error releasing storage. Return Code = nn.

Explanation: In releasing virtual storage during trace formatting, a CMS free storage macro failure occurred. The cause of the error is described by the Return Code value *nn* (which is the return code from the free storage macro).

System Action: The program ends.

Operator Response: Report the problem to your system programmer.

System Programmer Response: Make a record of the return code and contact the designed support group for your installation.

ARI0974E operation **error on DDNAME** xxxxxxxx**. Return Code =** nn**. EREASON=**nn**.**

Explanation: An error occurred during a sequential input/output operation on the *xxxxxxxx* file.

The Return Code value is the Primary Error Code value. The EREASON value is the Secondary Error Code value. For further explanation, look up the displayed error codes under either “Chapter 23. Sequential File Errors” on page 407, or “SEQ FILE ERRORS” in the HELP text.

System Action: The program ends.

Operator Response: Return console output with the output listing.

System Programmer Response: For corrective action,

look up the displayed error codes under either "Chapter 23. Sequential File Errors" on page 407, or "SEQ FILE ERRORS" in the HELP text.

ARI0975E Number of parameters in EXTLUWID exceeds one.

Explanation: The maximum number of parameters used with the control statement keyword EXTLUWID is one.

System Action: The program ends.

Programmer Response: Enter only one parameter.

ARI0976E Parameter in EXTLUWID is invalid.

Explanation: The specified parameter used with the control statement keyword EXTLUWID is not valid. The parameter must be specified again.

System Action: The program ends.

Programmer Response: Specify a valid LUWID in external format.

ARI0980A DB2 Server for VM will invoke CMS XEDIT to allow you to prepare the trace formatter control statements. You will use XEDIT to edit the CMS file SQLTRFMT SQLTRACE A1. PF3 will store the edited file. Press Enter when you are ready to edit this file.

Explanation: Control statements are needed by the DB2 Server for VM trace formatter. These statements are read from a card-image file named SQLTRFMT SQLTRACE A1, which is the SYSIN file for the trace formatter.

When you press the ENTER key, the CMS XEDIT facility will be invoked to let you create (or update) this file. In XEDIT you can enter control statements in lowercase letters, but they will be stored in uppercase form. In XEDIT the PF3 key will be set to do the FILE operation (storing a copy of the file on your A-disk). After you store the file on your A-disk, SQLTRFMT EXEC processing continues.

The SQLTRFMT SQLTRACE A1 file is not erased by SQLTRFMT EXEC processing. This file is kept on your A-disk so that it can be used again. Each time you use the SQLTRFMT EXEC, XEDIT presents you with the previous copy of the file (if any).

System Action: SQLTRFMT waits for you to press the ENTER key. When you press the ENTER key, the CMS XEDIT facility will be invoked to let you create (or update) the SQLTRFMT SQLTRACE A1 file.

Operator or User Response: Press the ENTER key. Create or update the control statements for the trace formatter. Press the PF3 key when you are ready to have the trace formatter use the control statements. The control statements are described in the *DB2 Server for VSE & VM Diagnosis Guide and Reference* manual.

ARI0981D Trace formatter output can be a virtual print file or a CMS file. Enter PRINT for print file or enter CMS for CMS file.

Explanation: The trace formatter output (SYSPRINT file) can be directed to either the virtual printer or a CMS file. This prompting message asks you to make your choice.

System Action: The system waits for your reply.

If you reply PRINT, the trace formatter output will be sent to the virtual printer.

If you reply CMS, message ARI0983A will be displayed so that you can enter the file name, file type, and file mode of the CMS file to be used for the output.

Any other reply causes message ARI0981D to be displayed again.

Operator or User Response: If you want to look at the trace formatter output immediately (using an editor), type CMS. You can later print the file with a CMS PRINT command (using the CC option).

If you want only printed output from the trace formatter, reply PRINT.

ARI0982I Trace formatter program ended successfully.

Explanation: The DB2 Server for VM trace formatter has successfully selected, formatted, and displayed the trace file. No errors occurred.

System Action: The SQLTRFMT EXEC completes processing with a return code of 0.

ARI0983A Enter file name, file type, and (optional) file mode for CMS output file. File mode defaults to A1.

Explanation: You have chosen (through message ARI0981D) to direct the output of the trace formatter to a CMS file. The SQLTRFMT EXEC is now requesting you to specify the file name, file type, and (optionally) the file mode of that CMS file.

System Action: The system waits for your reply.

If you enter three values (separated by one or more blanks), they will be used as the file name, file type, and file mode, respectively.

If you enter two values (separated by one or more blanks), they will be used as the file name and file type, respectively; file mode will default to A1.

If you enter fewer than two values, this message will be displayed again.

Operator or User Response: Enter any valid CMS file

name and file type. If you want the file to be placed on a minidisk other than your A-disk, also enter a valid CMS file mode. Separate your entries with one or more blanks.

ARI0984E Trace formatter program ended with an error.

Explanation: The DB2 Server for VM trace formatter ended with an error condition. The error message is contained in the trace formatter output file. If the error was an I/O error on the output file (SYSPRINT), the error message is displayed on the virtual machine console.

System Action: The SQLTRFMT EXEC completes processing with a return code of 8.

Operator or User Response: Take action based on the error message in the trace formatter output file. Save any CMS messages (those having a DMS prefix) for use in problem determination.

ARI0985E An error occurred in the FILEDEF process. DDNAME=xxxxxxx **Return Code=**nn**.**

Explanation: An error occurred while executing the CMS FILEDEF command for the file identified by the DDNAME value. The Return Code value is the CMS FILEDEF command return code. The FILEDEF command was issued by the DB2 Server for VM trace formatter EXEC SQLTRFMT.

System Action: The SQLTRFMT EXEC is ended with the Return Code value.

Operator or User Response: If the DDNAME value is SYSPRINT and you entered the CMS file name, file type, and file mode, verify that you entered the correct values. Use the CMS error message (message with prefix DMS) to verify this. If you made a specification error, re-execute the SQLTRFMT EXEC. Message ARI0983D prompts you to enter the file name, file type, and optionally the file mode for the SYSPRINT CMS output file.

Otherwise, this is a DB2 Server for VM system error. Save the EXEC output and give it to your system programmer.

System Programmer Response: If the terminal user did not specify an improper CMS file name, file type, or file mode for the SYSPRINT FILEDEF command, this is a system error in the SQLTRFMT EXEC. Determine the service level of the system and report this problem to the designated support group for your installation.

ARI0986E An error occurred in the NUCXLOAD process. NAME=ARIMTRA Return Code=nn**.**

Explanation: An error occurred while executing the CMS NUCXLOAD command for DB2 Server for VM load module ARIMTRA. This load module is in the ARISQLLD LOADLIB file on the DB2 Server for VM production minidisk. The Return Code value is the NUCXLOAD command return code.

Note that the CMS NUCXLOAD command does not work in the CMS DOS environment. Ensure that the CMS DOS environment has not been set on before executing the SQLTRFMT EXEC.

System Action: The EXEC SQLTRFMT is ended with the Return Code value.

Operator or User Response: Ensure that the CMS DOS environment has not been established for the virtual machine. Consult the *VM/ESA: CMS Command Reference* manual for a description of the CMS NUCXLOAD command return codes. If a CMS error message (one with prefix DMS) was displayed, consult the *VM/ESA: System Messages and Codes* manual. Take the corrective action indicated or suggested by the message or return code. If you cannot resolve the problem, consult your system programmer.

System Programmer Response: Perform problem determination as indicated above. If a DB2 Server for VM or VM system error occurred, make a record of what went wrong and contact the designated support group for your installation.

ARI0987E An error occurred in the NUCXDROP process. NAME=ARIMTRA Return Code=nn**.**

Explanation: An error occurred while executing the CMS NUCXDROP command for the DB2 Server for VM load module ARIMTRA. The Return Code value is the NUCXDROP command return code. This is a database manager or VM system error. (The NUCXDROP command is described in *VM/ESA: CMS Command Reference* manual for your system.)

Note: The DB2 Server for VM trace formatter program has completed processing. It has produced its output in either a virtual printer file or a CMS file, according to the options selected by the user of the SQLTRFMT EXEC.

System Action: The EXEC SQLTRFMT is ended with the Return Code value.

Operator or User Response: Save the EXEC output and give it to your system programmer. You can take normal action to examine or print the trace formatter output. The failure to do the NUCXDROP can be circumvented by re-ipling CMS. (Load module ARIMTRA is still occupying virtual storage.)

System Programmer Response: Determine the service level of the database manager and report this problem to the designated support group for your installation.

Chapter 17. VM Abend Codes

Notify your system programmer whenever one of these abend conditions occurs:

Code	Meaning
3	Error on a DIAGNOSE command issued to clear the screen before writing a line.
4	ISQL is abending because of a previous severe error.
5	An RDTERM/LINERD error occurred.
7	The SENTRIES command failed during routine processing.
9	The CONWAIT command failed during routine processing.
11	The DESBUF command failed during routine processing.
13	The DROPBUF command failed during routine processing.
15	The MAKEBUF command failed during routine processing.
17	An error occurred while using the ATTN function to add routine commands to the stack.
18	CMS set FULLSCREEN command failed.
19	Error on a DIAGNOSE/CONSOLE command issued to write a full screen (query result display).
21	Error on a DIAGNOSE/CONSOLE command issued to clear the screen after an END command.

free storage Return Code
The free storage request failed.

WRTERM/LINEWRT Return Code
A WRTERM/LINEWRT error occurred during CANCEL processing.

© Copyright IBM Corp. 1987, 2000

Chapter 18. DBSS Return Codes

DBSS return codes are generally found in SQLERRD2 (displayed by message ARI0504I). A very few error messages will also directly display the DBSS return codes (for example, ARI0523E).

Note: Many of the DBSS return codes will not be of interest or value to the end user, system programmer, etc. These will be of most use to the IBM support personnel.

The return codes are in three groups:

0 Successful. Operation executed. No unusual conditions occurred.

Positive Warning. Operation executed, but one or more unusual conditions occurred.

Negative Error. Operation not executed, database not changed, database only partially updated, or the logical unit of work has been backed up.

Warning Return Codes

Return Code Name
Meaning

1 DIRTSCAN
Used in read call. For Fetch Row, Open Scan, or Get Parent, scan is 'on' a row but flagged dirty. For Get Next Row, scan is flagged dirty and may or may not be 'on' a row. Get Next Row resets the scan to 'clean' and resets the scan to 'on' (or 'EOF').

2 DOMTRUNC
Field truncation. On a read request, a requested domain is longer than the domain FREQLTH value and thus the returned domain has low order truncation.

4 EOF End of file on a scanned set

4 TOF Top of file on a scanned set. No operation currently sets this code.

8 KEYVIOL
Key violation on Get Next Row for an index scan. Submitted key lower than the current key as indicated by SCAN and QUALF. Scan is positioned 'on' first offending row.

16 NFORMSEG
DBSPACE requested by CFETCH, CNEXT exists but has no SCR record or was not opened

64 FORCEND
Automatic commit work is generated on deletion of DBSPACE, master or link control records

128 BADSARG
Record does not satisfy the search arguments

4096 INCONSEG
NSEG or SEGMS(*) is inconsistent with the REL(*) supplied to STARTUS call.

Bad Return Codes

Bad DBSS Call Parameters

-1 ILOPCODE Illegal OPCODE on DBSS call

-2 ILLEGALP Illegal parameter (PARAMS) pointer on DBSS call (0, or not word/doubleword aligned).

Illegal parameters for BASE. Some codes also used by CBASE, SBASE.

-3 ISEGMENT
Illegal DBSPACE number, missing SCR, or type conflict with operator

-4 ILLRID Illegal table ID or illegal operation for list

-5 NFTID TID not found or open of an empty table or list

-6 ILLPSEGM Illegal parent DBSPACE number

-7 ILLPRID Illegal PRID or illegal operation for list

-8 NFPTID PTID not found

-9 ILSCANID Illegal scan ID

© Copyright IBM Corp. 1987, 2000

DBSS Return Codes

-10 INKEYDOM
Illegal number of key fields

-11 ILLKPTR Illegal address of KDOMAINS

-12 INREQDOM
Illegal number of DOMAINS fields

-13 ILLDPTR Illegal address of DOMAINS

-14 INSARGS Illegal number of search arguments

-15 ILLSPTR Illegal address of search arguments

-16 ILLEGLID Illegal link ID

-17 ILLEGIID Illegal index ID

-18 ILLICOMP
Illegal comparison operator (ICOMP)

-19 ILLQUALF
Illegal call qualifier in BASE (QUALF); illegal call qualifier in CBASE (CQUALF)

Illegal Parameters in KDOMAINS

-20 IKDOMPTR
Illegal key-field address

-21 IKDOMLTH
Illegal key-field length

Illegal Parameters in DOMAINS

-22 IFLDPTR Illegal field address

-23 IFACLTH Illegal submitted field length

Illegal Parameters in Search Arguments

-24 ISNUMBER
Illegal field number in search arguments

-25 ISPOINT Illegal field address in search arguments

-26 ISLENGTH
Illegal field length in search arguments

-27 ISCOMP Illegal comparison operator in search arguments

-28 ISBOP Illegal boolean operator in search arguments

Exceptional Conditions on a Scan

-29 EMPSCAN
Empty scan in SCANID

-30 SCANOW Too many scans are opened by RDS

-31 UPDIRTY Write call using a dirty scan

-32 NOTON Call different from NEXT, CLOSE using a scan which is not 'on'

Exceptional Conditions on an Index or on a Row

-33 NFKEY Submitted key not found

-34 UNIQKEY Violation of index key-uniqueness

-35 LONGKEY
Submitted key is too long

-36 LONGTPLE
Submitted row is too long

-37 ILLTYPE2 Illegal call for a Type 2 table. (Should never occur because the database manager does not have Type 2 tables)

Exceptional Conditions on a Link

-38 CLINKED Delete of a row illegally disconnecting some link; also connection of a row in a link in which it is already connected

-39 CNLINKED
Disconnect, or delete and disconnect of a non-linked child; link open on a non-linked child in a binary link; connect, insert and connect to a non-linked child in a binary link

-40 NOPARENT
Parent does not exist (in PARENT call)

-41 NOCHILD
Child does not exist (in OPEN call)

Invalid Index Conditions

-42 INVENTU Attempt made to use an index that has been marked invalid

-43 INVENTM
An index has been marked invalid during ROLLBACK or UNDO

-44 NOINVALD
The limit of invalid entities has

been reached when combining the current invalid indexes with the potentially invalid indexes due to active CREATE INDEX and REORGANIZE INDEX operations.

Illegal Parameters in CBASE and Auxiliary Control Structures

-45 IQUALFSG
Illegal QUALFSEG

-46 IAUXPTR Illegal auxiliary pointer

-47 INPAGES Illegal NIPAGES or NHPAGES

-48 ILLFREE Illegal percentage of free space in SCR or ICR

-49 ILOCKPRT
Illegal lock protocol in SCR or ICR

-50 ILLNDOM
Illegal number of field entries in MCR or ICR

-51 ILLDOMS Illegal field entries (lengths in MCR, field numbers in ICR)

-52 ILUNIQUE
Illegal unique key specification

-53 TOOMANYC
Too many control objects as a result of a CINSERT

-54 LISTSEGM
Operation illegal on a list DBSPACE. Only scan operations, Insert Row, and Sort are legal.

Illegal Parameter in BASE or SBASE

-58 ILIXSARG
Illegal index page search argument specified. Index is not being used.

-59 ILLHOLD Illegal HOLDIND or SHOLDIND value from RDS, bad Isolation level/Hold value.

Illegal Conditions Related to Logical Unit of Work Management

-60 NOSTART Begin Work was not issued prior to a Save Work call.

-61 NOTEND COMMIT or ROLLBACK not issued prior to Begin Work call (logical unit of work active)

-62 NOLOG Operation requires a log, but none exists (should not occur on BEGIN WORK or COMMIT as RDS does not support user log data)

-63 INVSCHED
SCHEDULE opcode issued for active logical unit of work

-64 SYABORT System either requested logical unit of work abort, has run out of log space, or transaction reached save point limit of 2147483647. Breaking up the LUW into smaller LUWs with the COMMIT WORK statement will prevent reaching the save point limit.

-65 INLUW Operation not valid while agent has active logical unit of work

-66 OPABORT Logical unit of work rolled back due to operator FORCE command

Illegal Parameters in SBASE

-70 ILLEGALL
Illegal length of the DBSPACE list

-71 ILISTPTR Illegal pointer to DBSPACE list

Exceptional Conditions on a DBSPACE

-72 SGISOPEN
At least one DBSPACE already open

-73 SEMPTIED
Cannot be reopened until a synchronous save because CDELETE was issued

-74 NOAVAILS
No available temporary DBSPACE

-75 NOTAVSEG
DBSPACE not available or write operation with *DBSPACE opened for read*

-76 NOROOMH
No header room in DBSPACE in write call

-77 NOROOME
No entity room in DBSPACE in write call

-78 NOROOMI
No index room in DBSPACE in write call

-79 NOROOMSP
No available space in storage pool nnn

DBSS Errors and Version Limitations

-80 RSSERROR
Error in DBSS code. Error ID in feedback area (Should not occur since DBSS terminates the application server on any detected DBSS code error)

-81 IOERROR DASD I/O error. DBSS always abends the application server on I/O error. (Should not occur)

-82 NOTIMPL Feature not yet implemented. (Should not occur)

-83 NOMEMORY
Unable to allocate needed main storage

RETURN CODES USED ONLY IN SYSTEM WITH LOG

Illegal Parameters in LUW Operators

-85 OUTSIDE Pointer to user save data that is bad. (Should never occur because RDS does not support user log data)

-86 TOOLONG
User wants to log more than 64K. (Should never occur because RDS does not support user log data)

-87 BADSAVNO
Tried to back over non-existent save point

-89 OKBUTMOR
Read user log data but there was more. (Should never occur since RDS does not support user log data)

-90 INVIDLST
Output area too small for INDOUBT list

-91 SYBACKUP
System requested LUW backup

RETURN CODES USED ONLY IN MULTI-USER SYSTEM

-92 CHECKING
Checkpoint in progress, try later

Illegal Parameters Related to Locking

-94 BADCLVL Consistency level not 1, 2, or 3 (should not occur as RDS always uses consistency level 3)

-95 ILCKMODE
Lock mode not in (SHAR,SIX,EXCL)

Exceptional Conditions Due to Locking

-97 LOCKLIM Too many locks held by this LUW

-98 PRIVSEGM
Requested private DBSPACE is locked

LUW Backout Due to Locking; (Reached Save Point in FEEDBACK(1))

-99 DEADLOCK
LUW backout due to deadlock

-100 OVERLOAD
LUW backout due to too many locks being held system-wide

-101 DEADATSI
Deadlock backout to a DBSI boundary...internal retry failed

RETURN CODES USED ONLY IN SINGLE-USER SYSTEM

-120 NOTMUL
Operation not allowed in single-user system issued by operator command OPCODE (OPCMMD)

MORE RETURN CODES RELATED TO LUW MANAGEMENT

-123 ATTNBACK
OPCODE terminated due to asynchronous request to terminate LUW; caller should issue ROLLBACK. Can occur only if YTABLE1 flag ATTNATTN set to Y. The database manager does not set ATTNATTN flag to Y. Should not occur.

RETURN CODES USED BY THE SORT SUBSYSTEM

-130 ILLSOPTR
Illegal address of SORTSPEC

-131 INORDSPC
Illegal number of sort order specifications

-132 ISORTSPC
Illegal sort order specification

-133 NOEXDOMN
Field number does not exist for input table-id

-134 IDUPELIM
Illegal duplicate elimination option

-135 INSLDOMS
Illegal number of SORTLIST fields

-136 NORMTPSG
No space in temporary DBSPACE for write call

-137 DUPFOUND
Duplicate found during sort/merge; stop due to user option

-138 ILLSLPTR
Illegal pointer to SORTLIST specification

-139 EMPTYSRT
Empty set of rows to sort: sort DBSPACE is unchanged

-140 ILTIDSPC
Illegal TID specification: input list has no TIDs

-141 ILDOMREP
Illegal domain repetition: output list domain in SORTSPEC specified more than once

-142 ENCTPTLG
Encoded row too long: does not fit in an intermediate page

-143 ISORTERR
Internal sort error: number of rows sorted and merged different

RETURN CODES SET BY RDS FOR DBSPACE ACCESS

-200 DISABDB
Attempt to access a disabled DBSPACE

RETURN CODES USED BY THE UPDATE STATISTICS COMMANDS

-400 ILGMODE
Illegal update statistics mode

-401 ILNUMREL
Illegal number of specified tables

-402 DUPRELID
Duplicate table-id in input list

-403 IPATHTYP
Illegal path type

-404 ILPATHID
Illegal path ID

-405 MISMATCH
RELID and PATHID do not correspond

-406 NOTAROOT
First table-id on input list is not a root

-407 NOPARNT
No parent specified for binary link

-409 UNCLUSTI
Unclustered indexes exist

-410 INCONREL
NREL or RELS(*) inconsistent with that previously specified with STARTUS call.

DBSS Return Codes

Chapter 19. SQLCODEs

The SQLCODEs are associated with the SQL messages (ARI0500I - ARI0513E). The SQLCODEs provide specific information about the condition of the system. For more information, refer to the descriptions of the SQL messages issued with the SQLCODE.

An SQLSTATE field as well as an SQLCODE, is returned in the SQLCA. Each SQLSTATE corresponds to an SQLCODE. See "Chapter 20. SQLSTATEs" on page 377 for more information.

SQLCODEs 0, and +12 through +863

The SQLCODE descriptive text identified for these SQLCODEs is displayed as continuation lines for message ARI0501I.

SQLCODEs 0 and 100 are associated with normal conditions.

---

0 **Database manager processing is completed. Warning may indicate a problem.**

Explanation: Execution of the SQL statement has completed. Message ARI0502I should follow this text and indicate the specific warning conditions.

SQLSTATEs:

00000	With no warnings
01004	SQLWARN1=W
01555	SQLWARN1=Z
01003	SQLWARN2=W
01503	SQLWARN3=W
01504	SQLWARN4=W
01511	SQLWARN5=W
01506	SQLWARN7=W
01507	SQLWARN7=Z
01508	SQLWARN8=W
01509	SQLWARN9=W
01510	SQLWARNA=W
01517	SQLWARN8=Z

System Action: Normal processing; the SQL statement has been processed successfully.

User Response: Take action based on message ARI0502I.

+12 **An unqualified column name has been interpreted as a correlated reference.**

Explanation: An unqualified column name has been interpreted as a correlated reference.

SQLSTATE 01545.

System Action: Normal completion.

User Response: Take suitable action based on the SQLCODE descriptive text.

+100 **There are no (or no more) rows that satisfy the condition.**

Explanation: For a query that uses a cursor, the cursor is empty or all rows have been selected. For a query that does not use a cursor, no row was found that satisfied the WHERE condition. An INSERT via SELECT statement may return this SQLCODE if the SELECT statement does not retrieve any rows.

SQLSTATE 02000.

System Action: Normal completion.

User Response: Take suitable action based on the SQLCODE descriptive text.

+114 **The server name part of the** *server-name1.creator.object* **does not match the name of the connected application server** *server-name2***.**

Explanation: The code is used to report that the server name (standalone or qualifying a column) does not match the name of the connected application server. This code is returned only when the program is being preprocessed with the NOEXIST option.

SQLSTATE 01536.

System Action: Processing of the SQL statement is ended. The statement is recorded in the package, and it will be rechecked when the program is executed. Preprocessing continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, modify the statement and preprocess the program again. If the condition is corrected before the statement is executed, execution occurs normally. If the condition persists at execution time, the execution is unsuccessful.

© Copyright IBM Corp. 1987, 2000

In either situation, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, after the statement is corrected. If this section is not executed at the application server specified by *server-name2*, the negative version of this code will be returned.

+117 The number of data values to be inserted does not equal the number of columns specified or implied.

Explanation: The number of values on an INSERT statement must equal the number of columns to be updated. Either too many or too few data items were listed for the columns specified or implied. This code is returned only when the program is being preprocessed with the NOEXIST option.

SQLSTATE 01525.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, either create or delete data items to equal the number of columns specified, before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the correct number of data items exist.

+134 Improper use of long string.

Explanation: The SQL statement improperly references a long string column. Long strings cannot be used to represent datetime values and cannot be referenced in:

- A CREATE INDEX statement
- A function other than SUBSTR or LENGTH
- A GROUP BY clause
- A PRIMARY KEY, FOREIGN KEY, or UNIQUE clause
- A SELECT DISTINCT statement
- A subselect of an INSERT
- A subselect of a UNION or UNION ALL
- An expression in a SET clause of an UPDATE
- An ORDER BY clause
- By itself in a predicate other than LIKE

A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(*n*) where *n* is greater than 254 but less than or equal to 32767, or VARGRAPHIC(*n*) where *n* is greater than 127 but less than or equal to 16383.

Note: This code is returned only when a statement is being preprocessed.

SQLSTATE 01582.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, check the use of the long string column before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the long string column has been checked. Refer also to the *DB2 Server for VSE & VM Application Programming* manual for details on using long strings.

+135 The input for a long string column in an INSERT statement or UPDATE statement must be from a host variable or be the keyword NULL.

Explanation: The UPDATE or INSERT is using constants, column names, or subqueries where it should be using NULL or a host variable.

A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(*n*) where *n* is greater than 254 but less than or equal to 32767, or VARGRAPHIC(*n*) where *n* is greater than 127 but less than or equal to 16383.

Note: This code is returned only when a statement is being preprocessed.

SQLSTATE 01584.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, check the use of the long string column before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by

preprocessing the program again, once the long string column has been checked.

Refer also to the *DB2 Server for VSE & VM Application Programming* manual for details on using long strings. Correct the statement. Try again.

+142 *statement-type* **is not supported.**

Explanation: The database manager does not support the SQL statement shown (**statement-type**). Examples of **statement-type** are DECLARE TABLE, or DECLARE STATEMENT. In other IBM SQL products, these statements cause code expansions to occur. Because the database manager does not support these statements, the preprocessors have not performed these code expansions. Your program is preprocessed successfully unless other errors occur, and you did not specify the ERROR option as a preprocessing parameter.

SQLSTATE Spaces.

System Action: Preprocessing continues.

User Response: If your program depends on the code expansion, you should change your application. It should not depend on any of the above statements if you want to preprocess your application using the database manager.

+149 **The view cannot be used to modify data because it is based on more than one table.**

Explanation: INSERT, DELETE, or UPDATE of a view is not permitted if the VIEW is based on more than one table. This code is returned only when a statement is being preprocessed.

SQLSTATE 01574.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, check how many tables the VIEW is based on before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the tables that the VIEW is based on have been checked.

+150 **Connect authority is still active for user IDs.**

Explanation: This message is a reminder that revoking a user ID's DBA, SCHEDULE, or RESOURCE authority does not remove the user ID's CONNECT authority.

SQLSTATE 01512.

System Action: Normal completion.

User Response: None required. If you want to remove the CONNECT authority from the user IDs, enter a REVOKE CONNECT statement for the user IDs.

+151 **A column of a view cannot be updated since it is derived from an expression.**

Explanation: A column of a view may be updated only if it is derived directly from a column of a base table. This code is returned only when a statement is being preprocessed.

SQLSTATE 01575.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, check the origin of the column of a view before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the origin of the column of a view has been checked.

+154 **View limitations do not allow one of the operations.**

Explanation: Certain limitations exist when creating views:

- INSERT, UPDATE, or DELETE of a view is not permitted if the view involves a GROUP-BY CLAUSE or the keyword DISTINCT.
- The DELETE statement is not permitted in views defined with column functions.
- An index cannot be created on a view.
- ALTER TABLE is not permitted on a view.
- No host input variables are allowed for views.
- No cursor references are allowed in views.
- Inserts are not permitted in views whose columns are derived from expressions and not directly from a column of a stored table.

- The UNION operator is not allowed in a view.

Note: This code is returned only when a statement is being preprocessed.

SQLSTATEs:

01576 The bind process detected a CREATE INDEX statement for a view.

01577 The bind process detected a CREATE VIEW statement that includes an operator or operand that is not valid for views.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the correct number of data items exist.

+179 INCLUDE SQLCA is not permitted when you request NOSQLCA support.

Explanation: When preprocessing an application program with the NOSQLCA support, you cannot include the SQLCA structure in the source program.

SQLSTATE Spaces.

System Action: Processing of the SQL statement has ended. The SQLCA code is not generated into your program.

User Response: Remove the INCLUDE SQLCA statement.

+180 The string representation of a datetime value has invalid syntax.

Explanation: The string representation of a date or time value does not conform to the syntax of any supported format. This code is returned only when a statement is being preprocessed.

SQLSTATE 01572.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, check that the date or time value conforms to the syntax of a supported format such as ISO, JIS, EUR, USA, or LOCAL before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the date or time value conforms to the syntax of a supported format such as ISO, JIS, EUR, USA, or LOCAL.

+200 Committing the current logical unit of work revokes all existing run authority, except the creator's, for package *package-name***.**

Explanation: You used the Extended Dynamic PREPARE statement against a package that was created by using the CREATE PACKAGE statement with the MODIFY option. The current SQL statement has been added to the named package. When the logical unit of work is committed, all existing RUN authority for the named package, except its owner, is revoked. This occurs because the authorization required for the statement that has just been added is not sufficient to maintain the existing RUN authority with the GRANT option as it applies to the modified package.

SQLSTATE 01513.

System Action: Processing of the SQL statement was successfully ended.

User Response: If you do not want existing RUN authority to be revoked, roll back the current logical unit of work. To prevent this situation from occurring again, you must acquire the authority needed to execute the current SQL statement.

+202 Column *column-name* **was not found in any table referenced by the statement.**

Explanation: Since the column name was unqualified, all the columns associated with the tables or views in the SQL statement were searched and **column-name** was not among them.

This SQLCODE may result if:

- You use the single quotation character (') to mark the beginning and ending of a character string constant, instead of using the double quotation character (") required by SQL. For example, the following query generates SQLCODE +202 because the system attempts to use "STORED QUERIES" as a column name, not a character string constant:

```
SELECT * FROM SYSTEM.SYSCATALOG
WHERE TNAME='STORED QUERIES'
```

This query should be:

```
SELECT * FROM SYSTEM.SYSCATALOG
WHERE TNAME="STORED QUERIES"
```

- NOFOR support was invoked and the **column-name** in the appended FOR UPDATE OF clause was not found in the associated tables or views.
- The program is being preprocessed with the NOEXIST option.

SQLSTATE 01533.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the program is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, either create the **column-name** before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the **column-name** is created.

+204 *owner.object-name* **was not found in the system catalog.**

Explanation: The name (**owner.object-name**) specified in the message text is incorrect. This code is only returned when the program is being preprocessed with the NOEXIST option.

SQLSTATE 01532.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the program is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, either create the required object before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the required object is created.

If the *object-name* is one of the EXPLAIN tables, you can use the IBM-supplied macro (ARISEXP MACRO) to create the EXPLAIN tables. See the *DB2 Server for VSE & VM SQL Reference* manual for details about the ARISEXP MACRO file. The implicit EXPLAIN version of this SQLCODE is -649.

+205 **Column** *column-name* **was not found in table** *owner.table-name***.**

Explanation: The PRIMARY KEY, FOREIGN KEY, or UNIQUE CONSTRAINT contained an invalid column name **column-name**. This code is only returned when the program is being preprocessed with the NOEXIST option.

SQLSTATE 01533.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the program is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, either create the column before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the column is created.

+206 **The** *key-type constraint-name* **on** *owner.table-name* **was not found.**

Explanation: **Key-type** can be a PRIMARY KEY, FOREIGN KEY, or UNIQUE CONSTRAINT. For a primary key, **constraint-name** is omitted.

The SQL statement containing a primary key, foreign key constraint, or unique constraint was not found for the named table. Either you typed the **constraint-name** incorrectly, or the key or constraint does not exist for the specified table. This code is only returned when the program is being preprocessed with the NOEXIST option.

SQLSTATE 01533.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package, and it will be rechecked when the program is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, either create the required key or constraint before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the

statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the required key or constraint is created.

+304 **You cannot assign a value with a data type of** *data-type1* **to a host variable because the value is not within the range of the host variable in position** *position-number* **with data type** *data-type2***.**

Explanation: A FETCH or SELECT into a host variable list or structure, position number **position-number** failed because the host variable having data type **data-type2** was not large enough to hold the retrieved value having data type **data-type1**. The FETCH or SELECT does not return the data for the indicated SELECT item. The indicator variable associated with the host variable in error has been assigned a value of -2 to indicate the return of a NULL value.

SQLSTATE 01515.

System Action: Processing of the SQL statement continues.

User Response: Correct the output host variable so that it is large enough to contain the source data value. You must repeat the steps from preprocessing to execution of the program.

+331 **A null value has been assigned to a nullable host variable because at least one of the characters in the source value cannot be converted.**

Explanation: A warning occurred while converting a string value returned by the application server to the application requester. The value is incompatible with the nullable host variable because of a mismatch in data representation. The FETCH or SELECT does not return the data for the indicated SELECT-list item, the indicator variable is set to -2 to indicate the return of a NULL value, and processing continues.

SQLSTATE 01520.

System Action: Processing of the SQL statement continues.

+401 **Incompatible data types were found in an expression or compare operation.**

Explanation: The expression or compare operation violated one of the following rules:

- An arithmetic expression that does not involve datetime data types can only have operands with numeric data types.
- A datetime arithmetic expression cannot contain elements with graphic data types.
- Comparison operations can only be performed on compatible data types. The data types must both be numeric, character, graphic, date, time, or timestamp, or one must be a datetime data type and the other a character.

Note: This code is returned only when a statement is being preprocessed.

SQLSTATE 01578.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, correct the data types used before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the data types have been corrected.

+404 **A character string specified in an INSERT or UPDATE statement is too large for the target column.**

Explanation: A character string in the data items-list of an INSERT statement or one of the character strings in the SET clause of an UPDATE statement is longer than the associated target column in the database.

Note: This code is returned only when a statement is being preprocessed.

SQLSTATE 01570.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, correct the character string used before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the character string has been corrected.

+405 **The numeric value, *value*, is not within the range of the data type.**

Explanation: A numeric value specified in the statement exceeds the value allowed for the associated target column in the database. Following are the limitations:

- SMALLINT: -32768 to +32767
- INTEGER: -2147483648 to +2147483647
- FLOAT: -7.2E75 to -5.4E-79, 0, +5.4E-79 to +7.2E75
- DECIMAL: precision larger than that of the target column.

Note: This code is returned only when a statement is being preprocessed.

SQLSTATE 01579.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, correct the numeric value used before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the numeric values have been corrected.

+407 **Either an UPDATE statement or an INSERT statement with a null value for a column defined as NOT NULL is not allowed, or a null host variable value is not allowed in a SELECT list.**

Explanation: An attempt was made to update a column with a null value and the column does not allow nulls. The null value may be from a constant value, a host variable, an expression, or a value returned on a subquery.

This SQLCODE will also be issued if you made an attempt to SELECT a host variable in a select-list, and the select-list had an indicator variable value of <0, indicating a NULL host variable value. A NULL host variable value is not allowed in a select-list. For more information about using host indicator variables, refer to the *DB2 Server for VSE & VM Application Programming* manual.

Note: This code is returned only when a statement is being preprocessed.

SQLSTATE 01573.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the statement has been modified.

+408 **An UPDATE or INSERT of a data value is incompatible with the data type of the associated target column.**

Explanation: A source value was incompatible with the associated target column. This can happen when the source value was derived from:

- A subquery used in an UPDATE or INSERT statement.
- A VALUES clause in an INSERT statement.

Typical situations are:

- The source value has a numeric data type, and the target column has a character, DBCS, or datetime data type.
- The source value has a character data type, and the target column has a numeric or DBCS data type.
- The source value has a DBCS data type, and the target column has a numeric, character, or datetime data type.
- The source value has a character data type that is too short for the target datetime column.
- The source value has a datetime data type, and the target column has a numeric, DBCS, or different datetime data type.

Note: This code is returned only when a statement is being preprocessed.

SQLSTATE 01580.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, correct the command before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the

statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the command has been corrected.

+414 **The LIKE predicate was used for a numeric or datetime column type. LIKE must only be used with graphic or character columns.**

Explanation: The SQL statement uses LIKE against a numeric or datetime column. It can only be used against columns that have a data type that is compatible with the graphic data type or that have a character data type. This code is returned only when a statement is being preprocessed.

SQLSTATE 01571.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, correct the predicate used before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the predicate has been corrected.

+415 **The corresponding columns, *column-number*, of the operands of a UNION or a UNION ALL do not have comparable column descriptions.**

Explanation: The SQL statement contains SELECT clauses connected by a UNION or UNION ALL, and the corresponding columns in each SELECT clause cannot be compared. Specifically, columns of ordinality **column-number** of the operands in this UNION or UNION ALL cannot be compared. For columns to be comparable, corresponding columns must be numeric, character, graphic, date, time, or timestamp. Do not mix these data types. If corresponding columns have field procedures, both must have the same field procedure, and the same column CCSID value.

Note the following default data types for constants in a SELECT list:

- Alphanumeric constant: default data type is varying character
- All numeric constant (no E or decimal point): default data type is integer.

Note: This code is returned only when a statement is being preprocessed.

SQLSTATE 01581.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, correct the columns before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the columns have been corrected.

+416 **You cannot specify a long string column in the SELECT clause of a UNION.**

Explanation: The SQL statement contained a SELECT clause item whose data type is a long string. This is not valid.

A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(*n*) where *n* is greater than 254 but less than or equal to 32767, or VARGRAPHIC(*n*) where *n* is greater than 127 but less than or equal to 16383.

Note: This code is returned only when a program is preprocessed.

SQLSTATE 01582.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, correct the long string column before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the long string column has been corrected.

+419 **The precision of the numerator and/or the scale of the denominator are too large for decimal division.**

Explanation: The internal formula used to calculate the resulting scale of decimal division is:

```
Scale of result = 31 - np + ns - ds

Where  np = precision of numerator,
       ns = scale of numerator,
       ds = scale of denominator.
```

A negative resulting scale will cause incorrect results due to internal decimal constraints. This may also happen when you use the AVG() function.

Note: This code is returned only when a program is preprocessed.

SQLSTATE 01583.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the statement has been corrected.

+464 **FORMAT 1**
Procedure *proc-name* **returned** *num* **query result sets, which exceeds the defined limit** *integer*
FORMAT 2
Procedure *proc-name* **with AUTHID** *authid* **returned** *num* **query result sets, which exceeds the defined limit** *integer*

Explanation: The stored procedure named by *proc-name* completed normally. However, the stored procedure exceeded the defined limit on the number of query result sets the procedure can return.

num identifies the number of query result sets returned by the stored procedure.

integer identifies the defined limit on the number of query result sets for the stored procedure

Only the first *integer* query result sets are returned to the application that issued the SQL CALL statement.

The possible causes are as follows:

- The stored procedure is unable to return *num* result sets due to the limit established in the RESULTSETS column of SYSTEM.SYSROUTINES.
- The stored procedure is unable to return *num* result sets due to the DRDA limitations imposed by the client. The DRDA client establishes this limit with the MAXRSLCNT DDM code point.

SQLSTATE: 01609

System Action: The SQL statement is successful. The SQLWARN9 field is set to 'Z'.

+466 **FORMAT 1**
Procedure *proc-name* **returned** *num* **query result sets.**
FORMAT 2
Procedure *proc-name* **returned** *num* **query result sets for AUTHID** *authid.*

Explanation: The stored procedure referenced completed normally. The procedure returned the number of query result sets specified in *num*.

SQLSTATE: 01610

System Action: The SQL statement is successful. The SQLWARN9 field is set to 'Z'.

+494 **The number of result sets is greater than the number of locators.**

Explanation: The number of result set locators specified on the ALLOCATE LOCATORS statement is less than the number of result sets returned by the stored procedure. The first "n" result set locator values are returned, where "n" is the number of result set locator variables specified on the SQL statement.

SQLSTATE: 01614

System Action: The SQL statement is successful.

User Response: Increase the number of result set locator variables specified on the SQL statement.

+541 *key-type name* **contains the same column definitions as an existing** *entity***.**

Explanation: The **key-type** is PRIMARY KEY, FOREIGN KEY, or UNIQUE CONSTRAINT.

name is the FOREIGN KEY or UNIQUE CONSTRAINT key name that either the user specified or the system generated. For PRIMARY KEY, **name** is omitted.

entity can be PRIMARY KEY, FOREIGN KEY, or UNIQUE CONSTRAINT.

You receive a referential constraint warning because the FOREIGN KEY clause defines the same column list and parent table as another referential constraint. You receive a UNIQUE CONSTRAINT warning because the unique constraint defines the same column list as an

existing UNIQUE CONSTRAINT or PRIMARY KEY. You receive a PRIMARY KEY warning because the PRIMARY KEY clause defines the same column list as an existing UNIQUE CONSTRAINT.

SQLSTATE 01543.

System Action: Processing of the SQL statement ends.

User Response: If you want to define a PRIMARY KEY, drop the UNIQUE CONSTRAINT that causes the duplicate and define the PRIMARY KEY again. For the other situations, you do not have to define the FOREIGN KEY or UNIQUE CONSTRAINT again because a similar **entity** already exists.

+551 **User** *user-id* **does not have the** *privilege-type* **privilege.**

Explanation: An attempt was made to reference an object that the user is not authorized to use, or the SQL statement needs specific authority that the user does not have (for example, RUN authority). This can occur if a user with DBA authority preprocesses a program, but because of the nature of the SQL statements in the program, the DBA does not have the authority to grant RUN authority to another user. For more information, refer to the *DB2 Server for VSE & VM Application Programming* manual for information on defining privileges on packages.

When this SQLCODE results from an SQL statement in a program that an DB2 Server for VM preprocessor is processing, the statement is flagged in the DB2 Server for VM package for rechecking when the program is executed. Preprocessing of the program continues. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run.

When this SQLCODE results from GRANT ALL, the grantor has no privilege on the object table or view upon which he or she can grant privileges.

No error message is issued for GRANT ALL on a view if the grantor has at least one grant privilege on the view.

No error message is issued for GRANT ALL on a table as long as the grantor has at least one privilege.

Note: This code is returned only when the program is being preprocessed with the NOEXIST option.

SQLSTATE 01548.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package, and it will be rechecked when the program is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, acquire the required authorization for the object before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the required authorization for the object is created.

+552 *user-id* **is not authorized to perform this statement.**

Explanation: An attempt was made to reference an object that the user is not authorized to use, or the SQL statement needs specific authority that the user does not have.

Note: This code is returned only when the program is being preprocessed with the NOEXIST option.

SQLSTATE 01542.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package, and it will be rechecked when the program is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, create the required object before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once the required object is created.

+558 **The WITH GRANT option is ignored because the GRANT is PUBLIC.**

Explanation: Granting privileges to PUBLIC with the GRANT option does not give PUBLIC the privilege to grant to other users. Grant privileges cannot be passed to PUBLIC.

If PUBLIC is specified in a user list of GRANT ... WITH GRANT OPTION, PUBLIC receives the granted privileges without the grant option, while the other users receive the granted privileges with the GRANT option.

For example, as a result of the statement:

```
GRANT SELECT ON TABLE1 TO USER1, PUBLIC, USER2
WITH GRANT OPTION
```

USER1 and USER2 have the SELECT WITH GRANT privileges on TABLE1 while PUBLIC has only the select privilege.

SQLSTATE 01516.

System Action: The statement was executed successfully. The privileges are granted to PUBLIC without the GRANT option.

+600 **Automatic statistics collection was discontinued for the loading of table** *owner.table-name*.

Explanation: The Database Services Utility made a request to begin collecting statistics for table **owner.table-name** while adding rows to the table. After this request, the database manager detected that an index has been added to the table. The database manager does not support this form of statistics collection for tables that have indexes defined on them.

SQLSTATE Spaces.

System Action: The operation was executed successfully. The statistics accumulated thus far were discarded. The database manager does not accumulate statistics while the Database Services Utility loads rows into the specified table.

User Response: To get statistics for the specified table, the database manager must issue an UPDATE STATISTICS command for the table after the Database Services Utility has finished loading the table. If, instead, the Database Services Utility requests to finalize statistics collection for the table, this request fails because the database manager discarded the statistics.

+668 **Table** *owner.table-name* **is inactive and you cannot access it.**

Explanation: No operation can be performed on this table because at least one inactive key participates in a relationship with this table.

This may be a PRIMARY KEY, a FOREIGN KEY or a dependent FOREIGN KEY which references the PRIMARY KEY.

While an inactive key exists, only the table owner or a DBA can issue DML statements against the table. In addition, no one can issue DML statements against an active table which result in a reference to an inactive table in order to enforce Referential Constraints. This code is returned only when a statement is being preprocessed.

SQLSTATE 01585.

System Action: Processing of the SQL statement has ended. However, the statement has been recorded in the package and it will be rechecked when the package is executed. Preprocessing of the statement continues.

User Response: If this statement is not intended to be executed on the application server that returned the warning, the warning can be ignored. Otherwise, check that all the keys in which the table participates are active, before the statement is executed, or modify the statement and preprocess the program over again. If the condition is corrected before the statement is executed, execution occurs normally. However, if the condition persists at execution time, the execution fails. In either case, the statement is dynamically rebound each time the program is run. This dynamic rebinding can be avoided by preprocessing the program again, once all the keys in which the table participates have been checked.

+802 **Exception error** *exception-type* **has occurred during** *operation-type* **operation on** *data-type* **data, position** *position-number*.

Explanation: An arithmetic operation in the SQL statement has caused an exception condition. The arithmetic operation can be in the select-list of the SELECT statement, the search condition of a SELECT, UPDATE or DELETE statement, or the SET clause of an UPDATE statement. **exception-type**, **data-type**, **operation-type** and **position-number** may be returned in SQLCA, depending on where the error was detected.

If the user is using ISQL or DBS Utility, only the first occurrence of a +802 code on a query causes the message text to be displayed. If more +802s are encountered, no more messages will be displayed even when the error type and operation are different. The indicator variable associated with the host variable in error has been assigned a value of -2 to indicate the return of a NULL value. The associated data variable remains unchanged. The following is a list of **exception-type** values that can be tolerated by DB2 Server for VM:

- **FIXED POINT OVERFLOW** can occur during any arithmetic operation on either INTEGER or SMALLINT fields.
- **DECIMAL OVERFLOW** exception can occur when one or more non-zero digits are lost because the destination field in any decimal operation is too short to contain the result.
- **DIVIDE EXCEPTION** can occur on a division operation on any numeric field type (DECIMAL, FLOAT, SMALLINT, or INTEGER) by zero, or on a decimal division operation when the quotient exceeds the specified data-field size.
- **EXPONENT OVERFLOW** can occur when the result characteristic of any floating point operation exceeds the maximum value supported by System/370™ architecture (approximately 7.2E+75).

You can also get an **exponent underflow** when the result characteristic of any floating-point operation exceeds the minimum value supported by System/370 Architecture (approximately 5.4E-79).

SQLSTATE 01519.

System Action: Processing of the SQL statement continues.

User Response: Check the arithmetic operation and the columns within the command to determine what data may be causing the program exception. More information can be found in the *DB2 Server for VSE & VM Interactive SQL Guide and Reference* manual.

+825 **Operator command processing has been completed successfully.**

Explanation: The database operator command has completed successfully and all output has been returned.

SQLSTATE 01541.

System Action: Normal processing continues.

User Response: You may enter another database operator command or a database statement.

+863 **The connection was successful but only SBCS will be supported.**

Explanation: The application requester supports only the target application server's SBCS CCSID. Either the target application server's mixed CCSID or graphic CCSID (or both) is not supported by the application requester. Application requester character data sent to the target application server must be restricted to SBCS.

SQLSTATE 01539.

System Action: The CONNECT statement is successful.

User Response: Do not execute any SQL statements which use either mixed data or graphic data in the statement.

Note: You may pass mixed and graphic data in host variables.

SQLCODEs -084 through -30090

SQLCODEs -084 through -30090 are associated with error conditions. The SQLCODE descriptive text identified for these SQLCODEs is displayed as continuation lines for message ARI0503E.

-084 **Unacceptable SQL statement.**

Explanation: This SQL statement is unacceptable to DB2. One of the following has occurred:

- CALL statement cannot be the object of dynamic SQL statements : PREPARE, EXECUTE, or EXECUTE IMMEDIATE.
- An attempt was made to prepare an ALLOCATE CURSOR statement, but the statement identifier is already associated with a declared cursor.

SQLSTATE 42612.

System Action: Processing of the SQL statement has ended.

User Response: If the situation involves an invalid use of the SQL CALL statement, correct the statement. Try again.

If the situation involves an invalid PREPARE of an ALLOCATE CURSOR statement, change the application program to use a statement identifier that is not associated with a declared cursor.

-101 **SQL statement is greater than 8192 positions or is greater than an internal limitation of the system. Separate SQL statement into smaller statements.**

Explanation: The SQL statement has caused an internal limitation of the database manager to be exceeded. By examining the **SQLERRP** of the SQLCA, you can determine the nature of the exceeded limitation. See the following list:

SQLERRP	Explanation
ARIPBDL	Exceeded maximum SQL statement length of 8192 bytes when expanding a host structure reference.
ARIPPAR	Exceeded maximum SQL statement length of 8192 bytes when appending the FOR UPDATE OF clause.
ARIPSQA	Exceeded maximum SQL statement length of 8192 bytes.
ARIPSQB	Exceeded maximum SQL statement length of 8192 bytes.
ARIPSQC	Exceeded maximum SQL statement length of 8192 bytes.
ARIPSQP	Exceeded maximum SQL statement length of 8192 bytes.
ARITFST	Exceeded maximum SQL statement length of 8192 bytes.
ARIXECK	Exceeded maximum size of a key because of an internal sort.
ARIXEPP	The SQL statement contains an incorrect length. The length of the string should be from 0 to 8192.
ARIXESX	SQL statement is too complex. Exceeded database manager internal limitations.
ARIXIAF	Exceeded maximum number of 32767 foreign keys.
ARIXOB2	Exceeded maximum number of 40

| join columns in an SQL statement. The SQLERRD1 code for this first reason is -120 or -130.

Exceeded maximum number of 200 predicates in an SQL query block (WHERE or HAVING clause). The SQLERRD1 code for this second reason is -110.

ARIXOFC Exceeded maximum number of 255 column references in an SQL statement.

ARIXOFE Exceeded maximum number of 32 table references in an SQL statement. The maximum refers to data definition statements only (including FROM list, views, and temporary tables).

ARIXOFP Exceeded maximum number of 400 predicates in an SQL statement.

ARIXOFQ Exceeded maximum number of 16 queries in an SQL statement. Each view materialized is counted as a query.

ARIXOFT Exceeded maximum number of 15 tables in a FROM list.

ARIXOML Exceeded maximum number of 15 table references in an SQL statement (including FROM list, views, and temporary tables).

ARIXRST Exceeded maximum size of a key because of an internal sort.

If a SQLERRP is not shown in the preceding list, or SQLERRD1 = -199, the explanation is:

- The SQL statement is too complex, it exceeded SQL internal limitations.

When an SQL statement is found to be too complex, the complexity caused DB2 Server for VM internal storage or table limits to be exceeded. Possible causes for such complexity are if the command:

- Contains a number of operations, especially operations involving decimal or floating point numbers.
- Involves inserting a number of constants.
- Involves a number of functions, especially those involving arithmetic functions. Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column and scalar functions.
- Involves a number of complex views.

SQLSTATE 54001.

System Action: Processing of this SQL statement ends.

User Response: Either break the SQL statement into smaller statements or try to remove unnecessary conditions in the statement.

-102 Statement contains a string constant whose length exceeds 254 bytes. (An SBCS character counts as one byte and a DBCS character counts as two bytes. A hexadecimal constant is allowed a maximum of 254 numbers.)

Explanation: The SQL statement contains a string constant whose length exceeds the allowed maximum of 254 bytes. When determining the length of this string constant, it is necessary to count each EBCDIC character as one byte and each graphic character as two bytes. A hexadecimal constant has a maximum of 254 numbers, which result as 127 bytes. The beginning graphic delimiter (shift-out) and the ending graphic delimiter (shift-in) each count as one byte.

SQLSTATE 54002.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Decrease the length of the constant string and rerun.

-103 Statement contains the incorrect floating point number *number***.**

Explanation: A floating-point number did not end with a digit (exponent). For example, 12E is not correct, but 12E0 is correct.

SQLSTATE 42604.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-104 A portion of the SQL statement is incorrect following or involving *string***.**

Explanation: The portion of the SQL statement shown in *string* is incorrect.

Some of the possible causes are:

- **Format 2**:

  A portion of the SQL statement is incorrect following or involving DESCRIBE/ USING-descriptor.
- **Format 3**

  A portion of the SQL statement is incorrect following or involving missing SQL statement in PREPARE/EXEC.
- **Format 4**

A portion of the SQL statement is incorrect following or involving a literal, or a host variable is too long.

- **Format 5**

  A portion of the SQL statement is incorrect following or involving illegal format or length of SQL statement literal.
- **Format 6**: A portion of the SQL statement is incorrect following or involving a subquery in the SET clause.
- **Format 7**

  A portion of the SQL statement is incorrect following or involving the semicolon (;).
- **Format 8**

  A portion of the SQL statement is incorrect following or involving a missing semicolon (;).
- **Format 9**

  A portion of the SQL statement is incorrect following or involving an integer number that is larger than the maximum allowable number.

A **string** of either :H or a question mark (?) displayed in the message refers to a host variable found within the SQL statement.

SQLSTATEs:

42617 Statement string empty or blank.

42601 Everything else.

System Action: Processing of this SQL statement has ended.

User Response: Check the SQL statement for all errors. Correct the statement and resubmit it.

-105 **Either** *string1 string2* **or the text that follows is an incorrect sequence of words or symbols, or** *string2* **is an incorrect or reserved keyword.**

Explanation: The SQL statement does not follow the rules for the statement type. The symbols printed in the message show where the error was encountered. **string2** is not allowed to appear in the statement in the position where it was found. If either **string1** or **string2** was enclosed in double quotation marks ("), the double-quote characters do not appear in the message. If the symbol is a hexadecimal literal, it is not shown. (It is replaced by "X....".) If **string1** or **string2** appears as :H or a question mark (?), that string refers to a host variable, a field of a host structure, or an element of an indicator array found within the SQL statement. Strings of more than 34 characters may be truncated in the message.

SQLSTATE 42601.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-106 **SQL statement begins properly but is incomplete.**

Explanation: The SQL statement was correct up to the point where no more input was found. This may be caused by failure to properly end a literal. String literals need a final quote mark.

SQLSTATEs:

42603 Unterminated string constant.

42601 Anything else.

System Action: Processing of this SQL statement has ended.

User Response: Check that the statement has all the necessary parts to complete the desired function and that all clauses are complete.

For PL/I: Check that the SQL statement is complete before the semicolon.

For Assembler: Check that the continuation rules are properly followed. (A nonblank character must be in column 72, and continued lines must start in column 16 or beyond.)

For COBOL: Check that the SQL statement is complete before END-EXEC.

-107 **The name** *name* **is too long. Maximum allowable size is** *size***.**

Explanation:

One of the following has occurred:

- The SQL statement contains a name whose length exceeds the maximum size. When determining the length of this name, you must count each SBCS character as one byte and each DBCS character as two bytes. The beginning DBCS delimiter (shift-out) and the ending DBCS delimiter (shift-in) each count as one byte.
- NOFOR support was invoked and a name whose length exceeds the maximum size was found in the appended FOR UPDATE OF clause.

SQLSTATE 42622.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Shorten the length of the name and rerun.

-108 **The following user ID, password, server name, or host variable does not have the proper length:** *name***.**

Explanation: If *name* was an application program host variable, the declared length of *name* was not exactly 8 bytes (or 9 bytes in C) for the user ID or password, or was longer than 18 bytes (or 19 bytes in C) for a server name. If *name* was not a host variable, either the value specified exceeded 8 bytes for a user ID or password, or exceeded 18 bytes for a server name.

SQLSTATE 42622.

System Action: Processing of the SQL statement ends.

User Response: Correct the statement. Try again.

-109 **You must provide an INTO clause on the SELECT statement or the FETCH statement.**

Explanation: An INTO clause is needed for a SELECT statement if written in an application program. The INTO clause may appear in the SELECT statement itself, or in a FETCH statement if the SELECT statement was specified within a DECLARE CURSOR statement

SQLSTATE 42601.

System Action: Processing of this SQL statement has ended.

User Response: Supply an INTO clause and re-preprocess the program.

-110 **The following incorrect function was specified:** *function-type***.**

Explanation: The function specified is not known to the DB2 Server for VM product. Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column and scalar functions.

SQLSTATE 42601.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-111 **A SELECT, HAVING, or WHERE clause contains a column function without an associated column name.**

Explanation: A SELECT, HAVING, or WHERE clause contains a column function that should be associated with a column name. Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column and scalar functions.

SQLSTATE 42901.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-112 **A column function was used incorrectly. Either a column function is nested in a column function, or DISTINCT is used with an expression.**

Explanation: One of the following has occurred:

- The statement contains one of the column functions (AVG, MAX, MIN, COUNT, SUM) which contains another column function. You cannot nest column functions.
- If DISTINCT occurs inside a column function, DISTINCT must be associated with a column name, not an expression. For example, COUNT(DISTINCT QONHAND+QONORDER) is not permitted.
- If DISTINCT occurs inside a column function, the function can only have the unary operators (+, -) applied to it. For example, -COUNT(DISTINCT PARTNO) is permitted; COUNT(DISTINCT PARTNO) + 10 is not permitted.

Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column functions.

SQLSTATE 42607.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-114 **The server name portion of** *server-name1.creator.object* **does not match the name of the connected application server** *server-name2***.**

Explanation: The code is used to report that the server name part (standalone or qualifying a column) does not match the name of the connected application server. This code is returned either when the statement is being executed, or when the program is being preprocessed with the EXIST option.

SQLSTATE 42961.

System Action: Processing of the SQL statement has ended.

User Response: Correct **server-name2** or use the ERROR preprocessing option and try again. If the statement is an SQL CALL, an SQL CONNECT may need to be issued to the server *server-name1* before issuing the SQL CALL statement.

-115 **The comparison operator,** *operator***, was used incorrectly. Follow SOME, ANY, or ALL with a subselect. Follow IN or NOT IN with a subselect or a list.**

Explanation: One of the following has occurred:

- A comparison operator (=, ¬=, <>, <, >, <=, >=) was used with SOME, ANY, or ALL, but was not followed by a subselect.
- A comparison involved a parenthesized list but the list was not preceded by IN or NOT IN.

SQLSTATE 42601.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-116 **The length of the following label is greater than 30 bytes:** *label*.

Explanation: The length of the label for a column or table exceeds the maximum of 30 bytes. Only the first 30 bytes are displayed.

SQLSTATE 42622.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement and try it again.

-117 **The number of data values to be inserted does not equal the number of columns specified or implied.**

Explanation: The number of data values on an INSERT statement must equal the number of columns to be updated. Either too many or too few data values were listed for the columns specified or implied.

This SQLCODE can be produced during re-preprocessing of a package, when the number of columns that a table contained has been changed.

It can also be returned when either the statement is being executed, or when the program is being preprocessed with the EXIST option.

SQLSTATE 42802.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-118 **A table to be modified cannot be referenced in the subquery.**

Explanation: You cannot change (INSERT, DELETE or UPDATE) a table while naming it in the FROM clause of a SELECT (subquery) used in the same SQL statement.

SQLSTATE 42902.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-119 **An item in a SELECT clause or in a HAVING clause was neither a column function nor a column in the GROUP BY clause.**

Explanation: One of the following has occurred:

- The SELECT clause contains column functions which imply grouping. At least one of the items in the SELECT clause which is not a column function is not in the GROUP BY clause. The following is an example of an incorrect statement:

  ```
  SELECT DEPTNO, DEPTNAME, AVG(SAL)
  FROM EMPLOYEE GROUP BY DEPTNO
  ```

  It should be:

  ```
  SELECT DEPTNO, DEPTNAME, AVG(SAL)
  FROM EMPLOYEE GROUP BY DEPTNO,
  DEPTNAME
  ```

- The SQL statement contains a HAVING clause which includes a predicate which does not have a group-qualifying property. The HAVING clause is a search condition applied against each selected group. Therefore, you cannot use a column or a column function which is not a property of the selected group. The following is an example of an incorrect statement:

  ```
  SELECT DEPTNO, AVG(SAL)
  FROM EMPLOYEE GROUP BY DEPTNO
  HAVING SAL > 10000
  ```

  It should be:

  ```
  SELECT DEPTNO, AVG(SAL)
  FROM EMPLOYEE GROUP BY DEPTNO
  HAVING AVG(SAL) > 10000
  ```

  SAL > 10000 is incorrect because SAL is not part of the selected groups. Each group in the answer set only contains an AVG(SAL).

- The SQL statement contains a SELECT clause or column expression which contains a column function on one side of the expression operator and a column name on the other side. For example:

  ```
  AVG(AGE) + SENIORITY.
  ```

Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column and scalar functions.

SQLSTATE 42803.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-120 **The argument of a column function is not a correlated reference to a subselect which contains a HAVING clause, or a column function should not occur as the value to be assigned to a column in the SET clause of an UPDATE statement.**

Explanation: One of the following has occurred:

- The argument of a column function contains a correlated reference, but the correlated reference does not refer to a subselect that contains a HAVING clause.

The following are invalid queries:

```
SELECT ENAME FROM EMP
    WHERE AGE=MAX(AGE)
SELECT * FROM T0
   WHERE T0.B = (SELECT SUM(T1.D)
      FROM T1 GROUP BY T1.C HAVING
         SUM(T1.D) = (SELECT MIN(T2.F)
      FROM T2 GROUP BY T2.E HAVING
         MIN(T0.A) = T2.E))
SELECT * FROM T0
   WHERE T0.B = (SELECT SUM(T1.D)
      FROM T1 GROUP BY T1.C HAVING
         SUM(T1.D) = (SELECT T2.F
      FROM T2 WHERE
         MIN(T0.A) = T2.E))
```

The following are valid queries:

```
SELECT MAX(AGE) FROM EMP GROUP BY ENAME
HAVING MAX(AGE) > 1
SELECT SUM(T1.D) FROM T1 GROUP BY T1.C
HAVING SUM(T1.D) = (SELECT MIN(T2.F) FROM
   T2 GROUP BY T2.E HAVING
      MIN(T1.A) = T2.E)
SELECT SUM(T1.D) FROM T1 GROUP BY T1.C
HAVING SUM(T1.D) = (SELECT T2.F FROM T2
WHERE MIN(T1.A) = T2.E))
SELECT SUM(T0.B) FROM T0 GROUP BY T0.BB
HAVING SUM(T0.B) = (SELECT SUM(T1.D)
   FROM T1 GROUP BY T1.C
HAVING SUM(T1.D) = (SELECT MIN(T2.F)
   FROM T2 GROUP BY T2.E
HAVING MIN(T0.A) = T2.E))
SELECT SUM(T0.B) FROM T0 GROUP BY T0.BB
HAVING SUM(T0.B) = (SELECT SUM(T1.D)
   FROM T1 GROUP BY T1.C
HAVING SUM(T1.D) = (SELECT T2.F FROM T2
WHERE MIN(T0.A) = T2.E))
```

Column functions cannot be contained in the SET-CLAUSE for column updates. Thus, the following is not allowed:

```
UPDATE EMP SET SAL=MAX(SAL) (invalid)
```

Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column and scalar functions.

SQLSTATE 42903.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-121 **You cannot apply an INSERT or UPDATE to the same column more than once in the same statement.**

Explanation: An attempt was made to change the same column in a table more than once in the same statement.

SQLSTATE 42701.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-124 **More than 16 columns were used in an ORDER BY clause.**

Explanation: The maximum number of columns in an ORDER BY clause is 16.

SQLSTATE 54005.

System Action: Processing of the SQL statement has ended.

User Response: Decrease the number of columns in the ORDER BY clause to sixteen or fewer and resubmit the statement.

-125 **The statement contains ORDER BY** *column-number***. The ORDER BY clause does not correspond to a column in the SELECT clause.**

Explanation: The SQL statement contains an ORDER BY column which does not point to a valid column in the SELECT clause.

SQLSTATE 42805.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-126 **A column to be updated is specified in the ORDER BY clause.**

Explanation: The SQL query contains both a FOR UPDATE OF clause and an ORDER BY clause for the same column. The FOR UPDATE OF clause may have been appended when the NOFOR support was invoked.

SQLSTATE 42829.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-127 DISTINCT is specified more than once in a query or subquery.

Explanation: The SQL statement contains the keyword DISTINCT more than once in the SELECT clause or HAVING clause.

SQLSTATE 42905.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-128 The keyword, NULL, cannot be used as a data value in a WHERE clause.

Explanation: The NULL predicate was used as a data value in a data items list associated with a search condition predicate. This is not allowed. For example, the following statement is incorrect:

```
SELECT * FROM SAMPLE
WHERE PARTNO IN ('SMITH',NULL,'JONES')
```

It should be:

```
SELECT * FROM SAMPLE
WHERE PARTNO IN ('SMITH','JONES')
OR PARTNO IS NULL
```

SQLSTATE 42601.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-129 SQL statement contains too many *string***.**

Explanation: The **string** that may appear here is listed below along with the allowed associated limits:

- **Format 1**

  SQL statement contains too many input host variable names.

  ```
  INPUT HOST VARIABLE NAMES    - 4096
     (CHARACTERS)
  ```

- **Format 2**

  SQL statement contains too many output host variable names.

  ```
  OUTPUT HOST VARIABLE NAMES   - 4096
     (CHARACTERS)
  ```

- **Format 3**

  SQL statement contains too many input host variables.

  ```
  INPUT HOST VARIABLES         -  256
  ```

- **Format 4**

  SQL statement contains too many output host variables.

  ```
  OUTPUT HOST VARIABLES        -  256
  ```

- **Format 5**

  SQL statement contains too many table names.

  ```
  TABLE NAMES                  -   15
  ```

- **Format 6**

  SQL statement contains too many selected columns.

  ```
  SELECTED COLUMNS             -  255
  ```

- **Format 7**

  SQL statement contains too many cursor references.

  ```
  CURSOR REFERENCES            -  512
  ```

- **Format 8**

  SQL statement contains too many CCSID overrides.

  ```
  CCSID OVERRIDES              -   80
  ```

- Format 9

  SQL statement contains too many host identifiers.

  ```
  HOST IDENTIFIERS             -  512
  ```

Note: For Format 8, the 80 associated with CCSID OVERRIDES refers to the number of UNIQUE combinations of CCSID and data type. If you override the same data type with the same CCSID for several different host variables or table columns, this constitutes one override only.

SQLSTATEs:

54013	- Characters in host variables - Input host variables - Output host variables - Host identifiers.
54004	- Table names - Selected columns.
54019	- CCSID overrides

System Action: Processing of the SQL statement ends.

If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: For Formats 1 through 7, correct the statement and try again. For Format 8, the correct user response is determined by the cause of the error.

- This error can occur because you have overridden too many host variables in the SQLDA. Correct the situation by eliminating some of the overrides.
- This error can also occur if the columns which you have selected from the table contain too many unique combinations of CCSID and data type. Try breaking your statement into one or more queries each having no more than 80 CCSID overrides.

  To determine which columns have overrides, SELECT the CCSID column from the SYSTEM.SYSCOLUMNS catalog table for each column in your query. Those that differ from the application server defaults are overrides. The CCSID information for columns can also be obtained from the SQLDA returned by a DESCRIBE operation. For

more information on CCSIDs, see the *DB2 Server for VM System Administration* manual.

-130 **A subquery was not enclosed in parentheses.**

Explanation: A subquery in the SQL statement is not enclosed in parenthesis.

SQLSTATE 42601.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-131 **Either the LIKE predicate has an invalid escape character, or the string pattern contains an invalid occurrence of the escape character.**

Explanation: For "column-name LIKE pattern", the escape character can only appear in the character pattern if followed by itself, %, or _. You cannot use the escape character when the column is defined as mixed data, or the pattern is valid mixed data.

SQLSTATEs:.

22019 The LIKE predicate has an invalid escape character.

22025 The LIKE predicate string pattern contains an invalid occurrence of an escape character.

System Action: Processing of the SQL statement has ended.

Programmer Response: Correct the string pattern, or choose a different ESCAPE character and change the pattern accordingly.

-132 **LIKE or NOT LIKE is used incorrectly. You must precede it with a column name and follow it with either a quoted string, an input variable, or the special register USER.**

Explanation: LIKE or NOT LIKE is incorrectly specified. In **x [NOT] LIKE y**, **x** must be a column name and **y** must be in quotation marks.

If **x** appears in a WHERE clause when updating or deleting from a view, it must not be derived from an expression, a constant, or a scalar function in a VIEW definition.

SQLSTATE 42824.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-133 **You must use a column name, rather than an expression, as the argument of a column function when that column is a correlated reference column in a WHERE clause or HAVING clause.**

Explanation: The argument of a column function in a WHERE clause or HAVING clause cannot be an expression involving a correlated reference column; the argument must be a column only. The following example is incorrect:

```
SELECT DEPT,SUM(YEARS) FROM STAFF X
GROUP BY DEPT HAVING SUM(YEARS) >
ANY (SELECT AVG(YEARS) FROM STAFF
GROUP BY DEPT HAVING
MAX(YEARS) > MIN(1*X.YEARS))
```

The expression (1*X.YEARS) must not be the argument of the column function MIN. Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column and scalar functions.

SQLSTATE 42906.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-134 **Improper use of the long string column** *column-name***.**

Explanation: The SQL statement references a long string column improperly. The following items are not allowed:

- A CREATE INDEX statement
- A function other than SUBSTR or LENGTH
- A GROUP BY clause
- A PRIMARY KEY, FOREIGN KEY, or UNIQUE clause
- A SELECT DISTINCT statement
- A subselect of an INSERT
- A subselect of a UNION or UNION ALL
- An expression in a SET clause of an UPDATE
- An ORDER BY clause
- By itself in a predicate other than LIKE.

If it is none of the items listed above, it may be a long string column that is in a base table for a view that must be materialized in order to process the SQL statement. View materialization is explained in the *DB2 Server for VSE & VM Database Administration* manual.

A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(*n*) where *n* is greater than 254 but less than or equal to 32767, or VARGRAPHIC(*n*) where *n* is greater than 127 but less than or equal to 16383.

Another format for this code is:

Format 2

- Improper use of the long string input variable.

SQLSTATE 42907.

System Action: Processing of the SQL statement has ended.

User Response: The use of long strings is described in the *DB2 Server for VSE & VM Application Programming* manual. If you diagnose it as a view materialization restriction, define a view that does not include the long string column. Correct the statement. Try again.

-135 **The input for a long string column in an INSERT statement or UPDATE statement must be from a host variable or be the keyword NULL.**

Explanation: The UPDATE or INSERT is using constants, column names, or subqueries where it should be using NULL or a host variable.

A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(*n*) where *n* is greater than 254 but less than or equal to 32767, or VARGRAPHIC(*n*) where *n* is greater than 127 but less than or equal to 16383.

SQLSTATE 56033.

User Response: Refer to the *DB2 Server for VSE & VM Application Programming* manual for information on the use of long strings. Correct the statement. Try again.

-136 **A package name consisting of more than two parts is not supported.**

Explanation: A name consisting of more than two parts was used for the package name of a DROP PACKAGE or an extended dynamic SQL statement. This is not supported.

SQLSTATE 56048.

System Action: Processing of the SQL statement ends.

User Response: Correct the statement. Try again.

-137 **The length resulting from** *operation* **is greater than** *size***.**

Explanation: The specified scalar operation **operation** gave a result whose length was greater than the acceptable maximum, **size**.

SQLSTATE 54006.

System Action: Processing of the SQL statement has ended.

Programmer Response: Correct the statement. Try again.

-140 **The WITH CHECK OPTION cannot be used for the specified view.**

Explanation: The WITH CHECK OPTION cannot be used in a view definition under the following circumstances:

- The view is defined in such a way as to be read-only.
- The SELECT statement of the view contains a subselect.

SQLSTATE 42813.

System Action: The statement cannot be processed. The specified view was not created.

User Response: Refer to the *DB2 Server for VSE & VM SQL Reference* manual for rules regarding the use of the WITH CHECK OPTION in the view definition.

-141 **The INSERT or UPDATE is not allowed because a resulting row does not satisfy the view definition.**

Explanation: The WITH CHECK OPTION was specified in the definition of the view (or on at least one of the views on which this view is based), that is the object of the INSERT or UPDATE statement.Consequently, all attempts to insert or update in that view are checked to ensure that the results conform to the view definition.

SQLSTATE 44000.

System Action: The statement cannot be processed. No inserts or updates were performed, and the contents of the object view (and underlying base table) remain unchanged.

User Response: Examine the view definition to determine why the requested INSERT or UPDATE was rejected.

-144 **Invalid section number.**

Explanation: The specified section number is invalid for one of the following reasons:

- It exceeds the maximum section number supported by the database.
- The section number is already assigned to another valid SQL statement (no error detected) that is not related to the current SQL statement.
- The specified section number is less than zero.

SQLSTATE 58003.

System Action: The preprocessing of the program ends. No package is created.

User Response: Do not continue further processing. Notify your system programmer of this distribution protocol error.

-149 **The view cannot be used to modify data because it is based on more than one table.**

Explanation: INSERT, DELETE, or UPDATE of a view is not permitted if the VIEW is based on more than one table.

SQLSTATE 42807.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-151 **A column of a view cannot be updated because it is derived from an expression.**

Explanation: A column of a view may be updated only if derived directly from a column of a base table.

SQLSTATE 42808.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-153 **You must specify a COLUMN name-list in the CREATE VIEW statement because a view column name is derived from an expression or a built-in function.**

Explanation: When a view column is not derived directly from a column (for example, from an expression or function), a column name-list must be supplied in the CREATE VIEW statement.

Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column and scalar functions.

SQLSTATE 42908.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-154 **View limitations do not allow you to use the following operation:** *operation***.**

Explanation: Certain limitations exist when creating views:

- **Format 2:**

  View limitations do not allow you to use either of the following operations: GROUP-BY or DISTINCT.
- **Format 3**

  View limitations do not allow you to use the following operation: INPUT VARIABLES.
- **Format 4**

  View limitations do not allow you to use the following operation: CURSOR REFERENCE.
- **Format 5:**

  View limitations do not allow you to use the following operation: UNION in views.

SQLSTATEs:

42809 CREATE INDEX on a view

42909 Anything else.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-157 **View** *owner.view-name* **cannot be identified in a foreign key clause.**

Explanation: The indicated view was identified in a FOREIGN KEY clause of a CREATE or ALTER TABLE statement. A FOREIGN KEY clause must identify a table, not a view.

SQLSTATE 42810.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-158 **The CREATE VIEW statement failed. The number of entries in the COLUMN name-list is not equal to the number of items in the SELECT clause.**

Explanation: If a column name-list is specified on a CREATE VIEW statement, the number of column name-list entries (or view column names) must be equal to the number of columns in the SELECT clause defining the view. For example, the following command is not correct:

```
CREATE VIEW SAMPLE (SAMP1) AS
SELECT SUPPNO, PARTNO FROM QUOTATIONS
```

SQLSTATE 42811.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement so that the number of columns specified in the column name-list for the view is equal to the number of columns in the SELECT clause.

-159 **The following clause, keyword, or name was expected, but not found in the SQL statement:** *string***.**

Explanation: A required keyword was incorrectly specified or was omitted from the SQL statement. Some specific examples of this error are:

- **Format 1**:

The following clause, keyword, or name was expected, but not found in the SQL statement: IN clause.

- **Format 2**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: USING descriptor or FROM clause.

- **Format 3**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: USING descriptor.

- **Format 4**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: SETTING clause.

- **Format 5**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: SETTING or FOR clause.

- **Format 6**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: FROM clause.

- **Format 7**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: DESCRIPTOR keyword.

- **Format 8**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: SET QUERYNO.

- **Format 9**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: host variable or string constant in FROM clause.

- **Format 10**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: LANGUAGE.

- **Format 11**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: EXTERNAL.

- **Format 12**:

  The following clause, keyword, or name was expected, but not found in the SQL statement: SERVER GROUP.

SQLSTATE 42601.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Correct the statement to use the proper form of the PREPARE statement and rerun.

-160 **Use or value of *string1* is inconsistent with the use or value of *string2* in this statement.**

Explanation: The user specification included parts of an SQL statement that are not allowed to coexist with each other. Usually this is a result of using syntax that is an incorrect mixture of two forms of an SQL statement. Some specific examples of this condition are:

- **Format 1**:

  Use of FETCH is inconsistent with the use of non-SELECT statement in this statement.

  Note: The following is an example of where this can occur:

  ```
  stmt1='DROP TABLE FRED';
  PREPARE STMT1 FROM :stmt1;
  DECLARE CURSOR c1 CURSOR FOR STMT1;
  OPEN c1;
  FETCH c1 USING DESCRIPTOR :SQLDA;
  ```

- **Format 2**:

  Use of PUT is inconsistent with the use of non-INSERT statement in this statement.

  Note: PUT cannot be executed against any statement but an INSERT. This is similar to Format 1.

- **Format 3**:

  Use of the USING host variable clause is inconsistent with the use of DECLARE CURSOR FOR INSERT in this statement.

  Note: The USING clause cannot appear in the OPEN statement for an insert-cursor.

- **Format 4**:

  Use of OPEN with cursor name host variable is inconsistent with the use of USING host variable list in this statement.

  Note: An Extended OPEN must not use a host variable list (an SQLDA is needed).

  ```
  PREPARE s1 FROM 'SELECT * FROM t1
     WHERE c1=?';
  DECLARE c1 CURSOR FOR s1;
  OPEN :cur USING :hv1;
  ```

- **Format 5**:

  Use of OPEN with the cursor name host variable is inconsistent with the use of the input variable in DECLARE CURSOR FOR SELECT in this statement.

- **Format 6**:

  Use of statement name is inconsistent with the use of IN clause in this statement.

  Note: An IN package-tag clause is not supported in a non-Extended EXECUTE, a non-Extended

DECLARE CURSOR, or a non-Extended DESCRIBE statement. These statements are recognized as being Extended by virtue of the fact that package section is identified by a host variable.

- **Format 7**:

  Use of USING with the host variable list is inconsistent with the use of Extended EXECUTE in this statement.

  Note: An Extended EXECUTE must not use a host variable list (an SQLDA is needed).

- **Format 8**:

  Use of the USING clause or FROM clause is inconsistent with the use of the non-Dynamic PUT statement in this statement.

  Note: A PUT statement that is linked to a static INSERT statement may not have either a FROM clause or a USING clause.

- **Format 9**:

  Use of FETCH is inconsistent with the use of DECLARE CURSOR FOR INSERT in this statement.

  Note: This differs from Format 1 in that it is for static cursors, and is caught at program run time.

- **Format 10**:

  Use of FETCH with cursor name host variable is inconsistent with the use of INTO clause in this statement.

  Note: An Extended FETCH must not use a host variable list (an SQLDA is needed).

- **Format 11**:

  Use of cursor for SELECT is inconsistent with the use of cursor name host variable in this statement.

  Note: A static DECLARE CURSOR statement cannot exist with a host variable cursor name.

  ```
  DECLARE :cur CURSOR FOR SELECT * FROM t1;
  ```

- **Format 12**:

  Use of CURSOR FOR INSERT is inconsistent with the use of cursor name host variable in this statement.

  Note: A static DECLARE CURSOR cannot exist with a host variable cursor name.

  ```
  DECLARE :cur CURSOR FOR INSERT
     INTO t1 VALUES ('a', 'b');
  ```

- **Format 13**:

  Use of cursor name constant is inconsistent with the use of statement ID in this statement.

  Note: In an Extended DECLARE CURSOR, the cursor name must be a host variable.

- **Format 14**:

  Use of statement name is inconsistent with the use of cursor name host variable in this statement.

  Note: In an Extended DECLARE CURSOR a section-number host variable is required. A non-host variable statement name was provided instead.

- **Format 15**:

  Use of NEW is inconsistent with the use of KEEP/REVOKE in this statement.

  ```
  CREATE PROGRAM PACK1 USING OPTIONS NEW, KEEP
  ```

- **Format 16**:

  Use of EXECUTE is inconsistent with the use of SELECT in this statement.

  Note: This is raised when a dynamic EXECUTE is issued against a SELECT section. It is not issued when an Extended EXECUTE without an OUTPUT DESCRIPTOR is issued against a SELECT section.

- **Format 17**:

  Use of FROM NULL is inconsistent with the use of PREPARE SINGLE ROW in this statement.

  ```
  PREPARE SINGLE ROW FROM NULL
  ```

- **Format 18**:

  Use of FOR is inconsistent with the use of PREPARE SINGLE ROW in this statement.

  ```
  PREPARE SINGLE ROW FROM :hv1 FOR :hv2
  ```

- **Format 19**:

  Use of OUTPUT descriptor is inconsistent with the use of statement name in this statement.

  Note: This is issued when a non-Extended EXECUTE statement contains an OUTPUT DESCRIPTOR clause.

  ```
  EXECUTE s1 USING OUTPUT DESCRIPTOR SQLDA
  ```

- **Format 20**:

  Use of *value* is inconsistent with the use of EXPLAIN in this statement.

  ```
  value = ALL
          REFERENCE
          STRUCTURE
          COST
          PLAN
  ```

- **Format 21**:

  Use of PUT is inconsistent with the use of DECLARE CURSOR for SELECT in this statement.

  Note: Similar to Format 9.

- **Format 22**:

  Use of CHECK/ERROR is inconsistent with the use of MODIFY in this statement.

- **Format 23**:

Use of EXECUTE is inconsistent with the use of blocked INSERT in this statement.

- **Format 24**:

  Use or value of SERVER GROUP is inconsistent with the use or value of DEFAULT SERVER GROUP in this statement.

- **Format 25**:

  Use or value of COMMIT ON RETURN is inconsistent with the use or value of RESULT SETS in this statement.

SQLSTATEs:

24501 Format 1.

24503 Format 2.

07003 Format 16.

42601 Format 20.

42867 Format 15, Format 22, Format 24, and Format 25.

Spaces Format 3 through 14, Format 17 through 19, and Format 21.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Check the syntactical forms of the statement, correct the inconsistencies, and repreprocess the program or repeat the function.

-161 **You specified more than one alternative of the *option* option or else you specified the option more than once.**

Explanation: When options were offered, you chose more than one. You can select only one. Specifically, the following errors cause this message:

- In CREATE PACKAGE, both possibilities of an option pair were chosen or a combination of more than one possibility was chosen. For example, you cannot specify option pairs such as REPLACE/NEW and KEEP/REVOKE, or a combination like SBLOCK/BLOCK/NOBLOCK.
- Repeating an option may also cause this error condition. For example, in CREATE PACKAGE, options such as KEEP cannot be specified more than once.

SQLSTATE 42853.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Choose only one of the alternatives, or choose it only once. Repeat the run. Refer to the *DB2 Server for VSE & VM SQL Reference* manual for information about related options.

-162 **A *name-type* name is missing, is too long, or contains incorrect characters.**

Explanation: SQL names must start with an alphabetic character (A-Z) or national character (@, #, $) and may follow with these alphabetic characters, national characters, numbers (0-9), or underscore (_). Ordinary identifiers (table names, column names, dbspace names, etc.), cursor names, and statement names are limited to a length of 18 characters. Authorization names, passwords, and package names are limited to 8 characters. When authorization names or passwords are specified as host variables, they are fixed-length character variables of length 8 (or length 9 in C), padded on the right with blanks.

This message indicates a violation of the above rules for identifier contents or length, or the omission of an identifier of the indicated type.

Other formats of this code include:

- **Format 2**

  A cursor name is missing, is too long, or contains incorrect characters.

- **Format 3**

  Either a user ID or a package name is missing, is too long, or contains incorrect characters.

- **Format 4**

  Either a user ID or a package name is missing, is too long, or contains incorrect characters.

- **Format 5**

  A statement name is missing, is too long, or contains incorrect characters.

- **Format 6**

  A user ID is missing, is too long, or contains incorrect characters.

- **Format 7**

  A package name is missing, is too long, or contains incorrect characters.

- **Format 8**

  A qualifier or field name is missing, is too long, or contains incorrect characters.

SQLSTATEs:

42601 *name-type* missing

42622 *name-type* too long

42602 *name-type* incorrect.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this

error occurs in other than a preprocessing run, the function ends.

User Response: Correct the error in the specification of the identifier and repreprocess the program or repeat the function.

-163 A host variable is not allowed in this SQL statement.

Explanation: This particular SQL statement does not support the use of a host variable specification. In particular, the DROP PACKAGE statement permits a host variable name for the specification of the target program name, but other DROP statements require that you specify a constant for the name of the entity that is to be dropped.

SQLSTATE 42618.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-164 A graphic constant was not ended correctly.

Explanation: All graphic constants must be started and ended with the proper shift-out and shift-in characters. The SQL form of graphic constant is:

```
G'so...data...si'
or
N'so...data...si'
```

This form can be used in SQL statements, COBOL programs, and DBS Utility.

If this error occurred in a COBOL program or in DBS Utility, a graphic constant was started properly with G' so or N' so and the terminating si was found, but the si was not followed by an ending quote.

In a PL/I program, a graphic constant was started properly, but was not ended properly. The correct formats for a PL/I graphic constant are (format 1):

```
soX'427D'...data...X'427D42C7'si
```

or (format 2):

```
'so...data...si'G
or
'so...data...si'N
```

SQLSTATE 42604.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Correct the error in the specification of the constant and repreprocess the program or repeat the function.

-165 You cannot have leading or trailing blanks in a delimited identifier.

Explanation: You have entered a delimited identifier (".....") that contains either leading blanks, trailing blanks, or all blanks within the double quotes.

SQLSTATE 42602.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Correct the statement. Try again.

-166 A graphic constant began incorrectly.

Explanation: All graphic constants must be started and ended with the proper shift-out and shift-in characters. The SQL form of graphic constant is:

```
G'so...data...si'
or
N'so...data...si'
```

This form can be used in ISQL, COBOL programs, and DBS Utility. In these situations, a graphic constant was started with G', or with N', but a so character was not found following it.

In a PL/I program, a graphic constant was started improperly. For instance, the SQL form was used (G'so...data...si' or N'so...data...si'). The correct formats for a PL/I graphic constant are (format 1):

```
soX'427D'...data...X'427D42C7'si
```

or (format 2):

```
'so...data...si'G
or
'so...data...si'N
```

SQLSTATE 42604.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Correct the statement. Try again.

-167 **You specified either an odd number of hexadecimal characters or incorrect hexadecimal characters in a hexadecimal constant.**

Explanation: Characters other than 0–9 or A–F were found in a hexadecimal constant (for example, X'0F2AG4') where G is not permitted. This can also result from failure to provide an even number of hexadecimal characters (for example, X'0F429C6').

SQLSTATE 42606.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Correct the statement. Try again.

-168 **You specified an incorrect** *parameter* **parameter or parameter value.**

Explanation: The parameter or the value for the parameter named is not valid. For example, the value is above or below the value limit, or it is an incorrect value type.

SQLSTATE 42615.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Correct the statement. Try again.

-169 **Incorrect continuation of a graphic constant.**

Explanation: A graphic literal in an SQL statement must be contained on one line in a source program.

SQLSTATE Spaces.

System Action: Processing continues, but a new package will not be produced. If the package existed previously, it will be restored.

User Response: Correct the error in the continuation and repreprocess the program.

-170 **A graphic string either is too long, is improperly terminated, or contains bytes that are incorrectly paired for graphic strings.**

Explanation: All graphic strings must be started and ended with the proper shift-out and shift-in characters.

In addition to the proper shift-out and shift-in characters, a graphic string must have properly paired bytes, and not be too long.

The SQL form of a graphic string is:

```
G'so...data...si'
or
N'so...data...si'
```

This form can be used in ISQL, COBOL programs, and DBS Utility. In these situations, the termination sequence for a graphic constant (a si (X'0F') character followed by an apostrophe) was detected, but it either was not found in the proper byte positions or was incomplete following the graphic constant string. The terminating sequence should follow a pair of bytes. Instead it was found following a single byte or not found.

There are two formats for a PL/I graphic constant, (format 1):

```
soX'427D'..data...X'427D42C7'
```

or (format 2):

```
'so...data...si'G
or
'so...data...si'N
```

In a PL/I program, the following conditions in a graphic constant in an SQL statement can cause this error:

- A DBCS apostrophe within the DBCS constant was started with only one X'427D'. The correct form requires two: X'427D427D', for both formats.
- The beginning of the ending sequence of the DBCS constant (X'427D') was not followed by X'42C7' (format 1 only).

The correct formats for inserting a graphic apostrophe within a PL/I graphic constant are (format 1):

```
soX'427D'...X'427D427D'...data...
           X'427D42C7'si
```

or (format 2):

```
'so...X'427D427D'...data...si'G
or
'so...X'427D427D'...data...si'N
```

- After so (X'0E'), a si (X'0F') (both formats) was not found on an even byte boundary within the length limit (18 bytes for identifiers and 254 bytes for constants) in order to terminate graphic data.

If the graphic option is YES, so/si pairings are imposed on host variable and label names, and character constants in PL/I and COBOL. A violation of this rule causes SQLCODE -170.

The scan for proper termination continued to the end of the SQL statement or to the limit of 127 graphic characters for character string constants in dynamic

SQL statements, and to the limit of the line for the SQL statements imbedded in a host language program.

SQLSTATE 42604.

System Action: If this error occurs during DB2 Server for VM preprocessing, processing continues, but a new package is not produced, and no SQLSTATE is issued. If the package existed previously, it is restored. If this error occurs in other than a preprocessing run, the function ends.

User Response: Provide the correct termination of a graphic constant or decrease the size of a graphic constant. Try again.

-171 **The data type, length, or value of argument** *argument-number* **of** *function-type* **is invalid.**

Explanation: The above message is issued under the following circumstances:

function-type	argument number	reason
CHAR	1	DATE or TIME
	2	must be USA or ISO or EUR or JIS or LOCAL
CONCATENATE		not both character or both graphic
DATE	1	character string=7, timestamp, date or positive NUMBER
DAY	1	date, timestamp, DECIMAL(8,0), DECIMAL(20,6)
DAYS	1	date or timestamp
DECIMAL	1	not numeric
	2	less than 1 or greater than 31
	2	not a numeric constant
	3	less than 0 or greater than PRECISION
	3	not a numeric constant
DIGITS	1	not SMALLINT, INTEGER or DECIMAL
FLOAT	1	not numeric
HEX	1	exceeds 127 in length and >63 DBCS for graphic
HOUR	1	timestamp, time, DECIMAL(6,0), DECIMAL(20,6)
INTEGER	1	not numeric
MICROSECOND	1	timestamp, DECIMAL(20,6)
MINUTE	1	timestamp, time, DECIMAL(6,0), DECIMAL(20,6)
MONTH	1	date, timestamp, DECIMAL(8,0), DECIMAL(20,6)
SECOND	1	timestamp, time, DECIMAL(6,0), DECIMAL(20,6)
STRIP	1	not short character com- patible or graphic
	2	not L T or B (without quotes)
	3	not a character or graphic constant of length 1
	3	does not match datatype of STRING
SUBSTR	1	not character compatible or graphic
	2	less than 0 or greater than max_length(no decimal or float
	3	less than 0 greater than max_length - START + 1 (no numeric constant=zero
TIME	1	TIMESTAMP or TIME
TIMESTAMP	1	TIMESTAMP, CHAR STRING=8 or 14

function-type	argument number	reason
	2	TIME
TRANSLATE	1	not short character com- patible or graphic
	4	not a character constant of length 1
VALUE		incompatible types
VARGRAPHIC	1	not character or character compatible
	1	exceeds 127 characters
YEAR	1	date, timestamp, DECIMAL(8,0), DECIMAL(20,6)

Note: Long strings are only allowed for LENGTH and SUBSTR. See description for limitations.

SQLSTATEs:

22011 An argument of SUBSTR is out of range

42815 Anything else.

System Action: Processing of the SQL statement has ended.

Programmer Response: Correct the statement. Try again.

-172 **The number of arguments specified for *function-type* is invalid.**

Explanation: A SQL statement has specified the scalar function **function-type** with either too many or too few arguments. Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column and scalar functions.

SQLSTATE 42605.

System Action: The statement cannot be executed.

User Response: Correct the statement. Try again.

-173 **UR is specified on the WITH clause but the cursor is not read-only.**

Explanation: The cursor is not read-only. WITH UR can be specified only if the database manager can determine that the cursor is read-only.

SQLSTATE 42801.

System Action: Statement execution fails.

User Response: Change the isolation level specified on the WITH clause.

-180 **The string representation of a date/time value has invalid syntax.**

Explanation: The string representation of a date/time value does not conform to the syntax, or does not meet the minimum length requirements of any supported formats.

SQLSTATE 22007.

System Action: The statement cannot be executed.

User Response: Check that the datetime value conforms to the syntax of a supported format such as ISO, JIS, EUR, USA, or LOCAL. Correct the value where required.

-181 **The string representation of a date/time value is not a valid date/time value.**

Explanation: The string representation of a datetime value contains a value that is out of range.

The proper ranges for datetime values are as follows:

- 0001 to 9999 for years.
- 1 to 12 for months.
- 1 to 31 for days; except when the month is 4, 6, 9, or 11 when the proper range is 1 to 30 days; or when the month is 2 when the proper range is 1 to 28 days in a non-leap year or the proper range is 1 to 29 days in a leap year.
- 0 to 24 for hours. If the hour is 24, the other parts of the time, including the microsecond part of a TIMESTAMP value, are zero.
- 0 to 59 for minutes.
- 0 to 59 for seconds.

Note: The above ranges do not apply to LOCAL format values.

SQLSTATE 22007.

System Action: The statement cannot be executed.

User Response: Provide a value within the valid range. Try again.

-182 **An arithmetic expression with a date/time value is invalid.**

Explanation: The specified arithmetic expression contains an improperly used date/time value or labeled duration.

SQLSTATE 42816.

System Action: The statement cannot be executed.

User Response: Correct the indicated arithmetic expression. Try again.

-183 **An arithmetic operation on a date or timestamp has a result that is not within the valid range of dates.**

Explanation: The result of an arithmetic operation is a date or timestamp that is not within the valid range of dates, which are between '0001-01-01' and '9999-12-31'.

SQLSTATE 22008.

System Action: The statement cannot be executed.

User Response: Examine the SQL statement to determine the cause of the problem. The problem may be data-dependent, in which case you must examine the data that was processed at the time the error occurred.

-184 **Invalid parameter in a date/time arithmetic expression or scalar function argument list.**

Explanation: The specified arithmetic expression contains a parameter marker improperly used, or the parameter marker is used as a scalar function argument.

SQLSTATE 42610.

System Action: The statement cannot be executed.

User Response: Correct the indicated arithmetic expression. Try again.

-185 **The application uses the local format option with a date or time, but the local exit has not been installed.**

Explanation: The local format option has been used with a date or time value but no date or time exit has been installed. This error may occur if the LOCAL option is used with the CHAR function, or if the SYSOPTIONS table indicated that an exit for date or time was supplied, but in fact the exit has not been installed.

SQLSTATE 57008.

System Action: The statement cannot be executed.

User Response: Contact your system programmer about installation of the date or time exit.

-187 **Invalid use of labeled durations.**

Explanation: Labeled durations such as 10 MONTHS can only be used in a valid datetime arithmetic context. They are not to be used as stand-alone expressions such as COL1 > 5 DAYS.

Some examples that cause this SQLCODE to be issued are:

- COL1 = 3 DAYS + 4 MONTHS
- SELECT 10 YEARS FROM TABLE1
- DAYS(4 YEARS)

SQLSTATE 42816.

System Action: The statement cannot be executed.

User Response: Correct the statement. Try again.

-189 **An invalid CCSID value** *ccsid_value* **was specified.**

Explanation: The *ccsid_value* was invalid for one of the following reasons:
- Not in range of 1 to 65534
- Not compatible with specified subtype.
- Not supported by the DB2 Server for VM database management system.

SQLSTATE 22522

System Action: Processing of the SQL statement ends.

User Response: Correct the statement. Try again.

-191 **A mixed data value is improperly formed. Reason Code =** *reason-code***.**

Explanation: The mixed data contained an improperly formed DBCS portion. This could be a missing X'0E'(shift-out), or X'0F'(shift-in), or a stranded half byte of a DBCS character. The *reason code*s for this message are:

1. Conversion of the input statement.
2. Conversion of the input data, either constant or host variable.
3. Conversion of internal data during a column to column move.
4. Conversion of internal data during a column to column compare.
5. Conversion of internal data during a scalar function computation.
6. Conversion of internal data during datetime arithmetic with a character operand.
7. Conversion of output data to host variable.
8. Conversion by the application server when using the DRDA initialized SQLDS protocol (that is, the PROTOCOL parameter AUTO was specified, and the application requester is a DB2 Server for VM requester) or by the application requester when using the DRDA protocol.

SQLSTATE 22504.

System Action: Processing of the SQL statement ends.

User Response: Correct the data. Try again.

-192 *function-type* **scalar function does not allow the first argument to be of MIXED subtype.**

Explanation: You used an expression (column, host variable, constant, and so on) of MIXED subtype as the first argument of the STRIP or TRANSLATE scalar function. This is not allowed.

SQLSTATE 42937.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-198 **A null SQL statement has resulted.**

Explanation: Only the SQL statement delimiters were found. This results in a null SQL statement. The SQL statement start delimiter is EXEC SQL. The SQL statement end delimiter depends on the host language and can be one of the following:

- Assembler - absence of a continuation marker
- C, PL/I - semi-colon
- COBOL - END-EXEC

SQLSTATE Spaces.

System Action: The empty statement is marked as incorrect. Processing of the statement ends.

User Response: Check the program to find the empty statement and determine if any SQL statements are missing. Correct the program and resubmit it.

-201 **The following name was not found in the FROM clause:** *name***.**

Explanation: One of the following has occurred:

- Column reference table qualifier is not in the FROM clause.
- Column reference table qualifier is in the FROM clause but it is an unexpected table name.
- Correlation variable used in the SELECT clause is not in the FROM clause that follows.
- WHERE clause table name qualifier or correlation variable not found in a FROM clause. It may be any FROM clause that precedes it.
- ORDER BY clause table name qualifier or correlation variable is not found at the outer most query level.
- GROUP BY clause table name qualifier or correlation variable is not found at the outer most query level.

Note: If a table-name in a FROM clause is qualified by an owner and this table-name is used to qualify a column reference in a WHERE clause, ORDER BY clause, GROUP BY clause, or HAVING clause, the column reference qualifier must also include the owner along with the table-name. As an example, the following is incorrect:

```
SELECT DESCRIPTION FROM SQLDBA.INVENTORY,
SQLDBA.QUOTATIONS WHERE INVENTORY.PARTNO
= QUOTATIONS.PARTNO
```

It must be entered as:

```
SELECT DESCRIPTION FROM SQLDBA.INVENTORY,
SQLDBA.QUOTATIONS WHERE SQLDBA.INVENTORY.
PARTNO = SQLDBA.QUOTATIONS.PARTNO
```

Or, if you are the owner of the table, you can enter it as:

```
SELECT DESCRIPTION FROM INVENTORY,
QUOTATIONS WHERE INVENTORY.PARTNO
= QUOTATIONS.PARTNO
```

SQLSTATE 42703.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-202 **Column** *column-name* **was not found in any table referenced by the statement.**

Explanation: Since the column name **column-name** was unqualified, all the columns associated with the tables or views in the SQL statement were searched and **column-name** was not among them.

This SQLCODE may result if you use the double quotation character (") to mark the beginning and ending of a character string constant, instead of using the single quotation character (') required by SQL. For example, the following query generates SQLCODE -202 because the system attempts to use "STORED QUERIES" as a column name, not a character string constant:

```
SELECT * FROM SYSTEM.SYSCATALOG
WHERE TNAME="STORED QUERIES"
```

This query should be:

```
SELECT * FROM SYSTEM.SYSCATALOG
WHERE TNAME='STORED QUERIES'
```

Note: This code is returned when either the statement is being executed, or the program is being preprocessed with the EXIST option.

SQLSTATE 42703.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-203 **The unqualified column** *column-name* **was found in more than one table in the FROM clause.**

Explanation: The SQL statement contained a column name **column-name** which was not associated with a

specific table. The column name must be qualified so that it can be associated with a table in the FROM clause.

SQLSTATE 42702.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-204 *owner.object-name* **was not found in the system catalog.**

Explanation: The name *owner.object-name* specified in the message text is incorrect. This code is returned when either the statement is being executed, or the program is being preprocessed with the EXIST option. The *owner.object-name* referenced may not be the direct object name being manipulated. You may need to check the system catalog tables to identify the type of object missing.

SQLSTATEs:

42809 DROP VIEW table or DROP TABLE

42704 Anything else.

System Action: Processing of the SQL statement has ended.

User Response: Create the required object or provide the correct name of the object. Try again.

If the required object is one of the EXPLAIN tables, you can use a supplied macro job file (ARISEXP MACRO) to create the EXPLAIN tables. See the *DB2 Server for VSE & VM SQL Reference* manual for details about the macro job file (ARISEXP MACRO). The implicit EXPLAIN version of this SQLCODE is -649.

If the specified object is a stored procedure, use the CREATE PROCEDURE statement to define the stored procedure. If the specified object is a stored procedure server, use the CREATE PSERVER statement to define the stored procedure server. Once the stored procedure or stored procedure server has been defined, issue START PSERVER or START PROC to activate the new definition.

Note: If the CREATE statement has already been issued for the PROCEDURE, ensure that it has been committed. The CALL statement searches a cache for the procedure being called, and the cache entry is not made until the unit of work in which the CREATE statement was issued has been committed.

-205 **Column** *column* **was not found in table** *creator.table***.**

Explanation: The SQL command contained a column name **column** which cannot be found in the table specified in the message text.

SQLSTATE 42703.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-206 **The** *key-type constraint-name* **on** *owner.table-name* **was not found.**

Explanation: **Key-type** can be a PRIMARY KEY, FOREIGN KEY, or UNIQUE CONSTRAINT. For a primary key, **constraint-name** is omitted.

The SQL statement containing a primary key, foreign key constraint, or unique constraint was not found for the named table. Either you specified the **constraint-name** incorrectly, or the key or constraint does not exist for the specified table.

This code is returned when either the statement is being executed, or the program is being preprocessed with the EXIST option.

SQLSTATE 42704.

System Action: Processing of the SQL statement ends.

User Response: Create the required key or constraint, or provide the correct name for the desired key or constraint.

-207 **A column name is not allowed in an ORDER BY clause of a UNION.**

Explanation: The UNION involves an ORDER BY clause which contains a column name in the order-list. The order-list must contain only integers.

SQLSTATE 42706.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-208 **The column** *column-name* **was specified in the ORDER BY clause but does not appear in the SELECT clause.**

Explanation: The ORDER BY clause contains a column name **column** which is not in the SELECT clause.

SQLSTATE 42707.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-209 **A column name appears more than once in the SELECT clause, but is not qualified in the ORDER BY clause.**

Explanation: The column name in the ORDER BY clause should be qualified so that it can be associated with the appropriate column in the SELECT clause.

SQLSTATE 42702.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-210 **Incorrect symbol used in either a cursor name, owner name, package name, or statement name.**

Explanation: One of the following has occurred:

- A cursor name, owner name, program name, or statement name contained characters other than A-Z, 0-9, $, @, #, or underscore.
- In a non-extended dynamic statement a cursor name, owner name, program name, or statement name started with a character other than A-Z, $, @, or #.
- In an extended dynamic statement a cursor name, owner name, program name, or statement name started with a character other than A-Z, $, @, #, or 0-9.

SQLSTATE 42602.

System Action: Processing of the SQL statement has ended. A package is not produced if the error occurs during preprocessing within a statement that is not an extended dynamic statement.

User Response: When the error occurs in an extended dynamic statement, rollback the logical unit of work, correct the error causing the incorrect name, and rerun the preprocessor. Otherwise, just correct the name and rerun the preprocessor.

-211 **The table name or correlation name** *name* **appears more than once in the FROM clause.**

Explanation: One of the following conditions was found in the FROM clause of the SQL statement:

1. You repeated a table name (or view name) **name** without associated correlation names to distinguish each occurrence.
2. You repeated a correlation name **name**.
3. You have identical correlation names **name** and table names (or view names) in the same clause, and you repeated the name as a column qualifier.

SQLSTATE 42712.

System Action: Processing of the SQL statement ends.

User Response: The actions below correspond to the numbered conditions above:

1. Use a unique correlation name with each occurrence of the replicated table name (or view name).
2. Use distinct correlation names with the same clause.
3. Use a correlation name that differs from all table names (view names) in the same clause.

Use correlation names as column qualifiers.

-301 **The data type of a host variable or parameter is not compatible with the target data type.**

Explanation: It was found that the data type of the input host variable or parameter is not compatible with the target data type:

- If the target column or parameter is CHAR, VARCHAR, or LONG VARCHAR, the input host variable or parameter must also be character data.
- If the target column or parameter is GRAPHIC, VARGRAPHIC, or LONG VARGRAPHIC, the input host variable or parameter must also be graphic data.
- If the target column or parameter is INTEGER, SMALLINT, FLOAT, or DECIMAL, the input host variable or parameter must also be numeric data. If the host variable is DECIMAL, the data passed to the database manager at execution time in the host variable must be a number in decimal format.
- If the target column or parameter is DATE, TIME, or TIMESTAMP, the input host variable or parameter must be CHAR or VARCHAR.
- If the column in a LIKE predicate is CHAR, VARCHAR, or LONG VARCHAR, the ESCAPE character must also be character data. If the column in a LIKE predicate is GRAPHIC, VARGRAPHIC, or LONG VARGRAPHIC, the ESCAPE character must also be graphic data.

The problem typically occurs on an INSERT or UPDATE statement. It also occurs on an SQL CALL statement if a host variable is specified for a parameter, and the data type of the host variable is incompatible with the expected data type of the parameter. In this case:

- If the parameter is defined correctly in SYSTEM.SYSPARMS, you must modify the calling application to provide a host variable with a data type that is compatible with the parameter. You must then repeat the steps from preprocessing to execution of the calling application.
- If the parameter is defined incorrectly in SYSTEM.SYSPARMS, then you must:
  1. Issue the STOP PROC operator command with the REJECT option to stop the procedure
  2. Use the DROP PROCEDURE statement to remove the existing definition
  3. Use the CREATE PROCEDURE statement to redefine the procedure
  4. Issue the START PROC operator command to activiate the procedure

SQLSTATEs:

07006 if the statement is Dynamic

42895 if the statement is Static

System Action: Processing of the SQL statement has ended.

User Response: Check data type of column or parameter and correct the input host variable or parameter. You must repeat the steps from preprocessing to execution of the program.

-302 **An input host variable or parameter is invalid for the target column or parameter.**

Explanation: The problem typically occurs on an INSERT or UPDATE statement. It also occurs on an SQL CALL statement when a host variable is specified for an input parameter, and the data type of the host variable is incompatible with the expected data type of the parameter. In this case:

- If the parameter is defined correctly in SYSTEM.SYSPARMS, you must modify the calling application to provide a host variable with a data type that is compatible with the parameter. You must then repeat the steps from preprocessing to execution of the calling application.
- If the parameter is defined incorrectly in SYSTEM.SYSPARMS, then you must:
  1. Issue the STOP PROC operator command with the REJECT option to stop the procedure
  2. Use the DROP PROCEDURE statement to remove the existing definition
  3. Use the CREATE PROCEDURE statement to redefine the procedure
  4. Issue the START PROC operator command to activiate the procedure

If the error occurred on an INSERT or UPDATE statement, then either the data type and length of your input host variables are not compatible with the target column, or the data type and length assigned to the parameter markers are not compatible with the target column.

DB2 Server for VM assigns a data type and length to a parameter marker. It gets the data type and length contextually from the expression in which the parameter marker is used. As a result, a host variable can be too large if it exceeds the assumed length of the parameter marker for which it is being substituted. Or, in numeric data, a host variable can be too large if its numeric value exceeds the size that can be handled by the assumed data type (for example, greater than 32767 for SMALLINT).

If this SQLCODE is received after migration by an application that previously executed successfully, refer to the appendixes in the *DB2 Server for VSE & VM SQL Reference* manual about the incompatibilities between the DB2 Server for VM and SQL/DS releases of software for more information about the problem. (For example, as of SQL/DS Version 3 Release 2, the precision of decimal variables is enforced and some applications that insert data into decimal columns may receive this error.)

This problem most typically occurs on INSERT or UPDATE where the data type and length assigned to the parameter marker is that of the column being inserted into or updated.

Any one of the following size conditions may have occurred for the assumed parameter marker data type:

- CHAR or VARCHAR (including VARCHAR >254), but the host variable contains a character string that is too long.
- GRAPHIC or VARGRAPHIC (including VARGRAPHIC >127), but the host variable contains a character string that is too long.
- INTEGER, but the host variable contains decimal or float data that is too large or too small for INTEGER.
- SMALLINT, but the host variable contains decimal, float, or integer data that is too large or too small for SMALLINT.
- DECIMAL, but the host variable contains float, decimal, integer, or small integer data that is too large or too small for the assumed DECIMAL precision or scale.
- FLOAT, but the host variable contains a float value that is too large or too small in value.

See the *DB2 Server for VSE & VM SQL Reference* manual for a description of numeric data type maximums and minimums.

This problem can also occur when the assumed data type of the parameter marker in the ESCAPE clause of a LIKE predicate is:

- CHAR or VARCHAR (including VARCHAR > 245), but the length of the ESCAPE character is not 1
- GRAPHIC or VARGRAPHIC (including VARGRAPHIC > 127), but the length of the ESCAPE character is not 2.

SQLSTATEs:

22001	Character or graphic string is too long for column.
22003	Numeric overflow or underflow.
22024	NUL missing from C NUL-terminated string with length greater than 1.

System Action: Processing of the SQL statement has ended.

User Response: Check the target column or parameter type and correct the host variable or input parameter so that the data will fit in the target column or parameter, or correct the length of the host variable or parameter. If an SQL statement is modified, you must repeat the steps from preprocessing to execution of the program.

-303 **The data type of the output host variable is not compatible with the data type of the fetched column or expression, or the returned value from an SQL CALL statement.**

Explanation: A SQL CALL, FETCH, or SELECT into the output host variable could not be performed. The data type of an output variable is not compatible with the data type of the fetched column, expression, or returned value from an SQL CALL. The data type of the output host variable and the data type of the fetched column, expression, or returned value must both be numeric, or both be character, or both be DBCS. If the data type of the fetched column, expression, or returned value is date, time, or timestamp, the data type of output host variable must be character, or the same data type as the fetched column, expression, or returned value from SQL CALL statement.

SQLSTATE 42806.

System Action: Processing of the SQL statement has ended.

User Response: Check output data type and correct output variable. If an SQL statement is modified, you must repeat the steps from preprocessing to execution of the program.

If this error is received on an SQL CALL statement, you may need to redefine the stored procedure as follows:
1. Issue the STOP PROC operator command with the REJECT option for the procedure
2. Use the DROP PROCEDURE statement to remove the existing definition
3. Use the CREATE PROCEDURE statement to redefine the procedure
4. Issue the START PROC operator command to activate the procedure.

-304 **A value with data type** *data-type1* **cannot be assigned to a host variable because the value is not within the range of the host variable in position** *position-number* **with data type** *data-type2***.**

Explanation: An SQL CALL, FETCH, or SELECT into a host variable, position number *position-number,* failed because the host variable with data type *data-type2* was not large enough to hold the retrieved value with data type *data-type1.* In the case of a datetime source data type, the datetime value was too large to fit in the character or datetime output host variable after conversion to character format. The SQL CALL, FETCH, or SELECT statement is not executed. No data is returned.

SQLSTATEs:

22001 Character or datetime value too large for host variable.

22003 Numeric overflow or underflow.

System Action: Processing of the SQL statement has ended.

User Response: Correct the output host variable so that it is large enough to contain the source data value. You must repeat the steps from preprocessing to execution of the program.

Note: If the error occurred in the select-list of an outer SELECT statement, indicator variables can be added to each of the expressions, to allow you to get values not in error, and retrieve subsequent rows. In that case, the +802 SQLCODE would be issued and the indicator VARIABLEs associated with the last VARIABLE in error has been assigned a value of -2.

-305 **You did not associate an indicator variable with an output host variable. An error has occurred because a NULL data value was fetched.**

Explanation: A null value has been fetched but the output host variable has provided no indicator variable.

SQLSTATE 22002.

System Action: Processing of the SQL statement has ended.

User Response: Declare an indicator variable and connect it to the output host variable. This will allow null values to be returned. You must repeat the steps from preprocessing to execution of the program.

-306 **You did not declare the host variable** *variable-name***.**

Explanation: The named host variable was not declared between the BEGIN DECLARE SECTION and the END DECLARE SECTION SQL statements, or an error occurred in the declaration of the host variable.

SQLSTATE Spaces.

System Action: Processing continues, but a new package will not be produced. If the package existed previously, it will be restored.

User Response: Add a declaration for the missing host variable in the host variable declare section. You must repeat the steps from preprocessing to execution of the program.

-307 **You have already declared host identifier** *ident-name***.**

Explanation: The host-identifier *ident-name* is declared more than once in a SQL DECLARE section in this program or subprogram unit. The host identifier must be the name of a host variable or indicator variable. In those languages that support host structures, the host

identifier can be a host structure or a host structure field. The indicator variable can be an indicator array.

SQLSTATE Spaces.

System Action: Processing continues, but a new package will not be produced. If the package existed previously, it will be restored.

User Response: Remove the duplicate declaration. Rerun the preprocessor.

-309 **You used a negative indicator variable in a predicate in an invalid context.**

Explanation: A predicate uses a negative indicator variable. Negative indicator variables are not allowed in predicates (except predicates using the "=" operator) of Dynamic SQL statements. Nor are they allowed in predicates (except predicates using the "=" operator) of Extended Dynamic SQL statements that have no input SQLDA in the PREPARE statement. See the *DB2 Server for VSE & VM Application Programming* manual for information about using negative indicator variables.

SQLSTATE 22512.

System Action: Processing of the SQL statement ends.

User Response: Correct the input host variable so that is does not have a negative indicator variable or change the SQL statement so that the negative indicator variable is used in a valid context. You must repeat the steps from preprocessing to execution of the program.

-311 **An invalid length was specified for an input host variable character string.**

Explanation: The value contained in the length portion of the variable length host variable is either negative or greater than the length of its data portion.

SQLSTATE 22501.

System Action: Processing of the SQL statement ends. If this error occurred during an explicit connect, the logical unit of work (if any) is rolled back and the resource adapter returns to the application with SQLCA fields SQLWARN0 and SQLWARN6 both set to S, indicating a serious system error.

User Response: If this SQLCODE was issued in response to a non-CONNECT statement, correct the value in the length portion of the host variable so that it is not negative and it is not greater than the length of its data portion. You must repeat the steps from processing the program through to running the program.

If this SQLCODE was issued in response to a CONNECT statement, only a CONNECT statement will be accepted. Check the length field of the host variable used for your database name. It should be ≥ 1 and ≤ 18.

-312 **An SQL statement containing host variables cannot be submitted by way of a PREPARE, EXECUTE IMMEDIATE, or DECLARE CURSOR statement.**

Explanation: Any SQL statement submitted on a PREPARE or EXECUTE IMMEDIATE statement must not contain input or output host variables. In addition, the SELECT portion of the DECLARE cursor statement must not contain input or output host variables.

SQLSTATE 42618.

System Action: Processing of the SQL statement has ended.

User Response: Replace the host variables with parameter markers. You must repeat the steps from preprocessing to execution of the program.

-313 **There is a mismatch between the number of host variables declared and the number of host variables required for the SQL statement.**

Explanation: The user has not supplied exactly as many input program host variables as needed by the SQL statement. SQLERRD1 contains further information about the error. The possible values are:

-100 Input program host variables are required and either no SQLDA has been provided or the number of variables is zero.

-110 The number of input program host variables is negative.

-120 The number of input program host variables does not equal the number required.

SQLSTATEs:

07001 The number of host variables is not the same as the number of parameter markers.

07004 The statement contains parameter markers but the USING clause is not specified.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement or define more input host variables. You must repeat the steps from preprocessing to execution of the program.

-314 **You declared a host variable incorrectly.**

Explanation: An error occurred while a DB2 Server for VM preprocessor was processing a host-identifier declaration. The host identifier must be the name of a host variable or indicator variable. In those languages that support host structures, the host identifier can be a host structure or host structure field. The indicator variable can be an indicator array.

SQLCODEs

- For the ASSEMBLER preprocessor, one of the following errors was found:

  SQLERRD1:

  -403 Only reserved names are found in a declaration statement and no variable name is supplied.

  -404 Constant too long for halfword variable.

  -405 No length value on character declaration.

  -406 A character other than a blank or quote was found after 'F' or 'D'.

  -407 Length value on fixed-length character string exceeds 254.

  -408 Length value on character string exceeds 32767.

  -409 No length value on decimal declaration.

  -410 Length value on decimal declaration exceeds 31.

  -411 Constant on decimal declaration exceeds 31 characters (excluding sign and decimal point).

  -412 A character other than a blank or quote was found after the precision (p) of a decimal declaration.

  -413 Precision of constant on a decimal declaration is greater than the precision (p) specified.

  -414 Unknown operand type.

  -415 Length value of 0 specified for character or decimal declaration.

- For the C preprocessor, one of the following errors was found:

  SQLERRD1:

  -402 Use of conflicting data types. Example: long short int Var1;

  -403 No name or only reserved names are found in a declaration statement. No name or only reserved names and no tags are found in a structure declaration.

  -404 Incorrect syntax in multiple declaration. Example:

  ```
  short aa,bb,,cc;
  ```

  -405 Value contains nonnumeric characters. Examples:

  ```
  SQLVARCHAR (lastname, 10-);
  projno     char[3e];
  ```

  -406 Scale holds more than 5 places or 0. Example:

  ```
  char var1[123456];
  char var1[0];
  ```

  -407 No closing bracket. Example:

  ```
  char var1[123 ;
  ```

  -408 Length of C NUL-terminated string exceeds 32767 EBCDIC characters. Example:

  ```
  char var1[32768];
  ```

  -409 A VARCHAR variable is declared as more than 32767 characters in length. Example:

  ```
  struct{
         short ll;
         char dd[35000];
         } Var1;
  ```

  -410 Invalid delimiter found in declaration. Examples:

  ```
  char empno[12]        - missing semi-colon
  short  projno actno;  - missing comma
  char   midinit-'E';   - dash instead of '='
  ```

  -411 Declaration missing right parenthesis. Example:

  ```
  long (var1;
  ```

  -412 Error in SQLVARCHAR declare.

  Examples:

  ```
  SQLVARCHAR ABC,35);   /* missing left parenthesis */
  SQLVARCHAR(ABC 35);   /* missing comma            */
  ```

  -413 Error in VARCHAR declaration.

  Examples:

  ```
  struct{
         short ll       /* missing semicolon        */
         char dd[200];
         } Var1;
  struct{
         short ll = 23; /* improper initialization  */
         char  dd[200];
         } Var1;
  ```

  -414 The declaration contains two consecutive 'struct' keywords.

  -415 Invalid structure declaration caused by one of the following:

  - The structure name is missing.
  - The right brace is missing.
  - There is a semicolon missing after the structure element or imbedded structure declaration.

  -416 More than one structure tag found. Example:

  ```
  struct employee auto {
                        short    empno;
                        SQLVARCHAR(firstname,12);
                        SQLVARCHAR(lastname,15);
                       } acct_emp;
  ```

  -417 Invalid data type in structure element declaration.

  -418 Invalid indicator array declaration caused by one of the following:

- The data type is not equivalent to the DB2 Server for VM data type small integer.
- The indicator array is declared as part of a structure.
- The number of occurrences is greater than 65535.
- The array is declared as a multiple occurrence structure.

-419 Indirection operator '*' being used in a VARCHAR or structure declaration.

-420 Indicator array or character string declared with multiple dimensions. Example:

```
short  indarr[5][5];
```

-421 Precision was not specified for decimal declaration. Example:

```
decimal Var1;
decimal() Var2;
```

-422 Precision specified in decimal declaration is greater than 2 digits long or has a value greater than 31. Example:

```
decimal(42) Var1;
decimal(123,3) Var2;
```

-423 Precision specified in decimal declaration is 0. Example:

```
decimal(0) Var1;
```

-424 Scale of decimal declaration is greater than its precision. Example:

```
decimal(5,8) Var1;
decimal(16,32) Var2;
```

-425 Precision and/or scale specified in decimal declaration cannot be an octal number. Example:

```
decimal(5,02) Var1;
decimal(05,2) Var2;
decimal(08,03) Var3;
```

-426 A result set locator variable has been declared incorrectly.

- For the COBOL preprocessor, one of the following errors was found:

  SQLERRD1:

  -402 Both the group or element name and the data type are missing.

  -403 Only reserved names are found in a declaration statement and no variable name is supplied.

  -404 VARCHAR or VARGRAPHIC length (first level 49) is declared as PIC S9(n)COMP., where n>4. The correct declaration is with 1<=n<=4.

  -405 VARCHAR or VARGRAPHIC second level 49 is not one of these forms:

```
PIC X(n)         1 <= n <= 32767
PIC G(n)         1 <= n <= 16383
```

  -406 The repetition factor (parenthesized value) is not preceded by 9 or X value that is to be repeated.

  Examples:

```
PIC (4)       should be PIC S9(4)
PIC S9(4)V(3) should be
                    PIC S9(4)V9(3)
```

  -407 Something other than 0 through 9 found:
  - in the parentheses in a PIC clause
  - in place of a level number
  - in the occurrence count of an indicator array.

  Examples:

```
01 NAME PIC X(2A0)
01 NAME PIC S9(-1)
```

  -408 Found more than one V (implied decimal place) in a PIC clause for a decimal declaration, or found one or more V in a PIC clause for an integer declaration. Examples:

```
PIC S9(10)V99V99 COMP-3.
PIC S9(4)V9      COMP.
PIC S9(9)V99     COMP.
```

  -409 Invalid character in a PIC clause. Examples:

```
PIC X(3)XXIXX
PIC Q(3)
```

  -410 Parenthesis left open in a PIC clause repetition factor. Examples:

```
PIC S9(10 COMP.
PIC S9(8)V9(4 COMPUTATIONAL-3.
```

  -411 The declaration is incomplete.

  -412 PIC clause does not start with X, S, G or N. Examples:

```
PIC Q(3).
PIC Z(4).
PIC (20).
```

  -413 A PIC clause beginning with S is required if COMPUTATIONAL or COMPUTATIONAL-3 is specified. Similarly, COMPUTATIONAL or COMPUTATIONAL-3 is required if a PIC clause beginning with S is specified. Examples:

```
PIC X(4) COMP.
PIC S9(4).
PIC X(4) COMP-3.
```

  -414 A variable with COMP specified is declared as: PIC S9(n) COMP where either n<1 or n>9. The valid values for n are: 1<=n<=9.

  Examples:

```
PIC S9(0) COMP
PIC S9(10) COMP
```

-415 No length for a DECIMAL PICTURE clause. Example:

```
PIC S COMPUTATIONAL-3.
```

-416 The precision of a DECIMAL variable is greater than 31.

-417 A variable has a PIC clause with 0 length. Example:

```
PIC X(0).
```

-418 The following COBOL DECLARE clauses are not permitted in host variable declarations:

```
... BLANK WHEN ZERO.
... SIGN...
... JUSTIFIED...
```

-419 A VARCHAR variable is declared as more than 32767 characters in length, or a VARGRAPHIC variable is declared as more than 16383 DBCS characters in length. Examples:

```
01 DESCRIPTION.
49 DESCRIPTION-LEN PIC S9(4) COMP.
49 DESCRIPTION-VAL PIC X(33000).
```

-420 A CHARACTER variable is declared as more than 254 characters in length, or a GRAPHIC variable is declared as more than 127 DBCS characters in length. Example:

```
01 NAME PIC X(300).
```

-421 The declaration is not correctly terminated. This error usually indicates a missing period in your DECLARE statement.

Examples:

```
Example 1.
  77 FIRST-NM   PIC X(10)    <- missing period
  77 LAST-NM    PIC X(10).   <- error reported
                             after this declaration
Example 2.
  01 DESCRIPTION.
     49 DESC-LEN   PIC S9(4) COMP <- missing       |
                                          period  |
     49 DESC-TXT   PIC X(100).    <- error
                                          reported
                             after this declaration
```

-422 Invalid structure declaration caused by one of the following:
- The structure does not begin with a level 01 group.
- The structure contains more than one level 01 group.
- The name of a subordinate group is FILLER. FILLER is only allowed as the name of elementary items within a structure. If the name of a level 01 group or an independent elementary item is FILLER, the entire declaration is ignored and no message is generated.
- A subordinate group or element name is missing.
- New level was found, but the previous level is not a group level.
- A group level was specified but no elements are specified for the group.
- A REDEFINES clause was found within the structure.

-423 Invalid indicator array declaration caused by one of the following:
- The data type is missing.
- The data type is not equivalent to the DB2 Server for VM data type small integer.
- The indicator array is declared as part of a structure containing other elements.
- The number of occurrences is greater than 65535.
- The level number of the array is 01 or 77. An indicator array must be an element of a level 01 group.
- The TIMES is missing from the OCCURS clause of an indicator array declaration.

-424 A level 77 element is declared within a structure, or as the level number of a major group.

-425 A level number is outside the range 01 to 49, 66, 77, and 88. Levels 66 and 88 are ignored.

-426 A literal string in the declaration statement contains an invalid DBCS string.

-427 A closing quotation mark is missing.

-428 The data type is missing from the declaration of an elementary item.

-429 There is an incomplete or invalid attribute specification, an attribute incorrectly placed in the declaration, or some other unknown string.

-430 A result set locator variable has been declared incorrectly.

- For the PL/I preprocessor, one of the following errors was found:

  SQLERRD1:

  -403 Missing parenthesis or parenthetical phrase in the value clause of an attribute specification.

  Examples:

```
DCL EMPNO     BINARY FIXED(31) INITIAL (2500;
DCL EMPNAME   CHAR (25) INITIAL;
```

  -404 Precision or scale specified in decimal declaration is greater than 2 digits long or has a value greater than 31.

-405 Scale of decimal declaration is greater than its precision.

-406 Value other than 15 or 31 found for binary fixed declarations.

-407 A fixed binary field was declared with a number other than zero specified for the number of places to the right of the binary point. For example, FIXED BINARY (15,0) is correct, but FIXED BINARY (15,3) is not correct.

-408 Length of BINARY FLOAT variable is greater than 53 or length of DECIMAL FLOAT variable is greater than 16.

-409 Length of character string exceeds 32767 EBCDIC characters or 16383 double-byte character set (DBCS) graphic characters.

-410 Length of fixed-length character string exceeds 254 EBCDIC characters or 127 DBCS graphic characters.

-411 A semicolon or a comma was found, but neither the variable name nor the attribute was specified.

-412 Incorrect syntax in variable list.

-413 FIXED found without corresponding BINARY or DECIMAL.

-414 BINARY found without corresponding FIXED or FLOAT.

-415 FLOAT found without corresponding BINARY or DECIMAL.

-416 DECIMAL found without corresponding FIXED or FLOAT.

-417 There is an incomplete or invalid attribute specification, an attribute incorrectly placed in the declaration, or some other unknown string.

Examples:

```
DCL EMPNO       BNARY FIXED (15);
                      - invalid attribute
DCL AUTOMATIC EMPNAME   CHAR(20);
                      - incorrect placement
DCL DEPTNAME    CHAR <10);
                      - unknown string <
DCL 01 EMPLOYEE  DEFINED,
       - incomplete attribute specification
        .
        .
```

-418 Value contains non-numeric characters. Examples:

```
DCL IND_ARRAY  (1-)  BINARY FIXED (15);
DCL EMPNAME    CHAR (1A);
```

-419 Length value of 0 specified for character or graphic string or numeric field.

-420 Level number missing. Only scalar variables may be declared without a level number.

-421 Invalid structure declaration caused by one of the following:
- The declaration has a duplicate level 1.
- The level number is greater than 65535.
- Data type attributes were found on a structure variable declaration.

-422 Invalid indicator array declaration caused by one of the following:
- The data type is missing.
- The data type is not equivalent to the DB2 Server for VM data type small integer.
- The indicator array is declared as part of a structure.
- The number of occurrences is greater than 65535.

-423 Factoring of structure names is not permitted.

-424 Datatype missing from a scalar variable or structure element declaration, or the name is missing, which causes the datatype to be taken as the variable name.

-425 A structure declaration does not begin with a level number of 1.

-426 A result set locator variable has been declared incorrectly.

SQLSTATE Spaces.

System Action: Processing has ended for this host variable.

User Response: Correct the statement. Try again.

-315 **The following host variable for user ID or password is not preceded by a colon in the SQL CONNECT statement:** *variable-name***.**

Explanation: User ID and password used in the CONNECT SQL statement should be declared as host variables. The names of those host variables in the CONNECT SQL statement should be preceded by a colon (:).

SQLSTATE Spaces.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-316 **The SQL statement prefix, EXEC SQL, was found in COBOL section A (columns 8-11). EXEC SQL should not start before column 12.**

Explanation: EXEC SQL was coded beginning in Section A of a statement in a COBOL program (columns 8-11). EXEC SQL should not start before column 12.

SQLSTATE Spaces.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-317 **Valid SQL statements in the WORKING-STORAGE SECTION are BEGIN DECLARE SECTION, END DECLARE SECTION, INCLUDE SQLCA, and INCLUDE.**

Explanation: The BEGIN or END DECLARE SECTION, the INCLUDE SQLCA or the INCLUDE command has been coded improperly. Following are probable causes:

- EXEC SQL not followed by BEGIN, END, or INCLUDE when appearing in the LINKAGE, FILE, or WORKING-STORAGE SECTION of COBOL
- More than one BEGIN DECLARE SECTION before an END DECLARE SECTION found
- An END DECLARE SECTION found before a BEGIN DECLARE SECTION
- Missing END-EXEC terminator or improper SQL statement in LINKAGE, FILE, or WORKING-STORAGE SECTION.
- PROCEDURE DIVISION is missing after the WORKING-STORAGE SECTION.

SQLSTATE Spaces.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-318 **INCLUDE SQLCA SQL statement is outside the WORKING-STORAGE SECTION.**

Explanation: An INCLUDE SQLCA statement must only be coded in the COBOL WORKING-STORAGE section.

SQLSTATE Spaces.

System Action: Processing of the SQL statement ends.

User Response: Move the statement. Try again.

-319 **More than one INCLUDE SQLCA statement was in the COBOL program.**

Explanation: Only one INCLUDE SQLCA statement can appear in a COBOL program.

SQLSTATE Spaces.

System Action: Processing of the SQL statement ends.

User Response: Remove the extra statement. Try again.

-320 **The SQL declare section did not end before the end of the containing COBOL section.**

Explanation: A BEGIN DECLARE SECTION statement was found, but a matching END DECLARE SECTION was not found before the containing WORKING-STORAGE SECTION, LINKAGE SECTION, or FILE SECTION ended.

SQLSTATE Spaces.

System Action: Processing of the SQL statement ends.

User Response: Insert an END DECLARE SECTION statement. Try again.

-321 **An SQL statement other than an INCLUDE statement appears in the declarative section of the COBOL PROCEDURE DIVISION.**

Explanation: An SQL statement other than an Extended INCLUDE statement cannot appear in the declarative section of the PROCEDURE DIVISION in COBOL.

SQLSTATE Spaces.

System Action: Processing of the SQL statement ends.

User Response: Move or remove the statement. Try again.

-322 **EXEC SQL was in a section other than the FILE, LINKAGE, WORKING-STORAGE SECTION, or PROCEDURE DIVISION sections.**

Explanation: An SQL statement (one beginning with EXEC SQL) must only appear in the WORKING-STORAGE, FILE, or LINKAGE sections of the DATA DIVISION or in the PROCEDURE DIVISION.

SQLSTATE Spaces.

System Action: Processing of the SQL statement ends.

User Response: Move or remove the statement. Try again.

-323 **A new SQL statement begins before the previous one properly ended. Both statements are ignored.**

Explanation: Missing END-EXEC following EXEC SQL. Another EXEC SQL was found before the previous END-EXEC was found.

SQLSTATE Spaces.

System Action: Processing of the SQL statement ends.

User Response: Correct the statements. Try again.

-324 **Host variable *variable* is declared with an attribute that is incorrect for its use in the SQL statement.**

Explanation: The SQL statement requires a host variable of a particular data type or length that was not found. If a variable number is specified, it corresponds to the variable's position in the SQL statement. Some specific examples of this error are:

- A locator variable passed in is not declared as integer or RESULT SET LOCATOR datatype.
- Statement-id is not an integer in one of the following extended dynamic commands: DROP STATEMENT, PREPARE, DECLARE CURSOR, DESCRIBE, or EXECUTE.
- Cursor name host variable is not character or is longer than 18.
- User ID, password, or package name host variable is not a fixed character string, or, in C, is not a NUL-terminated character string.
- User ID, password, or package name host variable is not 8 in length, or, in C, is not 9 in length.
- Options host variable on a CREATE PROGRAM/PACKAGE statement is not defined in varchar format or is longer than 8192.
- The server-name host variable is not a fixed or varying length character string with a length in the range of 1-18 or, in C, a C-NUL terminated character string with a length in the range of 2-19.
- The structure is not a valid host structure.
- Use of a host structure is not permitted.
- An indicator array is used in place of a host variable or main variable.

SQLSTATE Spaces.

System Action: Processing continues, but a new package will not be produced. If the package existed previously, it will be restored.

User Response: Change the host variable declaration to conform to the required type/length specification. Try again.

-325 **This preprocessor does not support the SQL statement.**

Explanation: The SQL statement used is only supported by the Assembler preprocessor of the database manager. The SQL statement was probably of the Extended Dynamic type.

SQLSTATE Spaces.

System Action: Processing continues, but a new package will not be produced. If the package existed previously, it will be restored.

User Response: Eliminate the failed statement either by moving it to an Assembler language program for preprocessing or by changing the statement to the Dynamic form as opposed to the Extended Dynamic form. Try again.

-326 **Indicator variable *variable-name* must be declared with a data type and length that equates to an SQL data type of SMALLINT.**

Explanation: The named indicator variable was not declared with a data type and length which equates to an SQL data type of SMALLINT.

System Action: Processing of the SQL statement has ended.

User Response: Correct the declaration of the indicator variable so that it has an SQL data type of SMALLINT. This information is in the *DB2 Server for VSE & VM Application Programming* manual.

SQLSTATE Spaces.

-330 **A string value cannot be used because at least one of its characters cannot be converted.**

Explanation: An error occurred when converting an application input string variable to the application server's representation. The value of the string host variable is incompatible with its use because of a mismatch in data representation. The value cannot be used as specified.

SQLSTATE 22021.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement.

-331 **A string value cannot be assigned to a non-nullable host variable because at least one of the characters in the source value cannot be converted.**

Explanation: An error occurred when converting a string value returned by the application server to the application requester. The string value cannot be

assigned to a host variable that does not have an indicator variable within a SELECT statement of an application program. The value is incompatible with the host variable because of a mismatch in data representation. The FETCH or SELECT statement is not executed. No data is returned.

SQLSTATE 22021.

System Action: Processing of the SQL statement has ended.

User Response: None.

-332 **A source string with a CCSID of *source-ccsid* cannot be assigned to its target with a CCSID of *target-ccsid* because of incompatible CCSIDs. Reason Code= *reason-code*.**

Explanation: No conversion supported for the specified pair of CCSIDs. The reason codes for this message are:

- 1 - Conversion of the input statement.
- 2 - Conversion of the data, which is either a constant or a host variable.
- 3 - Conversion of internal data during a column-to-column move.
- 4 - Conversion of internal data during a column-to-column compare.
- 5 - Conversion of internal data during a scalar function computation.
- 6 - Conversion of internal data during datetime arithmetic with a character operand.
- 7 - Conversion of output data to host variable.
- 8 - Conversion by the application server when using the DRDA initialized SQLDS protocol (that is, the PROTOCOL parameter AUTO was specified, and the application requester is a DB2 Server for VM requester) or by the application requester when using the DRDA protocol.
- 9 - Invalid source, or target CCSID of zero. This will occur if graphic or mixed data is used when the CCSIDMIXED or CCSIDGRAPHIC value is set to zero for the application server or the application requester. It will also occur if an input or output SQLDA is used to override the CCSID value, and this override CCSID value is zero. The CCSID values are set using the CHARNAME parameter. See the *DB2 Server for VM System Administration* manual for details.
- 10 - Invalid source, or target CCSID of zero. This will occur when a character column was created previous to SQL/DS Version 3 Release 3 with the FOR MIXED DATA clause and the MCCSIDMIXED value in SYSTEM.SYSOPTIONS is zero (or does not exist), or a graphic column was created previous to SQL/DS Version 3 Release 3 and the MCCSIDGRAPHIC value in SYSTEM.SYSOPTIONS is zero (or does not exist). See the *DB2 Server for VM System Administration* manual for information on how to insert or update the MCCSIDMIXED and MCCSIDGRAPHIC rows.

SQLSTATE 57017.

System Action: Processing of the SQL statement has ended.

User Response: Either correct the statement so that it does not move or compare data between incompatible CCSIDs, or add support for the specified conversion.

-334 **An ASCII Mixed string could not be converted because it would exceed the length attribute of its EBCDIC target.**

Explanation: The insertion of shift-out (X'0E') and shift-in (X'0F') characters to indicate the DBCS portion of a mixed string in EBCDIC has caused the converted string to be too large for the target.

SQLSTATE 22524.

System Action: Processing of the statement ends.

User Response: Correct the statement or increase the length of the EBCDIC target and try again.

-401 **Incompatible data types were in an expression or compare operation.**

Explanation: The expression or compare operation violated one of the following rules:

- An arithmetic expression which does not involve datetime data types can only have operands with numeric data types.
- A datetime arithmetic expression cannot contain elements with graphic data types.
- Comparison operations can only be performed on compatible data types. The data types must both be numeric, character, graphic, date, time, or timestamp, or one must be a datetime data type and the other a character data type.

SQLSTATE 42818.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-402 **You cannot use nonnumeric data in the AVG or SUM column functions or as part of an arithmetic operation.**

Explanation: One of the following has occurred:

- Character or DBCS data was specified as the argument for AVG or SUM column functions such as AVG(CHARCOL) where CHARCOL is a column of CHAR data type.

  Refer to the *DB2 Server for VSE & VM SQL Reference* manual for a complete list and description of column and scalar functions.

- Character or DBCS data was used in an arithmetic expression such -CHARCOL or CHARCOL+5 where CHARCOL is defined as in the first example. Numeric data (INTEGER, SMALLINT, DECIMAL, or FLOAT) may only be used in non-datetime arithmetic operations.
- The SQL statement contains an arithmetic expression with the special register USER. For example, USER + 100 is not permitted.

SQLSTATE 42819.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-404 **A character string specified in an INSERT or UPDATE statement is too large for the target column.**

Explanation: A character string in the data items-list of an INSERT statement or one of the character strings in the SET clause of an UPDATE statement is longer than the associated target column in the database.

SQLSTATE 22001.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-405 **The numeric value, *value*, is not within the range of the data type.**

Explanation: A numeric value specified in the statement exceeds the value allowed for the associated target column in the database. Following are the limitations:

- SMALLINT: -32768 to +32767
- INTEGER: -2147483648 to +2147483647
- FLOAT: -7.2E75 to -5.4E-79, 0, +5.4E-79 to +7.2E75
- DECIMAL: precision larger than that of the target column.

SQLSTATE 42820.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-407 **Either an UPDATE statement or an INSERT statement with a null value for a column defined as NOT NULL is not allowed, or a null host variable value is not allowed in a SELECT list.**

Explanation: An attempt was made to update a column with a null value and the column does not allow nulls. The null value may be from a constant value, host variable, an expression, or from a value returned on a subquery.

This SQLCODE will also be issued if you made an attempt to SELECT a host variable in a select-list, and the select-list had an indicator variable value of <0, indicating a NULL host variable value. A NULL host variable value is not allowed in a select-list. For more information about host indicator variables definition, refer to the *DB2 Server for VSE & VM Application Programming* manual.

SQLSTATE 23502.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-408 **An UPDATE or INSERT of a data value is incompatible with the data type of the associated target column.**

Explanation: A source value was incompatible with the associated target column. This can happen when the source value:

- Was derived from a subquery used in an UPDATE or INSERT statement.
- Was derived from a VALUES clause in an INSERT statement.

Typical cases are:

- The source value has a numeric data type, and the target column has a character, DBCS, or datetime data type.
- The source value has a character data type, and the target column has a numeric or DBCS data type.
- The source value has a DBCS data type, and the target column has a numeric, character, or datetime data type.
- The source value has a character data type that is too short for the target datetime column.
- The source value has a datetime data type, and the target column has a numeric, DBCS, or different datetime data type.

SQLSTATE 42821.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-409 **An incorrect specification was used with the column function COUNT. Only * or DISTINCT with a column name is allowed.**

Explanation: The column function COUNT may only be used with * or with the keyword DISTINCT and a column name. Examples of its use are: COUNT(*) or COUNT (DISTINCT PARTNO), where PARTNO is a column name.

Refer to the *DB2 Server for VSE & VM SQL Reference*

manual for a complete list and description of column and scalar functions.

SQLSTATE 42607.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-412 **You cannot specify more than one item in the SELECT clause of a subquery.**

Explanation: A subquery may only contain one item in its SELECT clause. For example, the following subquery is incorrect:

```
SELECT * FROM EMP_ACT
    WHERE EMPNO IN
      (SELECT EMPNO, JOB FROM EMPLOYEE
         WHERE WORKDEPT = 'E11')
```

SQLSTATE 42823.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-413 **An overflow occurred while the database manager was converting a data type internally.**

Explanation: A data value being converted by the database manager is too large for the data type to which it is being converted. For example, large floating point numbers will not convert to integers. This overflow is possible when using DECIMAL and INTEGER scalar functions.

SQLSTATE 22003.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-414 **The LIKE predicate was used for a numeric or date/time column type. LIKE must only be used with graphic or character columns.**

Explanation: The SQL statement uses LIKE against a numeric or datetime column. It can only be used against columns that have a data type that is compatible with the graphic data type or that have character data type.

SQLSTATE 42824.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-415 **The corresponding columns, *column-number*, of the operands of a UNION or a UNION ALL do not have comparable column descriptions.**

Explanation: The SQL statement contains SELECT clauses connected by a UNION or UNION ALL, and the corresponding columns in each SELECT clause cannot be compared. Specifically, columns of ordinality **column-number** of the operands in this UNION or UNION ALL cannot be compared. For columns to be comparable, corresponding columns must be numeric, character, graphic, date, time, or timestamp. Do not mix these data types. If corresponding columns have field procedures, both must have the same field procedure and the same column CCSID value.

Note the following default data types for literal constants in a SELECT list:

- Alphanumeric literal constant: default data type is varying character
- All numeric literal constant (no E or decimal point): default data type is integer.

SQLSTATE 42825.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-416 **You cannot specify a long string column in the SELECT clause of a UNION.**

Explanation: The SQL statement contained a SELECT clause item whose data type is a long string. This is not valid.

A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(*n*) where *n* is greater than 254 but less than or equal to 32767, or VARGRAPHIC(*n*) where *n* is greater than 127 but less than or equal to 16383.

SQLSTATE 42907.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-417 **Not all the operands of an arithmetic operator or predicate can be parameter markers.**

Explanation: The SQL PREPARE command contains an arithmetic operator or predicate with parameter markers used for all the operands. At least one of the operands must not be a parameter marker. Thus, the following are invalid:

```
? + ?
? < ?
? BETWEEN ? AND ?
? IN (?,?,?)
```

SQLSTATE 42609.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-418 **You cannot specify a parameter marker in the SELECT clause of a SELECT statement.**

Explanation: The SQL PREPARE statement contains a parameter in its SELECT clause.

SQLSTATE 42610.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-419 **The precision of the numerator and/or the scale of the denominator are too large for decimal division.**

Explanation: The internal formula used to calculate the resulting scale of decimal division is:

```
Scale of result = 31 - np + ns - ds

where  np = precision of numerator,
       ns = scale of numerator,
       ds = scale of denominator.
```

A negative resulting scale will cause incorrect results because of internal decimal constraints. This may also happen when using the AVG() function.

SQLSTATE 42911.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-422 **The operands of a UNION or UNION ALL do not have the same number of columns.**

Explanation: The operands of a UNION or UNION ALL must have the same number of columns.

SQLSTATE 42826.

System Action: Processing of the SQL statement ends.

User Response: Correct the SQL statement by putting the same number of columns in each operand.

-423 **Invalid result set locator value.**

Explanation: The value specified in a result set locator host variable does not identify a valid result set locator.

SQLSTATE: 0F001.

System Action: Processing of the SQL statement has ended.

User Response: There are two common causes for this error:

- The host variable used as a result set locator was never assigned a valid result set locator value. Result set locator values are returned by the DESCRIBE PROCEDURE and ALLOCATE LOCATOR statements. Make sure the value in your host variable is obtained from one of these statements.
- Result set locator values are only valid as long as the underlying SQL cursor is open. If a commit or rollback operation closes an SQL cursor, the result set locator associated with the cursor is no longer valid.

-424 **You specified an invalid sign digit in a decimal number.**

Explanation: Decimal numbers are stored with the sign indicator in the lower half byte of the final byte. For example, +300 is stored as 300C. 300 is the digit and the C is the sign digit. The sign digit can be from hexadecimal A to hexadecimal F. The digit must be in the range from hexadecimal 0 to hexadecimal 9. SQLCODE -424 is issued if the decimal data passed to the database manager has an invalid sign or digit.

For zoned decimal or numeric data, SQLCODE -424 indicates that at least one digit contains invalid zone bits or numeric bits.

SQLSTATE 22502.

System Action: Processing of the SQL statement ends.

User Response: Correct the data to have a valid sign and decimal digit. Try again.

-425 **The final result of a SUM is too large for the associated target column.**

Explanation: The database manager encountered a query similar to this:

```
SELECT SUM(column1) FROM table1
```

To process the sum, the database manager maintains an internal accumulator. This SQLCODE is issued when the value in the internal accumulator is greater than the final result value can be. For example, accumulating the sum of an integer column is done in a double word accumulator but the result can only be the length of a single word.

SQLSTATE 22003.

System Action: Processing of the SQL statement has ended.

User Response: If you need to work with the data values involved, break the calculation down into

smaller segments, so you do not exceed the capacity of the final result field.

-440 **FORMAT 1**
The number of parameters in the parameter list does not match the number of parameters expected for stored procedure *proc-name.*
n **parameters were expected.**
FORMAT 2
The number of parameters in the parameter list does not match the number of parameters expected for stored procedure *proc-name,*
AUTHID *authid. n* **parameters were expected.**

Explanation: The database manager received an SQL CALL statement for a stored procedure. The number of parameters supplied on the CALL statement does not match the number of parameters defined in SYSTEM.SYSPARMS.

proc-name
is the name of the stored procedure from the PROCEDURE field in SYSTEM.SYSROUTINES

authid is the authorization id from the AUTHID field in SYSTEM.SYSROUTINES

n is the expected number of parameters for the stored procedure with the NAME and AUTHID described above.

SQLSTATE: 42884

System Action: The statement cannot be executed.

User Response: If the SQL CALL statement is coded incorrectly, modify the SQL application to provide the correct number of parameters on the SQL CALL statement.

If SYSTEM.SYSPARMS contains incorrect information, you must redefine the stored procedure. You must do the following:
1. Use the STOP PROC operator command with the REJECT option to to disallow access to the procedure
2. Use the DROP PROCEDURE statement to remove the definition of the procedure
3. Use the CREATE PROCEDURE statement to redefine the procedure
4. Issue the START PROC command to allow access to the procedure

---

-444 **FORMAT 1**
User program *load-module-name* **could not be found for stored procedure** *proc-name.*
FORMAT 2
User program *load-module-name* **could not be found for stored procedure** *proc-name,*
AUTHID *authid.*

Explanation: DB2 received an SQL CALL statement for a stored procedure and found the row in the SYSTEM.SYSROUTINES catalog table associated with the requested procedure name. However, the load module or phase *load-module-name* identified in the LOADMOD column of the SYSTEM.SYSROUTINES row could not be found.

load-module-name
The name of the load module or phase that was not found.

SQLSTATE: 42724

System Action: Processing of the SQL statement has ended.

User Response: If the LOADMOD value in SYSTEM.SYSROUTINES is specified incorrectly, do the following:
1. Issue the STOP PROC operator command with the REJECT option to disallow access to the procedure
2. Use the ALTER PROCEDURE command to correct the value
3. Issue the START PROC operator command to allow access to the procedure

If the LOADMOD value is correct, ensure that the load module or phase exists and can be accessed by the stored procedure server.

-449 **CREATE PROCEDURE for** *name* **contains an invalid format of the external name clause or is missing the external name clause.**

Explanation: An error was found in the EXTERNAL NAME clause of the CREATE PROCEDURE statement for *name*, or the clause is needed but was not specified.

The external name must be a short identifier with letters or digits. A possible cause for this error is the inclusion of a blank in the name.

If the clause is omitted, the external name defaults to the name of the stored procedure. However, if the procedure name is longer than 8 characters then EXTERNAL NAME must be explicitly specified and a valid short identifier must be provided as the external name.

SQLSTATE: 42878

System Action: The statement cannot be executed.

User Response: Correct the syntax of the SQL statement. Refer to the *DB2 Server for VSE & VM SQL Reference* manual for information on the EXTERNAL NAME clause.

-450 **FORMAT 1**
Stored procedure *proc-name,*
parameter number *par-number,*
overlapped storage beyond its declared length.
FORMAT 2
Stored procedure *proc-name*
with AUTHID *authid,*
parameter number *par-number,*
overlapped storage beyond its declared length.

Explanation: While returning parameters from a stored procedure to an application, DB2 detected an overlay of one of the parameters. A stored procedure overwrote storage beyond a parameter's declared length.

proc-name
The name of the stored procedure from the PROCEDURE field in the SYSTEM.SYSROUTINES catalog table.

authid The authorization ID from the AUTHID column in SYSTEM.SYSROUTINES

par-number
The parameter position number. The parameter in error is the one for which the value of the column ORDINAL in SYSTEM.SYSPARMS matches the parameter position number provided in the error message.

SQLSTATE: 39501

System Action: The results of the stored procedure cannot be returned.

User Response: Examine the stored procedure to determine the cause of the overlay. Ensure that all parameters are declared correctly and that they match the corresponding definitions in SYSTEM.SYSPARMS.

-469 **SQL CALL statement must specify an output host variable for parameter** *par-number***.**

Explanation: DB2 received an SQL CALL statement for a stored procedure. DB2 found the row in the SYSTEM.SYSROUTINES catalog table with the requested procedure name. However, parameter *par-number* was identified in SYSTEM.SYSPARMS as an OUT or INOUT parameter. A host variable must be supplied on the SQL CALL statement for parameters defined as OUT or INOUT.

par-number
The parameter position number. The parameter in error is the one for which the value of the column ORDINAL in SYSTEM.SYSPARMS matches the parameter position number provided in the error message.

SQLSTATE: 42886

System Action: Processing of the SQL statement has ended.

User Response: If the SQL CALL statement is coded incorrectly, modify the SQL application to provide an output host variable on the SQL CALL statement.

If the parameter is defined incorrectly, you must change the procedure definition as follows:
1. Issue the STOP PROC operator command to disallow access to the stored procedure
2. Issue the DROP PROCEDURE statement to remove the procedure definition
3. Issue the CREATE PROCEDURE statement to redefine the procedure with the correct parameter definition
4. Issue the START PROC operator command to enable access to the stored procedure

---

-470 **SQL CALL statement specified a NULL value for input parameter** *par-number,* **but the stored procedure does not support NULL values.**

Explanation: DB2 received an SQL CALL statement for a stored procedure and found a null value in the incoming parameter list. The DB2 stored procedure was defined in the SYSTEM.SYSROUTINES catalog table with PARAMETERSTYLE = ' ', which specifies that the DB2 stored procedure does not accept null values.

par-number
The parameter position number. The parameter in error is the one for which the value of the column ORDINAL in SYSTEM.SYSPARMS matches the parameter position number provided in the error message.

SQLSTATE: 39002

System Action: Processing of the SQL statement has ended.

User Response: If the stored procedure should not accept null values, change the calling application to provide a nonnull value.

If the stored procedure should accept null values, then:
1. Issue the STOP PROC operator command with the REJECT option to disallow access to the stored procedure
2. Use the ALTER PROCEDURE statement to change the value of PARAMETERSTYLE to 'N'
3. Issue the START PROC operator command to reactivate the procedure

-471 **FORMAT 1**
SQL CALL for stored procedure *proc-name* **failed due to reason** *reason-code.*
FORMAT 2
SQL CALL for stored procedure *proc-name* **with AUTHID** *authid* **failed due to reason** *reason-code.*

Explanation: DB2 received an SQL CALL statement for a stored procedure. The CALL statement did not execute successfully.

Proc-name
The name of the stored procedure specified on the SQL CALL statement.

Authid The authorization ID associated with the stored procedure.

Reason code
The reason for the failure.

Possible reason codes are :

1 The load module or phase for the procedure was not found.

2 The procedure is STOPPED and calls to it are not being accepted.

3 No stored procedure server that can be used to run the stored procedure is available.

4 A Language Environment error occurred.

5 An SQL error occurred in the stored procedure.

6 The stored procedure server terminated abnormally.

7 The stored procedure handler was not able to load the load module or phase associated with the stored procedure.

8 An internal error has occurred.

SQLSTATE: 55023

System Action: Processing of the SQL statement has ended. A message describing the error may be displayed on the VM system console.

User Response: Correct the condition indicated by the reason code.

-478 **The stored procedure server** *pserver* **cannot be dropped because** *group-name*, **a new** *object-type*, **depends on it.**

Explanation: The stored procedure server cannot be dropped because it is the only stored procedure server in stored procedure group *group-name*, and stored procedures exist that must run in *group-name*.

SQLSTATE: 42893

System Action: Processing of the SQL statement ends.

User Response: Either define another stored procedure server in the stored procedure server group *group-name*, or modify the definition of any stored procedure that can run only in *group-name*. To modify the definition of a stored procedure, issue the ALTER PROCEDURE statement and specify the SERVER GROUP option, as follows:

1. If a default stored procedure server group exists and you want the stored procedure to run in it, do not specify a stored procedure server group name on the SERVER GROUP clause.
2. If you want the stored procedure to run in a different stored procedure server group, specify the name of that group on the SERVER GROUP clause.

-480 **The procedure** *procedure-name* **has not yet been called.**

Explanation: The procedure identified in a DESCRIBE PROCEDURE or an ASSOCIATE LOCATORS statement has not yet been called within the application process or the procedure has been called, but an explicit or implicit commit occurred before the statement.

SQLSTATE: 51030

System Action: Processing of the SQL statement has ended.

User Response: Correct the order of the statements and rerun the application.

-482 **The procedure** *procedure-name* **returned no locators.**

Explanation: The procedure identified in a ASSOCIATE LOCATORS statement returned no result set locators.

SQLSTATE: 51030

System Action: Processing of the SQL statement has ended.

User Response: Determine if result set locators are returned from the identified procedure by using the DESCRIBE PROCEDURE statement.

-499 **Cursor** *cursor-name* **has already been assigned to this or another result set from procedure** *procedure-name*.

Explanation: An attempt was made to assign a cursor to a result set using the SQL statement ALLOCATE CURSOR. One of the following applies:

- The result set locator variable specified in the ALLOCATE CURSOR statement has been previously assigned to cursor *cursor-name*.

- Cursor *cursor-name* specified in the ALLOCATE CURSOR statement has been previously assigned to a result set from stored procedure *procedure-name*.

SQLSTATE: 24516

System Action: Processing of the SQL statement has ended.

User Response: Determine if the target result set named in the ALLOCATE CURSOR statement has been previously assigned to a cursor.

If the result set has been previously assigned to cursor *cursor-name*, then either choose another target result set or call stored procedure *procedure-name* again and reissue the ASSOCIATE LOCATOR and ALLOCATE CURSOR statements.

If the result set has not been previously assigned to a cursor, the cursor *cursor-name* specified in the ALLOCATE CURSOR statement has been previously assigned to some result set from stored procedure *procedure-name*. You cannot assign cursor *cursor-name* to another result set, so you must specify a different cursor name in the ALLOCATE CURSOR statement.

-501 The cursor referenced in a FETCH, PUT, or CLOSE statement is not open.

Explanation: The cursor referenced in a CLOSE, PUT, or FETCH is not open now.

SQLSTATE 24501.

System Action: Processing has ended.

User Response: Correct the cause of the problem. Try again.

-502 The cursor referenced in an OPEN or ALLOCATE CURSOR statement is already open.

Explanation: You cannot open or allocate a cursor that is already in an open state.

SQLSTATE 24502.

System Action: Processing of the SQL statement has ended.

User Response: Correct the cause of the problem. Try again.

-503 A column used in the SET clause of an UPDATE statement containing a WHERE CURRENT OF clause was not identified in the FOR UPDATE OF clause of the DECLARE CURSOR statement.

Explanation: One or more columns named in the SET clause were not declared in the FOR UPDATE OF clause of the query which defined the cursor.

SQLSTATE 42912.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-504 The cursor name or statement name, *name*, has not been properly declared or prepared.

Explanation: One of the following has occurred:

- The cursor name has not been defined with a DECLARE CURSOR statement or allocated with an ALLOCATE CURSOR statement.
- An attempt was made to define the cursor name with a DECLARE CURSOR statement or an ALLOCATE CURSOR statement, but an error was encountered on the DECLARE CURSOR or ALLOCATE CURSOR statement.
- Cursor *name* was referenced in a positioned UPDATE or DELETE statement which is not a supported operation for an allocated cursor.
- Cursor *name* was allocated, but a CLOSE cursor statement naming cursor *name* was issued and deallocated the cursory before this cursor reference.
- Cursor *name* was allocated, but a ROLLBACK operator occurred and deallocated the cursor before this cursor reference.
- Cursor *name* was allocated, but its associated stored procedure was called again since the cursor was allocated, new result sets were returned, and cursor *name* was deallocated.
- The statement name has not been defined with a PREPARE statement.
- An attempt was made to define the statement name with a PREPARE statement, but an error was encountered on the PREPARE SQL statement.

SQLSTATEs:

34000 Cursor reference

07003 Statement reference

26505 An Extended EXECUTE, Extended DECLARE CURSOR, or Extended PREPARE against a NULL section.

System Action: Processing of the SQL statement has ended.

User Response: Define the name properly. Try again.

For an allocated cursor, if an implicit or explicit COMMIT, ROLLBACK, or CLOSE occurred after the cursor was successfully allocated, modify the application program logic so that after the COMMIT, ROLLBACK, or CLOSE operation, it calls the associated stored procedure again, and reissue the ASSOCIATE LOCATORS and ALLOCATE CURSOR statements.

For an allocated cursor, if the associated stored

procedure was called again and new result sets were returned after the cursor was allocated, reissue the ASSOCIATE LOCATORS and ALLOCATE CURSOR statements.

If this error was encountered while preprocessing of a FETCH statement that references a cursor allocated by a dynamic ALLOCATE CURSOR statement, the DYNALC preprocessor option must be used to allow the FETCH to be processed. For more information, refer to the *DB2 Server for VSE & VM Application Programming* manual.

-505 **A cursor has been declared or a statement has been prepared more than once.**

Explanation: The name used in a DECLARE CURSOR or PREPARE statement was incorrect. A cursor has been declared more than once or a statement name has been used in more than one PREPARE statement.

SQLSTATE Spaces.

System Action: Processing of the SQL statement has ended.

User Response: Correct the SQL statement using a different cursor or statement name. Try again.

-506 **More than one cursor is declared for statement** *statement***.**

Explanation: You can only declare one cursor for **statement** shown in this message.

SQLSTATE Spaces.

System Action: Preprocessing of the statement ends.

User Response: Ensure that you have defined only one cursor per statement in your application program. Try again.

-507 **An UPDATE or DELETE statement uses a WHERE CURRENT OF clause, but no cursor is open.**

Explanation: A reference to CURRENT OF CURSOR was made but cursor was not opened.

Note: If blocking is used on the cursor, you may get a -507 even though you may not have explicitly closed the cursor in DRDA protocol. This is because the application server has implicitly closed the cursor.

SQLSTATE 24501.

System Action: Processing of the SQL statement has ended.

User Response: Ensure the cursor defined on the SELECT statement is OPENED.

-508 **An UPDATE or DELETE statement uses a WHERE CURRENT OF clause, but the cursor is not properly positioned on a row.**

Explanation: A reference to a CURRENT OF CURSOR was made when the cursor was not positioned on a row or no row has been fetched.

SQLSTATE 24504.

System Action: Processing of the SQL statement has ended.

User Response: Verify that the value in the SQL statement matches the row in the table. Try again.

-509 **An UPDATE or DELETE statement with a WHERE CURRENT OF clause specifies a table for which no cursor exists.**

Explanation: A reference was made to CURRENT OF CURSOR when a cursor is not defined on the current table.

SQLSTATE 42827.

System Action: Processing of the SQL statement has ended.

User Response: Either correct the UPDATE or DELETE statement by removing the WHERE CURRENT OF clause or declare the necessary cursor. Try again.

-510 **You cannot use a WHERE CURRENT OF clause in this UPDATE or DELETE statement.**

Explanation: You can not use a DELETE WHERE CURRENT OF clause if the query is not deleteable. You can not use an UPDATE WHERE CURRENT OF clause if the query is not updateable.

1. The following are cases of nondeleteable queries:
   - Multiple tables are involved.
   - You can not use DISTINCT, GROUP BY, ORDER BY, UNION, or a column function.
   - Subqueries cannot reference the same table.
   - Isolation level UR is used.
2. The following are cases of nonupdateable queries:
   - Multiple tables are involved.
   - You can not use DISTINCT, GROUP BY, ORDER BY, UNION, or a column function.
   - Subqueries cannot reference the same table.
   - The FOR UPDATE OF clause must specify the column to be updated in the UPDATE statement.

SQLSTATE 42828.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-511 **You cannot declare a cursor with a FOR UPDATE OF clause that involves a query that is not updateable.**

Explanation: Following are some examples of queries that are not updateable:

- Multiple tables are being used.
- Subqueries cannot reference the same table.
- May not use DISTINCT, GROUP BY, ORDER BY, UNION, or a column. function
- The NOFOR option was specified during preprocessing and the database manager appended the FOR UPDATE OF clause to the statement.

SQLSTATE 42829.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-512 **You cannot open or close a cursor that has been prepared with an SQL statement other than SELECT or INSERT.**

Explanation: An OPEN or CLOSE has been attempted on a cursor that is not associated with a SELECT or INSERT statement.

SQLSTATE 07005.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-513 **More than 512 disjoint DECLARE CURSOR and PREPARE statements have been used in the program.**

Explanation: If a PREPARE statement uses a cursor, the preprocessor matches it to the appropriate DECLARE CURSOR statement. The total number of such pairs plus any unmatched DECLARE CURSOR and PREPARE statements cannot exceed 512.

SQLSTATE Spaces.

System Action: Processing of this SQL statement has ended.

User Response: Rewrite the program to contain fewer disjoint DECLARE CURSOR and PREPARE statements. Try again.

-515 **You cannot use the SQL EXECUTE statement to perform a COMMIT or ROLLBACK.**

Explanation: The statements that control logical units of work cannot be the object of dynamic SQL statements; PREPARE, EXECUTE, or EXECUTE IMMEDIATE.

SQLSTATEs:

2D528 Dynamic COMMIT is invalid.

2D529 Dynamic ROLLBACK is invalid.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-516 **The current statement references a statement-name which has not been properly prepared.**

Explanation: The current statement refers to a statement name that has not been properly prepared. The current statement is an EXECUTE, a DESCRIBE, or a dynamic cursor operation.

SQLSTATEs:

07003 EXECUTE

26501 DESCRIBE or Dynamic cursor operation.

System Action: Processing of the statement has ended.

User Response: Correct the logic of the application program to ensure that the referenced statement is properly prepared prior to the current statement, and in the same logical unit of work. Try again.

-517 **The cursor name** *cursor-name* **has already been declared for package** *package-name***.**

Explanation: The cursor name specified in a DECLARE CURSOR statement has already been declared for the named package in the current logical unit of work.

SQLSTATE 3C000.

System Action: Processing of the SQL statement ends.

User Response: Two cursors of identical names cannot be declared in the same logical unit of work. Move this statement outside the current logical unit of work, or end the current logical unit of work with a COMMIT or ROLLBACK command before executing this statement. Correct the error and rerun the program.

-518 **The application program references the cursor** *cursor-name***, but the cursor is not declared in the current logical unit of work.**

Explanation: The cursor name used in an Extended OPEN, FETCH, PUT, or CLOSE was not declared (using an Extended DECLARE cursor) in the current logical unit of work.

SQLSTATE 42719.

System Action: Processing of the SQL statement ends.

User Response: Roll back the logical unit of work, then correct the error. Try again.

-519 **The database manager is blocking INSERTs. The application has issued a COMMIT, and there is at least one INSERT cursor which the program has not yet closed.**

Explanation: The application, in which the database manager is blocking INSERTs, has executed a COMMIT statement, but there is still at least one opened INSERT cursor (which has been blocked) in the current logical unit of work that has not been closed. Note that the application may have run successfully on previous occasions because the database manager chose not to block because of storage limitations or was executing the program in single user mode.

SQLSTATE 24505.

System Action: The COMMIT has not been executed and the current logical unit of work has been rolled back.

User Response: Modify the program to ensure that all OPEN cursors have a corresponding CLOSE for the same cursor. Also ensure that all opened cursors will be closed prior to executing a COMMIT statement. After these two conditions are met, preprocess, compile, and execute the program again.

-520 **The database manager is blocking FETCHs for cursor** *cursor-name***. Therefore, a DELETE... WHERE CURRENT OF CURSOR statement cannot be executed against that cursor.**

Explanation: Because the database manager is blocking FETCHs for the mentioned cursor, any dynamic or extended dynamic PREPAREs, or EXECUTE IMMEDIATEs of DELETE... WHERE CURRENT OF CURSOR statements against that cursor are considered errors and will not be executed.

SQLSTATE 42828.

System Action: Processing of the PREPARE or EXECUTE IMMEDIATE statement is terminated.

User Response: Blocking applications cannot use DELETE... WHERE CURRENT OF CURSOR statements in conjunction with SELECT cursors that are qualified for blocking. Such DELETE statements should be removed from blocking applications. If the application continues execution after receiving this SQLCODE, any subsequent EXECUTEs, which are dependent upon the PREPARE of the DELETE, also will not be executed successfully by the database manager.

-521 **The database manager is blocking FETCHs or PUTs. The data type or length has changed for a host variable whose ordinal position is** *position-number* **in the host variable list or descriptor SQLDA.**

Explanation: The database manager is blocking FETCHs or PUTs. A discrepancy has been found concerning the use of host variables in the FETCH or PUT. The reason code is in SQLERRD1. The ordinal position of the host variable is in SQLERRD2. The meaning of the SQLERRD1 codes are as follows:

152, 156
The number of elements in the SQLDA or in the host variable list of the current statement does not match that of the original PUT or FETCH.

160 The nulls allowed setting of a host variable in the SQLDA or host variable list for the current statement does not match that of its counterpart in the original FETCH or PUT for the cursor. SQLERRD2 contains the ordinal position of the incorrect host variable.

164 The length of a host variable in the SQLDA or host variable list does not match the length of its counterpart in the first FETCH or PUT of this cursor. SQLERRD2 is the ordinal number of the incorrect host variable.

168 The data type of a host variable in the SQLDA or host variable list implies that nulls are allowed but no indicator variable is supplied. SQLERRD2 is the ordinal position of the incorrect host variable.

172 The length associated with a host variable is negative or zero. SQLERRD2 is the ordinal position of the incorrect host variable.

This reason code can also be generated by a NUMERIC or DECIMAL type with a length greater than 31.

176 The data type of a host variable in the SQLDA or host variable list does not match its counterpart in the original PUT or FETCH statement for this cursor. SQLERRD2 is the ordinal position of the incorrect host variable.

180 The SQLDATA or SQLIND pointer to a host variable is zero. SQLERRD2 is the ordinal position of the incorrect host variable.

Note: This error can also occur when the database manager is not blocking FETCHs or PUTs.

SQLSTATE 55021.

System Action: Processing of the PUT or FETCH statement is terminated.

For PUTs, the inserted data is not included in the block buffer.

For FETCHs, the row is not returned to the application and is skipped as part of the output answer set. The next FETCH of the cursor would return the next row, if there is one, of the answer set.

User Response: Check the application. If the host variable associated with an ordinal position in the host variable list or DESCRIPTOR SQLDA changes, make sure its data type and length are identical to the host variable in that position for the previous FETCH or PUT. Also check that the code does not alter the data type or length of elements in the DESCRIPTOR SQLDA from on FETCH or PUT to the next. The data types and lengths may be altered after the subject cursor is closed, but once the cursor is opened, each ordinal position is bound to the data types and lengths specified on the first FETCH or PUT after the OPEN.

You may wish to continue processing even though the row of information was not included in the input or output set. Or you may issue a ROLLBACK and terminate the program, make the correction, and then preprocess, compile, link edit, and run the application again.

-522 **NCSCANS, the initialization parameter that defines the number of cursors which you may open concurrently in a logical unit of work, has been exceeded.**

Explanation: The current logical unit of work in the user application has too many cursors opened concurrently. Each opened cursor usually corresponds to one internal scan, but when joins and subqueries are used, the database manager implicitly opens its own cursors. As a guideline, the database manager will open as many cursors as there are subqueries or tables joined in an SQL SELECT statement (or DECLARE CURSOR SELECT statement). Additional cursors may be opened, however, if the access path chosen requires sorts for ordering. The database manager also implicitly uses internal cursors to execute Searched DELETE, Searched INSERT via SELECT statement, Searched UPDATE, and CREATE INDEX statements and to access its catalog and packages.

One of the following limitations occurred for the current logical unit of work:

- The storage reserved for scans made by the NCSCANS DB2 Server for VM initialization parameter is exceeded, or
- The number of concurrently opened scans exceeded 255. 255 is the maximum allowed for an DB2 Server for VM user.

SQLSTATE 54014.

System Action: Processing of the SQL statement has ended.

User Response: Acquire information about the number of open cursors in the current logical unit of work and the nature of the SQL SELECT statements associated with those cursors and contact your database administrator. The database administrator may have to change the DB2 Server for VM initialization parameter NCSCANS, or the application may need more SQL COMMIT statements to reduce the number of opened scans in a logical unit of work.

-523 **Section** *section-number* **of package** *package-name* **was not created as the result of executing the NULL form of an extended dynamic PREPARE or of preprocessing a dynamic PREPARE statement.**

Explanation: The user's application is executing a dynamic PREPARE or the FOR form of an Extended Dynamic PREPARE for the package named **package-name**. The database manager does not allow such execution because the section number **section-number** was not created by the execution of either the NULL form of an Extended Dynamic PREPARE, or the previous preprocessing of a dynamic PREPARE statement.

SQLSTATE 54015.

System Action: Processing of the SQL statement has ended.

User Response: The program and the package are probably out of synchronization. Preprocess the program again; then compile and link-edit the program to ensure that the package matches the application. If you are executing the FOR form of the Extended Dynamic PREPARE, make sure that the statement-id specified was generated using the NULL form of the Extended Dynamic PREPARE.

-524 **You used an ORDER BY clause in a query with no cursor.**

Explanation: The system assumes that a statement involving an ORDER BY clause will refer to more than one row. Therefore, queries using an ORDER BY clause should use a cursor.

SQLSTATE 42601.

System Action: Processing of the SQL statement has ended.

User Response: Remove the ORDER BY clause, or define a cursor. Try again.

-525 **Section** *section-number* **of package** *package-name* **cannot be executed because of an error that was detected at preprocessing time.**

Explanation: At preprocessing time, the ERROR parameter was specified, and the package was stored in the database with incorrect sections. At execution time,

an attempt was made to execute the section that contained the error.

SQLSTATE 51015.

System Action: Execution of the statement has ended.

User Response: Correct the statement. Preprocess, compile, link edit, and run the program again.

-530 **The INSERT or UPDATE value of a foreign key** *constraint-name* **owned by** *owner* **is invalid.**

Explanation: There is an attempted violation of the referential constraint **owner.foreign-key** during the execution of an insert or update operation on a dependent table. The value is invalid because it does not also appear as a value of the PRIMARY KEY of the parent table of the relationship.

SQLSTATE 23503.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-532 **The DELETE or UPDATE operation is prevented by a rule of RESTRICT associated with the foreign key** *constraint-name* **owned by** *owner***.**

Explanation: There is an attempted violation of the referential constraint *owner.constraint-name* during the execution of a DELETE or UPDATE operation on a parent table. A row in the table cannot be deleted or updated because of either of the following:

- It has a dependent relationship with a rule of RESTRICT, or
- Its deletion deletes a row of another table that has a dependent in a relationship with a delete rule of RESTRICT.

SQLSTATE 23504.

System Action: Processing of the SQL statement has ended.

User Response: Examine the constraint rule for all descendent tables to determine the cause of the problem. The specific tables involved can be determined from the relationship *constraint-name*.

-534 **Invalid multiple row update.**

Explanation: An UPDATE statement cannot be processed because the table is in a nonrecoverable storage pool and a PRIMARY KEY column is an object of the SET clause, and more than one row is selected for updating.

SQLSTATE 21502.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-535 **Invalid UPDATE WHERE CURRENT OF clause.**

Explanation: The UPDATE WHERE CURRENT OF clause cannot be used when the column to be updated is part of a PRIMARY KEY column.

SQLSTATE 42913.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-536 **The DELETE or UPDATE statement is invalid because table** *owner.table-name* **can be affected by the operation.**

Explanation: The DELETE or UPDATE statement is invalid because the indicated table is referenced in a subquery of the statement. The indicated table is either:

- A dependent of the object table in a relationship with a delete rule of SET NULL or CASCADE, or
- A dependent of another table, call it T, in a relationship with a delete rule of SET NULL or CASCADE, and table T is delete-connected to the object table.

SQLSTATE 42914.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-537 **The** *key-type* **identifies column** *column-name* **more than once.**

Explanation: The indicated column-name appears more than once in a key. **Key-type** could be PRIMARY KEY, FOREIGN KEY, or UNIQUE CONSTRAINT.

SQLSTATE 42709.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-538 **Foreign key** *constraint-name* **does not conform to the description of the primary key of table** *owner.table-name***.**

Explanation: The definition of the indicated FOREIGN KEY does not conform to the description of the PRIMARY KEY of the indicated table. Either the keys do not have the same number of columns, or the description of the corresponding columns of the FOREIGN KEY is not identical to the PRIMARY KEY. The requirement for identical descriptions includes data

type, field procedure, and length, but not the null attributes. If a field procedure is defined for the corresponding columns, their column CCSID values must be identical. The name is the FOREIGN KEY name specified in the FOREIGN KEY clause or, if a constraint-name was not specified, the generated name.

SQLSTATE 42830.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-539 **Table** *owner.table-name* **does not have a primary key.**

Explanation: The indicated table, which is specified as a parent in a FOREIGN KEY clause, cannot be defined as a parent because it does not have a PRIMARY KEY.

SQLSTATE 42888.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-542 *column-name* **cannot be a column of a** *key-type* **because it can contain NULL values.**

Explanation: A column identified in a key is defined to allow null values. **Key-type** can be a PRIMARY KEY or UNIQUE CONSTRAINT. You cannot allow null values in a PRIMARY KEY or UNIQUE CONSTRAINT column.

SQLSTATE 42831.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-550 **You cannot create or acquire the** *object-type***. The owner SYSTEM is reserved for internal use.**

Explanation: The owner SYSTEM is reserved for internal use and therefore cannot be used as specified.

SQLSTATE 42939.

System Action: Processing of the SQL statement has ended.

User Response: Change owner to a valid name. Try again.

-551 **User** *user-id* **does not have the privilege to perform** *action* **on** *object-name***.**

Explanation: An attempt was made to reference an object that the user is not authorized to use, or the SQL statement needs specific authority that the user does not have (for example, RUN authority). This can occur if a user with DBA authority preprocesses a program but because of the nature of the SQL statements in the program, the DBA does not have the authority to grant RUN authority to another user. Refer to the *DB2 Server for VSE & VM Application Programming* manual for information on defining privileges on packages.

This error might also occur if the object is a read-only view (for GRANT DELETE/INSERT/UPDATE).

When this SQLCODE results from GRANT ALL, the grantor has no privilege on the object table/view upon which he/she can grant privileges.

No error message is issued for GRANT ALL on a view if the grantor has at least one grant privilege on the view.

No error message are issued for GRANT ALL on a table if the grantor has less than all possible privileges on the table as long as he/she has at least one privilege.

This code is returned either when, a statement is being executed, or a program is being preprocessed with the EXIST option.

Formats of this code include:

Format 1:

- User &1. does not have the privilege to perform &2. on &3..

Format 2

- User &1. does not have the &2. privilege to affect keys on &3..&4..

Format 3

- User &1. does not have the privilege to perform &2..

SQLSTATE 42501.

System Action: Processing of the SQL statement has ended.

User Response: Check that the user ID, owner, and object name are correct and acquire the required authorization for the object.

-552 *userid* **is not authorized to** *action***.**

Explanation:

Format 2:

- User ID is not authorized to CREATE a package for another user.

Format 3:

- User ID is not authorized to unload a package.

Format 4:

- User ID is not authorized to reload a package.

Format 5:

- User ID is not authorized to revoke special privileges.

Format 6

- User ID is not authorized to insert or delete rows from the system catalog.

Format 7

- User ID is not authorized to drop the catalog table SYSDBSPACES.

Format 8

- User ID is not authorized to alter the catalog table **action**.

Format 9

- User ID is not authorized to grant an authority.

Format 10

- User ID is not authorized to revoke an authority.

Format 11

- User ID is not authorized to REBIND a package for another user.

This code is returned when either a statement is being executed, or a program is being preprocessed with the EXIST option.

SQLSTATEs:

Spaces **action** is to unload or reload a package

42502 **action** is anything else

System Action: Processing of the SQL statement has ended.

User Response: Create the required object or provide the proper name of the object. Try again.

-554 **You cannot grant a privilege to yourself.**

Explanation: The database manager does not allow you to grant your own privileges or set your password even though you have been granted CONNECT authority implicitly through the command "GRANT CONNECT TO ALLUSERS". You can only change your password after the DBA has granted CONNECT authority to you explicitly and set your initial password.

SQLSTATE 42502.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-555 **You cannot revoke an authority or a privilege from yourself.**

Explanation: The authorization subsystem does not allow you to revoke your own authority or privilege.

SQLSTATE 42502.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-556 **An attempt to revoke a privilege from** *revokee* **was denied. Either** *revokee* **does not have this privilege, or** *revoker* **does not have the authority to revoke this privilege.**

Explanation: The authorization subsystem neither allows you to revoke a privilege which has not been granted nor to exceed your authority.

SQLSTATEs:

42502 Revoker does not have authority.

42504 Revokee does not have privilege.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-557 **You cannot grant or revoke this combination of privileges.**

Explanation: User granted a privilege which conflicted with the granting of another privilege in the same statement.

SQLSTATE 42852.

System Action: Processing of the SQL statement has ended.

User Response: Refer to the "Statements" section in the *DB2 Server for VSE & VM SQL Reference* manual. Correct the SQL statement and resubmit it.

-558 **You cannot revoke an authority from** *user-id* **because** *user-id* **has DBA authority.**

Explanation: If user has DBA authority, only this authority can be revoked by another user with DBA authority. All other authority (RESOURCE, CONNECT, SCHEDULE) cannot be revoked without revoking DBA authority first.

SQLSTATE 42504.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-560 **A CONNECT statement contains an incorrect password for** *user-id***.**

Explanation: The password you submitted to access the database manager is incorrect for your user ID.

SQLSTATE 42505.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-561 **User *user-id* does not have CONNECT authority.**

Explanation: The specified user is not authorized to access the database manager because CONNECT authority has not been granted.

SQLSTATE 42505.

System Action: Processing of the SQL statement has ended. If you are attempting to switch databases, SQLWARN0 and SQLWARN6 are set to 'S' to signal that a severe error has occurred.

User Response: See your database administrator to obtain the authorization you require. If a severe error has occurred, only a CONNECT statement will be accepted.

-562 **ALLUSERS is a reserved user ID. ALLUSERS can only be used in a GRANT CONNECT statement without a password.**

Explanation: You cannot grant DBA, SCHEDULE, or RESOURCE authority to the reserved user ID ALLUSERS. You can only grant ALLUSERS CONNECT authority without a password.

SQLSTATE 56034.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-564 **Either the statement contains an incorrect authorization name or password, or the password is missing.**

Explanation: Authorization name or password must begin with an alphabetic character and contain up to eight alphabetic or numeric characters. It cannot contain embedded blanks or special characters. A user ID of SYSTEM, PUBLIC, ALLUSERS, or $TOTAL% is not valid. This error also occurs if a password is not supplied.

If SECTYPE=ESM, SQLERRD1 provides extra information about error:

SQLERRD1	Meaning
09	External Security Manager - retryable error
10	External Security Manager - non-tryable error
14	Password Expired
15	Password Invalid
16	Password Missing
18	Userid Missing
19	Userid Invalid
20	Userid Revoked

SQLSTATE 28000.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-565 **Package *package-name1* was specified but another package, *package-name2*, is the subject of the current logical unit of work. You must end the logical unit of work before changing packages.**

Explanation: You established a package as the target of all activity in the current logical unit of work. The package in the SQL statement now being processed is not the same as the target for the logical unit of work.

SQLSTATE 56042.

System Action: Processing of the SQL statement has ended.

User Response: Roll back the logical unit of work and correct the error that is giving the incorrect package. Try again.

-566 **User ID *user-id* does not have authorization to modify package *package-name*.**

Explanation: Only the owner or a user who has DBA authority can modify a package.

SQLSTATE 42501.

System Action: Processing of the SQL statement has ended.

User Response: Rollback the logical unit of work, and correct the error. Try again.

-568 **INFILE contents invalid.**

Explanation: An error has been detected in the RELOAD PACKAGE input data file. The format or content of the input file is incorrect.

SQLSTATE Spaces.

System Action: The command is not processed. See "Common Error Action" on page 223 for more information.

Programmer Response: Contact the system programmer. The system programmer should perform problem determination. If a DB2 Server for VM system error occurred, make a record of what went wrong and contact the designated support group for your installation.

-601 **The name used in the CREATE statement or the ACQUIRE DBSPACE statement already exists.**

Explanation: An attempt was made to create a table, create an index, create a view, create a synonym, or acquire a dbspace using a name that already exists. SQLERRD1, module names, and descriptions are shown below:

SQLERRD1	Module Name and Description
-110	ARIXI06 Index already exists.
-110	ARIXI08 Name already exists as a table.
-120	ARIXI07 Name already exists for this user as a table.
-130	ARIXI09 View name already exists for this user.
-130	ARIXI08 Name already exists for this user as a synonym.
-140	ARIXI01 DBSPACE already exists.
-140	ARIXI07 Name already exists for this user as a synonym.
-150	ARIXI09 Name already exists for this user as a synonym.

SQLSTATE 42710.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-602 **You can neither create nor alter a table or a view to establish more than 255 columns.**

Explanation: The maximum number of columns for a table or view is 255. The statement contains more than 255 columns for the table or view.

SQLSTATE 54011.

System Action: Processing of the SQL statement has ended.

User Response: Correct the SQL statement so no more than 255 columns will be established for table or view.

-603 **A *key-type* cannot be created because the table contains rows that are duplicates with respect to the values of the identified columns.**

Explanation: The following are invalid:

- Creating a unique index, primary key, or unique constraint on columns that contain equal values.
- Reorganizing an index that contains equal values.
- Activating a primary key or unique constraint with a unique index that contains equal values.

The token **key-type** can be UNIQUE INDEX, PRIMARY KEY, or UNIQUE CONSTRAINT. (If a column has a type of VARCHAR or VARGRAPHIC, two columns that differ only in the number of trailing blanks might cause duplication.)

SQLSTATE 23515.

System Action: Processing of the SQL statement ends.

User Response: If you are creating a unique index, primary key, or unique constraint, correct the statement. Try again.

If you are reorganizing a unique index or activating a primary key or unique constraint, then the unique index or supporting unique index is marked invalid with `Duplicates Exist in UNIQUE INDEX`. You can use the SHOW INVALID operator command to display invalid indexes.

Either drop the index, remove the duplicates, and recreate the index, or deactivate the primary key or unique constraint, remove the duplicates, and activate the primary key or unique constraint again. For more information about invalid indexes, see the SHOW INVALID operator command in the *DB2 Server for VSE & VM Operation* manual.

-604 **An incorrect length, precision, or scale was specified for a column.**

Explanation: You cannot create or expand the table to establish the precision, scale (decimal type), or field length specified because it is beyond the design limits of the column type.

- Decimal precision must be greater than or equal to `0` and less than or equal to 31. Scale must be less than or equal to precision.
- CHAR type must have a length less than or equal to 254.
- FLOAT must have a length greater than `0` and less than or equal to 53.
- GRAPHIC type must have a length less than or equal to 127.
- VARCHAR type must have a length less than or equal to 32767.
- VARGRAPHIC type must have a length less than or equal to 16383.

SQLSTATE 42611.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-605 **The creation of an index failed because you exceeded the limit for indexes on a table.**

Explanation: The maximum number of indexes that can be created for a table is 255. The target table for an index already has 255 indexes. Adding an index to a table can occur from a CREATE INDEX statement or by adding a PRIMARY KEY or UNIQUE CONSTRAINT to the table.

SQLSTATE 54020.

System Action: Processing of the SQL statement ends.

User Response: Drop an index that you do not need from the table. Try again.

-606 **The COMMENT ON or LABEL ON statement failed because the specified table or column is not owned by** *owner***.**

Explanation: An attempt was made to comment or label on a table or column which does not exist or is not owned by the owner specified in the message text.

SQLSTATE 42502.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-607 **The CREATE TABLE statement failed because adding a table would cause more than 255 tables in dbspace** *owner.dbspace-name***.**

Explanation: The dbspace specified in the CREATE TABLE statement exists, but it has 255 tables, which is the maximum number of tables allowed in a dbspace.

The number 255 includes both tables in the dbspace and any internal tables created because of long fields defined in the tables. Each table with one or more long fields requires an internal table to be created in the dbspace.

If the dbspace specified is PUBLIC.SYSnnnn, where nnnn >= 0002 and nnnn <= 9999, the dbspace has 255 tables created for packages. Information about these tables is in catalog table SYSACCESS.

SQLSTATE 54016.

System Action: Processing of the SQL statement has ended.

User Response: There are the following options:

1. Create the table in a different dbspace.
2. Drop a table in the current dbspace to free up an entry.
3. Acquire a new dbspace for the table to be created. This may require assistance from your database administrator.

Database Administrator Response: If appropriate, acquire a private dbspace for the user.

-608 **The CREATE TABLE statement failed because the dbspace** *dbspace-name* **was not found for the table** *owner.table-name***.**

Explanation: The dbspace specified in the CREATE TABLE statement was not found. The search order is:

1. The database manager first searches for a dbspace with a qualifier of dbspaceowner.dbspace-name.
2. If the dbspace owner is not specified in the CREATE TABLE statement, then the database manager looks for a private dbspace qualified as owner.dbspace-name.
3. Finally, the database manager searches for a public dbspace qualified as public.dbspace-name.

SQLSTATE 42704.

System Action: Processing of the SQL statement has ended.

User Response: These are the options:

1. Create the table in a different dbspace.
2. Acquire a new dbspace for the table to be created. This may require assistance from your database administrator.

Then rerun the CREATE TABLE statement.

Database Administrator Response: If appropriate, acquire a dbspace for the user.

-609 **The ALTER statement cannot be applied to a system DBSPACE** *dbspace-name***.**

Explanation: An attempt was made to change a public system dbspace.

- For SYS001, you tried to change the level of the lockmode to DBSPACE or PAGE.
- For other than SYS001, you tried to change the lockmode or freepct of the DBSPACE.

SQLSTATE 42832.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-610 **The statement failed because a user without DBA authority attempted to create a table in a dbspace owned by another user or by the system.**

Explanation: Only users with DBA authority can create a table in another user's dbspace or a system dbspace.

SQLSTATE 42502.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-611 **The ALTER statement failed because the NOT NULL option was specified for the new column.**

Explanation: The new column specified in an ALTER TABLE statement must allow null values. The NOT NULL option cannot be used.

SQLSTATE 42601.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-612 **The statement failed because the column** *column-name* **was specified more than once in the CREATE TABLE or CREATE VIEW or CREATE INDEX, or the column** *column-name* **specified in the ALTER statement already exists in the table.**

Explanation: All columns within a specific table, view or index must be unique. The user tried to create a column with name as specified in the message, but it already exists in the table.

SQLSTATE 42711.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-614 **You cannot create a(n)** *key-type1* **on a long string column or on a table where the length of the encoded key derived from the** *key-type2* **column(s) is greater than 255 bytes.**

Explanation: Either:

- Creating an index, primary key or unique constraint on a column defined as a long string is not permitted. A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(*n*) where *n* is greater than 254 but less than or equal to 32767, or VARGRAPHIC(*n*) where *n* is greater than 127 but less than or equal to 16383.
- Creating an index, primary key or unique constraint on a table, where the encoded key created for the indexed columns is greater than 255 bytes, exceeds an internal limitation of the database manager. The maximum length for a non-unique index is 251 bytes.

SQLSTATE 54008.

System Action: Processing of the SQL statement has ended.

User Response: For a multiple column key, use fewer columns. If only one column is involved and its data type is VARCHAR or VARGRAPHIC, the length of the encoded value used as the key must not exceed 255. To prevent the encoded key from exceeding the limit, the length of a VARCHAR column must be less than or equal to 204; the length of a VARGRAPHIC column must be less than or equal to 102.

-615 **Package name** *owner.package-name* **not found in application server** *server-name***.**

Explanation: The database manager did not find the identified package in the SYSTEM.SYSACCESS catalog table for this application server.

SQLSTATE Spaces.

System Action: The command is not processed. See "Common Error Action" on page 223 for more information.

Programmer Response: Ensure the package-name specified in the command is correct. If it is incorrect, correct the command and reprocess it. (See "Rerunning DBS Utility" on page 223.) If the package-name is correct, create the package before processing it with the UNLOAD PACKAGE command, or specify the NEW parameter in the RELOAD PACKAGE command.

-616 **The RELOAD PACKAGE command with the NEW option cannot be used because the package already exists in server** *server-name***.**

Explanation: The Database Services Utility cannot create a package because one with the same package-name already exists in this application server.

SQLSTATE Spaces.

System Action: Command processing on this application server is ended. If the TO clause was used in command, processing continues on subsequent application servers.

Programmer Response: Correct the command by specifying a different package-name or the REPLACE parameter. Reprocess the command for the failed cases. (See "Rerunning DBS Utility" on page 223.)

-617 **You cannot create a(n)** *key-type* **on more than 16 columns.**

Explanation: The maximum number of columns for a **key-type** is 16. **key-type** can be an INDEX, a PRIMARY KEY, or a UNIQUE CONSTRAINT. The statement contains more than 16 columns for the INDEX, PRIMARY KEY, or UNIQUE CONSTRAINT.

SQLSTATE 54008.

System Action: Processing of the SQL statement has ended.

User Response: Correct the SQL statement so that no

more than 16 columns are established for the index, key, or constraint.

-618 **You cannot use the NEW option because the package already exists.**

Explanation: You specified the NEW option on a CREATE PACKAGE statement to create a new package, but the named package already exists.

SQLSTATE 42710.

System Action: Processing of the SQL statement has ended.

User Response: Unless you used the NEW option intentionally to determine whether a package already exists, roll back the logical unit of work, correct the statement, and try again.

-619 **An error was encountered while inserting data into the** *explain-table-name* **on behalf of the EXPLAIN statement. This may be caused by using the old EXPLAIN tables created in a release prior to SQL/DS Version 3 Release 4.**

Explanation: Explanation tables for the EXPLAIN statement may have been created improperly. Check for an incorrect column type, length, or nulls specification.

If there are no errors in the explanation tables, a system error has occurred. If this is the situation, apply the explanation and actions given for SQLCODE -901.

SQLSTATE 55002.

System Action: Processing of the SQL statement has ended.

User Response: Drop and recreate the required tables before using EXPLAIN. You can use a supplied macro (ARISEXP MACRO) to create the EXPLAIN tables. See the *DB2 Server for VSE & VM SQL Reference* manual for details about the ARISEXP MACRO file.

-620 **The CREATE TABLE statement failed because** *user-id* **has no private, recoverable dbspaces with less than 255 tables.**

Explanation: The dbspace name was not specified in the CREATE TABLE statement, so the database manager tried to find a private dbspace owned by **user-id**. This message will appear under one of the following conditions:

1. There were no private dbspaces found in the DB2 Server for VM database for the **user-id**.
2. There were one or more private dbspaces found for the **user-id**, but each contained 255 tables.
3. The private dbspaces were located in nonrecoverable storage pools. Only private dbspaces that exist in recoverable storage pools will be available when the CREATE TABLE statement does not specify the dbspace name.

SQLSTATE 57022.

System Action: Processing of the SQL statement has ended.

User Response: Suggestions for the three conditions above are:

For (1): Acquire a private dbspace in a recoverable storage pool. You may require a database administrator's assistance.

For (2): Drop a table in a private dbspace residing in a recoverable storage pool to free up an entry, or take the action indicated in (1) above.

For (3): If you want to create the table in a dbspace existing in a non-recoverable storage pool, then specify the dbspace name on the CREATE TABLE command. Otherwise, take the action indicated in (1) above.

Then rerun the CREATE TABLE statement.

Database Administrator Response: If appropriate, acquire a private dbspace for the user.

-621 **You have attempted to use invalid index** *index-id* **in dbspace** *dbspace-number***.**

Explanation: A statement made by the user attempted to use an index which has been marked invalid.

SQLSTATE 51013.

System Action: Processing of the SQL statement has ended.

User Response: If you are the owner of the invalid index then drop and recreate the index. The **index-id** can be found in the SYSINDEXES catalog table as follows:

```
SELECT A.DBSPACENO, B.IID, B.INAME,B.ICREATOR, A.TNAME
      FROM SYSTEM.SYSCATALOG A, SYSTEM.SYSINDEXES B
      WHERE A.TNAME = B.TNAME AND A.CREATOR = B.CREATOR
            AND A.DBSPACENO = dbspace_number
            AND B.IID = index_id
      ORDER BY A.DBSPACENO, B.IID   ;
```

Otherwise, notify your database administrator.

Database Administrator Response: Contact the owner of the invalid index and have the owner drop and recreate the index.

-622 **A request to continue collecting statistics while loading table** *owner.table-name* **has failed.**

Explanation: The Database Services Utility made a request to continue collecting statistics while loading table **owner.table-name**. This request failed because the database manager provided statistics as input, but the statistics are not for the table being loaded.

SQLSTATE Spaces.

System Action: Statistics are not collected during subsequent inserts into the table. If this message is received while the Database Services Utility is executing a DATALOAD or RELOAD command, DATALOAD or RELOAD processing is discontinued.

User Response: If you received this message while using the Database Services Utility, reissue the DATALOAD or RELOAD command to continue loading the table. To collect statistics for the specified table, issue an UPDATE STATISTICS command for the table.

-623 **You cannot collect statistics while loading table** *owner.table-name***. Reason Code =** *reason-code***.**

Explanation: The Database Services Utility requested statistics collection to begin for table **owner.table-name** while the utility added rows to the table. **owner** is the owner of the table. **table-name** is the name of the table. The database manager does not support this form of statistics collection for this table, as explained by the reason code:

01 The table has one or more indexes defined on it. This form of statistics collection does not support tables with indexes.

02 The ROWCOUNT column in the SYSCATALOG entry for this table does not have value 0 or -1, which indicates that the table is not empty. This reason can occur only when you have indicated that the value of ROWCOUNT should be respected.

SQLSTATE Spaces.

System Action: Statistics are not collected during subsequent inserts into the table. If you receive this message while executing a Database Services Utility DATALOAD command, the database manager generates table statistics by issuing an UPDATE STATISTICS statement after the Database Services Utility has loaded all the rows.

User Response: To use this form of statistics collection, you must first correct the indicated problem. Alternatively, you can issue an UPDATE STATISTICS statement for the table after all rows have been loaded into it.

-624 **Completion of statistics for table** *owner.table-name* **has failed. Reason Code =** *reason-code***.**

Explanation: The Database Services Utility requested the termination of statistics collection for table **owner.table-name** during the addition of rows to the table. **owner** is the owner of the table. **table-name** is the name of the table. The reason code could be either of the following:

01 Collected statistics did exist, but for a table other than the specified table. This can occur if a previous request to begin collecting statistics had specified a different table.

02 No collected statistics existed for any table. This can occur if no previous request had been made to begin collecting statistics for a table, or if statistics collection had begun but had been discontinued. In the latter case, an appropriate SQLCODE would have been issued to inform you that statistics collection had been discontinued.

SQLSTATE Spaces.

System Action: The database manager did not modify any table statistics. It discarded any statistics that it had collected.

User Response: To get statistics for the specified table, issue an UPDATE STATISTICS statement for the table.

-625 **A request to collect statistics while loading table** *owner.table-name* **has failed. Statistics collection is in progress for some table.**

Explanation: The Database Services Utility made a request to begin collecting statistics for table **owner.table-name** during the addition of rows to the table. **owner** is the owner of the table. **table-name** is the name of the table. This request has failed because statistics are already being collected for some table (this might be the same table). The database manager does not support concurrent collecting of statistics for more than one table by a user. This error occurs only if a previous request to begin collecting statistics did not have a matching request to terminate the collecting of statistics.

SQLSTATE Spaces.

System Action: Statistics are not collected during subsequent inserts into the specified table. The statistics that the database manager has already collected are not affected. If this message is received while you are executing a DATALOAD or RELOAD command, the Database Services Utility stops the DATALOAD or RELOAD processing.

User Response: To collect statistics for the specified table, you must first terminate the statistics collection that is currently underway. If you are using the Database Services Utility, reissue the DATALOAD or RELOAD command.

-629 **SET NULL cannot be specified because foreign key** *constraint-name* **cannot contain null values.**

Explanation: The SET NULL option of the indicated FOREIGN KEY clause is invalid because no column of the key allows null values. The **constraint-name** was specified in the FOREIGN KEY clause or, if a FOREIGN

KEY constraint-name was not specified, it is the generated name.

SQLSTATE 42834.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-630 **Duplicate** *keyword* **keyword.**

Explanation: The SQL statement contains a duplicate specification for the **keyword** keyword.

SQLSTATE 42614.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-632 **The table** *owner1.table-name1* **cannot be defined as a dependent of table** *owner2.table-name2* **because of rule restrictions. Reason Code =** *reason-code***.**

Explanation: The object table specified in the CREATE TABLE or ALTER TABLE statement cannot be defined as a dependent of the indicated table because one of the following reason codes = *reason-code*.

03 You were attempting to define the indicated table as its own dependent. Self-referencing constraints are not allowed.

06 The relationship forms a referential cycle that delete-connects a table to itself.

SQLSTATE 42915.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-640 **The clause or scalar function** *name* **is invalid because mixed and graphic data are not supported on this system.**

Explanation: One of the following has occurred:

- A FOR MIXED DATA clause was specified in an ALTER or CREATE TABLE statement when the package default mixed CCSID was zero.
- A GRAPHIC data type was specified in an ALTER or CREATE TABLE statement when the package default graphic CCSID was zero.
- The VARGRAPHIC scalar function was used when the application server default graphic CCSID, CCSIDGRAPHIC, was zero.

SQLSTATE 56031.

System Action: Processing of the SQL statement has ended.

User Response: Specify the appropriate CCSID for the package, or try the statement on an application server which supports mixed and graphic data.

-641 **You cannot reorganize a key index with a DBSU REORGANIZE INDEX command.**

Explanation: Using the DBSU REORGANIZE INDEX command to reorganize an index that the database manager created to enforce the uniqueness of a primary or unique key is invalid.

SQLSTATE Spaces.

System Action: The Database Services Utility ignores the command.

-644 *name* **is a duplicate name for the same keytype.**

Explanation: **name** cannot be defined as a new key-name for the same keytype because the name has already been defined on the current table.

The duplication can occur if the specified name already exists for the same FOREIGN KEY, PRIMARY KEY, or UNIQUE CONSTRAINT on the current table.

SQLSTATE 42710.

System Action: Processing of the SQL statement has ended.

User Response: Specify a different name on the statement. Try again.

-648 **The foreign key cannot be defined because it will cause table** *owner1.table-name1* **to be delete-connected to table** *owner2.table-name2* **through multiple paths. Reason Code =** *reason-code***.**

Explanation: The foreign key cannot be defined because one of the following reason codes = *reason-code*:

- 01 - The relationship will cause table *owner1.table-name1* to be delete-connected to table *owner2.table-name2* through multiple paths with the same delete rule of SET NULL.
- 02 - The relationship will cause table *owner1.table-name1* to be delete-connected to table *owner2.table-name2* through multiple paths with different delete rules.

SQLSTATE 42915.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-649 **The EXPLAIN table *owner.explain-table-name* was not found in the system catalog.**

Explanation: The EXPLAIN table (*owner.explain-table-name*) does not exist, so the EXPLAIN statement cannot be performed.

SQLSTATE 42704.

System Action: Processing of the SQL statement has ended, and the statement is not recorded in the package.

User Response: Create the EXPLAIN tables, and then resubmit the application to the preprocessor. You can use the IBM-supplied macro (ARISEXP MACRO) to create the EXPLAIN tables. See the *DB2 Server for VSE & VM SQL Reference* manual for details about the ARISEXP MACRO file. The explicit EXPLAIN versions of this SQLCODE are +204 or -204.

-656 **Table *owner.table-name* already has a primary key.**

Explanation: A PRIMARY KEY cannot be defined in an ALTER TABLE statement because the indicated table already has a PRIMARY KEY.

SQLSTATE 42889.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-657 **You cannot issue the UPDATE STATISTICS statement for a view.**

Explanation: An attempt was made to update statistics on a view rather than the base table of the view.

SQLSTATE 42809.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-666 **The LABEL ON statement failed because the row describing the *column-name* column does not exist in SYSTEM.SYSCOLUMNS or SYSTEM.SYSCATALOG.**

Explanation: The row for the label column (TLABEL or CLABEL) was not found in SYSTEM.SYSCOLUMNS. **column-name** is either TLABEL or CLABEL. The TLABEL column resides in SYSTEM.SYSCATALOG and contains correlation names. The CLABEL column resides in SYSTEM.SYSCOLUMNS and contains column labels.

SQLSTATE 42703.

System Action: Processing of the SQL statement has ended.

User Response: Verify that your SQL statement is valid. If you cannot determine the cause of the error, contact your system programmer.

Database Administrator Response: Use SQLERRP and SQLERRD1 in message ARI0504I to determine more information about the error. Verify that the DB2 Server for VM catalog has not been destroyed. If this is a system error, make a record of what went wrong and contact the designated support group for your installation.

-667 **The database manager cannot activate the *key constraint-name* defined for table *owner.table-name*. Reason Code = *reason-code*.**

Explanation: The **key-type** could be PRIMARY KEY, FOREIGN KEY, or UNIQUE CONSTRAINT. For a PRIMARY KEY, **constraint-name** would be omitted. For a FOREIGN KEY, **constraint-name** would be the name of the first key that causes the failure. For a UNIQUE CONSTRAINT, **constraint-name** would be omitted.

The reason code could be one of the following:

- 1 - Your attempt to activate a PRIMARY KEY or UNIQUE CONSTRAINT has failed because the PRIMARY KEY or UNIQUE CONSTRAINT contains duplicate values.

  You have to update or remove the erroneous rows before the PRIMARY KEY or UNIQUE CONSTRAINT can be activated.
- 2 - Your attempt to activate a FOREIGN KEY has failed because the FOREIGN KEY references a parent table that has an inactive PRIMARY KEY.

  You must activate the PRIMARY KEY of the parent table before you can activate the dependent FOREIGN KEY.
- 3 - The operation failed because not every value in the given FOREIGN KEY can be found in the PRIMARY KEY of its parent.

  You have to update or remove the erroneous rows before the key can be activated. The failure could occur when you attempt to activate a FOREIGN KEY, or when you attempt to activate a PRIMARY KEY which implicitly activates other FOREIGN KEYs that reference it.

  For more information see the *DB2 Server for VSE & VM Database Administration* manual.

SQLSTATE 22519.

System Action: Processing of the SQL statement has ended. No keys were added or activated.

User Response: See Explanation for the reason of failure and recovery action. For more information, refer to the *DB2 Server for VSE & VM Database Administration* manual.

-668 **Table *owner.table-name* is inactive and you cannot access it.**

Explanation: No operation can be performed on this table because at least one inactive key participates in a relationship with this table.

This may be a PRIMARY KEY, a FOREIGN KEY, or a dependent FOREIGN KEY which references the PRIMARY KEY.

While an inactive key exists, only the table owner and a DBA can issue DML statements against the table. In addition, no one can issue DML statements against an active table which results in a reference to an inactive table in order to enforce Referential Constraints.

SQLSTATE 57016.

System Action: Processing of the SQL statement has ended.

User Response: Before using the indicated table you must make sure that all the keys in which the table participates are active. Once the keys are active try the statement again.

-669 **You cannot drop a key index explicitly.**

Explanation: The database manager does not allow you to drop a primary key index or unique constraint index explicitly with the DROP INDEX statement. A key can be a PRIMARY KEY or a UNIQUE CONSTRAINT.

SQLSTATE 42917.

System Action: Processing of the SQL statement has ended.

User Response: If you want to reorganize the primary key or unique constraint index, you can drop and recreate the unique constraint, or deactivate and activate the unique constraint.

If you do not want to keep the unique constraint, you can use the DROP UNIQUE clause of the ALTER TABLE statement to remove the constraint and implicitly remove its index.

-670 **You cannot create the foreign key *constraint-name* for table *owner.table-name*. Reason Code = *reason-code*.**

Explanation: The reason code could be one of the following:

1 Your attempt to create a foreign key has failed because the foreign key references a parent table that has an inactive primary key.

2 Your attempt to create a foreign key has failed because not every value in the given foreign key can be found in the primary key of its parent.

For more information, refer to the *DB2 Server for VSE & VM Database Administration* manual.

SQLSTATEs:

23520 Reason code = 2

22521 Reason code = 1

System Action: Processing of the SQL statement has ended. A foreign key is not created.

User Response: The next step depends on the reason code you received:

Reason Code
Action

1 Activate the primary key of the parent table. Then create the dependent foreign key.

2 Update or remove the erroneous rows in the parent table. Then create the dependent foreign key.

-674 **The database manager cannot load the DBCS data conversion routine *transproc-name*. Reason Code = *reason-code(sub-code)*.**

Explanation: The DBCS data conversion routine *transproc_name* cannot be loaded. The *reason-code* has one of the following values:

1 The conversion routine is not available on this installation.

2 The conversion routine was loaded above 16MB when the DB2 Server for VM application server was running AMODE 24. It was dropped by the system.

3 The conversion routine was loaded above 16MB when the DB2 Server for VM application server was running AMODE 24. The system attempted to drop the routine, but was unsuccessful.

4 The conversion routine was loaded above 16MB when the DB2 Server for VM application server was running AMODE 24. It was not loaded by the system, and it was not dropped by the system.

5 Insufficient storage to set up conversion control block.

6 Insufficient storage to load the conversion routine.

7 Other system load error.

The *sub-code* is a one-byte hexadecimal value represented as a hexadecimal string constant. It has one of the following meanings:

X'01' - The error was discovered at the application requester.

X'02' - The error was discovered at the application server.

SQLSTATE 57011

System Action: Processing of the SQL statement has ended.

User Response: Report to your system programmer.

Database Administrator Response:

Reason Code	Action
1	Verify whether the conversion routine name in SYSTEM.SYSSTRINGS is correct, and that the routine has been set up properly according to the *DB2 Server for VM System Administration* manual.
2, 3, or 4	Ensure that the routine has RMODE 24. One way to ensure this is to link edit with the RMODE 24 option. If there is no longer any need for the DB2 Server for VM application server to run AMODE 24 (that is, if all your user exits and single user mode applications support 31-bit adressing), then you can use the AMODE(31) parameter when running SQSTART. The DB2 Server for VM application server's default addressing mode becomes AMODE 31 and your data conversion routine can be loaded above the 16MB line. For 3 or 4, use NUCXDROP to drop the conversion routine *transproc-name*.
5, 6, or 7	Check your installation storage and other system requirements and take the appropriate action.

-681 **Column** *column-name* **is in violation of installation defined field procedure** *fieldproc-name* **Return Code=***nn***, Reason Code=***cccc***, Message=** *string***.**

Explanation: The installation field procedure **fieldproc-name** has returned an error for column **column-name**. Use the return code **nn** to determine the problem.

The following return codes are reserved:

4 Invalid value passed for encoding or decoding, or invalid column description during field procedure definition function.

8 Invalid parameter value.

12 Field procedure error on any function.

Use the **nn** and **string**, which the field procedure returned, for further information.

SQLSTATE 23507.

System Action: Processing of the SQL statement ends.

User Response: If a field procedure did not cause the error, determine the requirements imposed by the field procedure; verify the data type and length of the values to encode (return code 4), or verify the contents of the Field Procedure Parameter Value List (return code 8). If a field procedure caused the error, examine the field procedure (return code 12).

-682 **The database manager cannot load the field procedure** *fieldproc-name***. Reason Code =** *reason-code***.**

Explanation: The field procedure **fieldproc-name** cannot be loaded. The reason code can have one of the following values:

4 The number of field procedures already loaded on this installation has reached its limit of 16.

8 The field procedure is not available on this installation.

10 The field procedure was loaded above 16MB when the DB2 Server for VM application server was running AMODE 24.

12 Insufficient storage.

16 Other system load error.

SQLSTATE 57010.

System Action: Processing of the SQL statement has ended. The application either commits or rolls back to the previous COMMIT; then the application terminates.

User Response: If you get a reason code of 8 check your application, correct it if necessary and try again. If the application is correct and your problem persists, contact your systems programmer. For all other reason codes, contact your systems programmer immediately.

System Programmer Response:

Reason Code	Action
4	Consider replacing multiple field procedures with a single multi-function field procedure containing a procedure list that identifies various functions that may be requested.
8	Verify whether the field procedure requested should be loaded and take the appropriate action.
10	NUCXDROP the field procedure *fieldproc-name*. Ensure that the field procedure has RMODE

24. One way to ensure this is to use the CMS command GENMOD with the RMODE 24 option.

If there is no longer any need for the DB2 Server for VM application server to run AMODE 24, (that is, all your user exits and single-user-mode applications support 31-bit addressing), then you can use the AMODE(31) parameter when running SQLSTART. AMODE 31 will then become the default addressing mode for the DB2 Server for VM application server, and your field procedure can be loaded above the 16MB line.

12 or 16

Check your installation storage and other system requirements and take the appropriate action.

-683 **Column type of column name** *column-name* **is invalid for** *string* **option.**

Explanation: The *string* option can either be FIELDPROC, FOR subtype DATA, or CCSID integer. The following actions are invalid:

- Specifying the FIELDPROC option on column **column-name** whose data type is other than CHAR, VARCHAR, GRAPHIC, or VARGRAPHIC.
- Specifying the FOR subtype DATA option on column **column-name** whose data type is other than CHAR, VARCHAR, or LONG VARCHAR.
- Specifying the CCSID integer option on column **column-name** whose data type is other than CHAR, VARCHAR, LONG VARCHAR, GRAPHIC, VARGRAPHIC, or LONG VARGRAPHIC.

SQLSTATE 42842.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-684 **The length of literal list beginning** *string* **is too long.**

Explanation: The length of literal list beginning with **string**, excluding insignificant blanks and delimiting parentheses, is greater than 254.

SQLSTATE 54012.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-685 **Invalid field type. Column:** *column-name* **in table** *owner.table-name***.**

Explanation: The field description returned by the field procedure field-definition function is invalid. The data type code denotes a long string or the data type has an invalid value.

SQLSTATE 58002.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-686 **Column defined with a field procedure cannot be compared to another column with a different field procedure.**

Explanation: The columns specified are not compatible. Each column has a different field procedure, or only one column has a field procedure. If the columns have the same field procedure, their column CCSID values are different.

SQLSTATE 53043.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Refer to the *DB2 Server for VSE & VM SQL Reference* manual for restrictions on comparing columns defined with a field procedure. Try again.

-687 **Field types incompatible.**

Explanation: Comparing a column defined with a field procedure to the value of a constant or host variable that has an incompatible type is invalid.

SQLSTATE 53044.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Refer to information on predicates definition in the *DB2 Server for VSE & VM SQL Reference* manual for restrictions on comparing columns defined with field procedures. Try again.

-688 **Incorrect data returned from field procedure. Column =** *column-name***. Table** *owner.table-name***. Reason Code =** *reason-code***.**

Explanation: A field procedure for column **column-name** in table **owner.table-name** returned unexpected data.

The reason code can have the following values:

1 Invalid data type returned from field procedure

2 Invalid precision returned from field procedure

3 String returned from field procedure is too long

4 Invalid length returned from field procedure

5 Invalid length of storage returned by field procedure

6 Invalid length of FPPVL returned by field procedure.

SQLSTATE 58002.

System Action: Processing of the SQL statement has ended.

User Response: Correct the field procedure according to the following reason codes, and try again:
- For reason codes 1, 2, 3, and 4: ensure that the field procedure returns values that are consistent with their descriptions.
- For reason code 5: ensure that the storage size is a positive integer.
- For reason code 6: ensure that the length of FPPVL is positive.

-698 **The statement failed because the invalid entities table could overflow.**

Explanation: The statement requires an available entry in the invalid entities table and the total number of invalid and transient indexes is greater than the internal limit. The statement can be CREATE INDEX, REORGANIZE INDEX, or searched UPDATE. These statements mark indexes transient or invalid during processing and therefore require room in the invalid entities table.

SQLSTATE 56004.

System Action: Processing of the SQL statement has ended.

User Response: Try the statement again later. If the error persists then contact your database administrator.

Database Administrator Response: Use the SHOW INVALID operator command to display the invalid entities table. Notify the owners of the invalid indexes and have them drop and recreate or reorganize the invalid indexes to free entries in the table.

If there are no invalid indexes, the error was caused by too many concurrent CREATE INDEX, REORGANIZE INDEX, and searched UPDATE statements. Try the statement again later.

If the problem persists, do one of the following:
- Reduce the number of internal dbspaces. Refer to the *DB2 Server for VM System Administration* manual for more information.
- Increase the size of the directory. Refer to the *DB2 Server for VM System Administration* manual for more information.

-699 **The ACQUIRE DBSPACE statement failed because all storage pools for available dbspaces are full.**

Explanation: Dbspaces are available, but the storage pools in which they reside do not have enough space for you to acquire dbspaces. The shortage can occur if someone deleted all dbextents from a storage pool, and other storage pools do not have available dbspaces.

SQLSTATE 57037.

System Action: Processing of the SQL statement ends.

User Response: Inform your database administrator.

Database Administrator Response: Use the SHOW POOL command to find the full storage pools. Increase the size of your storage pools.

-700 **No available space in storage pool** *pool-number***.**

Explanation: An SQL request requires a physical page (a new page or a shadow page) but the storage pool does not have enough space. If a REORGANIZE INDEX command returned this SQLCODE, the command processing may have marked the index invalid. Use the SHOW INVALID operator command to determine if the index is now invalid.

SQLSTATE 57038.

System Action: Processing of the SQL statement ends.

User Response: Inform your database administrator.

If an index that you are reorganizing is now invalid, either drop and recreate the index or reorganize it again after the database administrator has corrected the storage pool problem.

Database Administrator Response: Increase the size of the storage pool.

-701 **No available space in the dbspace number** *dbspace-number* **for data.**

Explanation: The space allocated for data in the dbspace is full. No more pages are available now. **dbspace-number** is the dbspace number which has run out of space.

SQLSTATE 57011.

System Action: Processing of the SQL statement has ended.

User Response: Notify your system programmer.

System Programmer Response: Check the PCTFREE value for the dbspace (the value in the FREEPCT column in SYSTEM.SYSDBSPACES). If this value is greater than `0`, it can be reduced to `0` by the ALTER DBSPACE statement. This reduction allows for more insertions of data. If PCTFREE is currently `0`, you must delete some data from the dbspace to make room for

the new data. You can delete space using the DROP TABLE statement. You can also, create a new dbspace and use the DBS Utility UNLOAD and RELOAD statements to transfer the information to the new larger dbspace.

Note: If the dbspace referenced in the text is the catalog dbspace (SYS0001), you must do another database generation specifying a larger size for the first dbspace. Then use the DBS Utility UNLOAD and RELOAD commands to transfer the information to the new database.

For more information, refer to the *DB2 Server for VSE & VM Database Administration* and the *DB2 Server for VM System Administration* manuals.

-702 **No available space in the dbspace number** *dbspace-number* **for indexes.**

Explanation: The space allocated in the dbspace for indexes is full. No more pages are available. **dbspace-number** is the dbspace number which has run out of space.

SQLSTATE 57011.

System Action: Processing of the SQL statement has ended.

User Response: Notify your system programmer.

System Programmer Response: You may use the DROP INDEX statement to make room for the new index. If this is not desirable, a new dbspace will have to be created with a larger number of index pages. This is determined by the PCTINDEX value on the ACQUIRE DBSPACE statement. You may specify a larger PCTINDEX value with the same number of pages as the current dbspace or increase the PAGES value to acquire a larger dbspace. Then use the DBS Utility UNLOAD and RELOAD commands to transfer the information to the new dbspace. Then the original statement may be resubmitted.

For more information, refer to the *DB2 Server for VSE & VM Database Administration* manual.

Notes:

1. If the dbspace referenced in the text is the catalog dbspace (SYS0001), you can try reorganizing the indexes for the catalog tables by running the catalog reorganization utility. For more information, refer to the *DB2 Server for VSE & VM Database Administration* manual.
2. If the catalog index reorganization does not solve the problem, you must do another database generation specifying a larger size for the first dbspace. Then use the DBS Utility UNLOAD and RELOAD commands to transfer the information to the new database.

-703 **No available space in the dbspace for header information.**

Explanation: The space allocated for the database manager to record information about the contents of the dbspace is full. No more header pages are available.

SQLSTATE 57011.

System Action: Processing of the SQL statement has ended.

User Response: Notify your system programmer.

System Programmer Response: Check the NHEADER value for the dbspace in the catalog table SYSTEM.SYSDBSPACES. If the value is less than 8, issue an ACQUIRE DBSPACE statement with a larger NHEADER value (less than or equal to 8). Then use the DBS Utility UNLOAD and RELOAD commands to transfer the information to the new larger dbspace. If the NHEADER value is already 8, the dbspace is full. You may either acquire a new dbspace or delete some objects in the current dbspace by using the DROP TABLE or DROP INDEX statements which will free space in the current dbspace. Then the original statement may be resubmitted.

For more information, refer to the *DB2 Server for VSE & VM Database Administration* manual.

Note: If the indicated action does not eliminate the problem, the dbspace referenced in the text is probably the catalog dbspace(SYS0001). You must do another database generation specifying a larger size for the first dbspace. Then use the DBS Utility UNLOAD and RELOAD commands to transfer the information to the new database.

-704 **The ACQUIRE DBSPACE statement failed because no dbspaces are available.**

Explanation: Either:

1. There are no dbspaces available of the requested type and size in the specified storage pool.
2. If the STORPOOL parameter was not specified, then there are no recoverable storage pools having any free dbspaces.
3. The specified storage pool does not exist.

SQLSTATE 57011.

System Action: Processing of the SQL statement has ended.

User Response: For reason 3, enter the SHOW POOL operator command to determine the existing storage pools. Otherwise, notify your system programmer.

System Programmer Response: Determine what type dbspace is needed and perform an ADD DBSPACE operation. For details on ADD DBSPACE, refer to the *DB2 Server for VM System Administration* manual. The

ACQUIRE DBSPACE statement may be resubmitted when a dbspace has been added.

-705 You specified an incorrect value for the parameter, *parameter*.

Explanation: An incorrect value was provided for the parameter specified in the message.

Format 2:

- An incorrect value was specified for the parameter, PCTINDEX or NHEADER.

Format 3:

- An incorrect value was specified for the parameter, PCTFREE or LOCK.

parameter can refer to the parameters associated with the following statements:
ACQUIRE DBSPACE
ALTER DBSPACE
CREATE INDEX
CREATE TABLE PRIMARY KEY
ALTER TABLE ADD PRIMARY KEY.
ALTER TABLE UNIQUE

SQLSTATEs:

37515

42615

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-706 The ACQUIRE PUBLIC DBSPACE statement failed because you cannot acquire *SYSxxxx* from a nonrecoverable storage pool.

Explanation: Public dbspace names beginning with SYS are system dbspaces and cannot be acquired from a nonrecoverable storage pool. The STORPOOL parameter value is for a nonrecoverable storage pool.

SQLSTATE 53060.

System Action: Processing of the SQL statement has ended.

User Response: Resubmit the request with a recoverable storage pool specified on the ACQUIRE DBSPACE STORPOOL parameter or omit the STORPOOL parameter.

System Programmer Response: Same as user action.

-707 The ACQUIRE DBSPACE statement failed because the name you specified was incorrect.

Explanation: The DBSPACE name you specified starts with SYS, but is not in the range SYS0002 to SYS9999.

SQLSTATE 42939.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-708 You cannot ALTER, LOCK, or DROP a public dbspace because you do not have DBA authority.

Explanation: DBA authority is required to DROP, LOCK, or ALTER a public dbspace.

SQLSTATE 42502.

System Action: Processing of the SQL statement has ended.

User Response: Notify your database administrator.

-709 You cannot drop a system dbspace.

Explanation: You tried to drop a dbspace which has a owner of SYSTEM. This is not allowed.

SQLSTATE 42832.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-710 An attempt to access a private dbspace failed because it is in use.

Explanation: The private dbspace you tried to access is now in use.

SQLSTATE 40503.

System Action: Processing of the SQL statement has ended.

User Response: Try again later.

-711 An attempt to access a dbspace failed because it is disabled.

Explanation: You tried to access an object in a disabled dbspace. The dbspace cannot be accessed until it is enabled by the system programmer.

This error condition can occur when accessing a view. When a view is accessed, the package generated at CREATE VIEW time must be retrieved. If the package dbspace (SYS000n) is disabled, the statement will fail. In this case, the SQLCA SQLERRP value is ARIXOVC. Note that if SQLERRP is not ARIXOVC (or ARIXELX), then the error occurred because an underlying table is in a disabled dbspace.

If this error condition also returns the SQLERRP value of ARIXELX, this indicates that the program's package resides in a disabled dbspace and cannot be accessed. SQLERRP is displayed in message ARI0504I. In this case, SQLCA fields SQLWARN0 and SQLWARN6 are

both set to S to indicate a severe error. This condition will cause ISQL, the DBS Utility, and the FORTRAN preprocessor to terminate.

This error condition can occur during preprocessing or during execution of CREATE VIEW if a package dbspace (SYS000n) is disabled.

System Action: Processing of the SQL statement has ended.

SQLSTATE 57011.

User Response: If you must access the disabled dbspace, you must request your system programmer to enable the dbspace. You can continue to access other dbspaces, unless your DB2 Server for VM application's package is in a disabled dbspace.

-712 **The SQL statement failed because internal dbspaces were not available or were too small.**

Explanation: The system may require internal dbspaces to process SQL statements involving ORDER BY, GROUP BY, or joins. Either it was unable to acquire enough internal dbspaces, or the internal dbspaces were not large enough to process the command.

SQLSTATE 57011.

System Action: Processing of the SQL statement has ended.

User Response: If SQLERRD2 = -74, internal dbspaces are returned to the system by a COMMIT or ROLLBACK statement. If you have issued previous SELECT statements without doing a COMMIT statement, you could alleviate the problem by issuing COMMIT statements interspersed with your other SQL statements. Also, notify your system programmer to increase the number of internal dbspaces in the system.

If SQLERRD2 = -77, -78 or -136 notify your system programmer to increase the size of the internal dbspaces.

System Programmer Response: If SQLERRD2 = -74, allocate additional internal dbspaces using the ADD DBSPACE operation (SQLADBSP EXEC) and resubmit the original command.

If SQLERRD2 = -77, -78 and -136, perform an ADD DBSPACE operation (SQLADBSP EXEC) for internal dbspaces with a larger size parameter than previously specified and resubmit the original command.

-713 **Incorrect isolation level value** *value* **specified.**

Explanation: During execution of an application program, an incorrect value for isolation level was found. The only values recognized for this option are:

C For concurrency at the cursor stability level.

R For concurrency at the repeatable read level.

U For concurrency at the uncommitted read level.

Note: If the package was created prior to Version 5 Release 1, you must explicitly repreprocess it before you can specify U for isolation level Uncommitted Read.

SQLSTATE 42815.

System Action: Processing of the SQL statement has ended.

User Response: Assign a correct value to the SQL isolation level variable in the application program. In COBOL, it is SQL-ISL. In other languages, it is SQLISL. Try again.

-714 **Referential constraints cannot span dbspaces in different types of storage pools.**

Explanation: Both the parent and dependent tables of a referential constraint must be in the same type of storage pool; that is, both are in a recoverable or nonrecoverable storage pool.

SQLSTATE 56035.

System Action: Processing of the SQL statement has ended.

User Response: Make sure both tables are in the same type of storage pool. Correct the statement. Try again.

-720 **The initial subcomponent message repository could not be accessed.**

Explanation: During NLS sub-component initialization, code -720 will be returned by the resource adapter if **all** of the following conditions have been met:

- The repository corresponding to the CMS language setting could not be accessed.

  Note: ″accessed″ implies the following:
  1. If a shared segment (saved segment) exists for a particular repository, it could not be found (using FINDSYS) and it could not be loaded (using SEGMENT LOAD);
  2. If a repository saved segment does not exist or it could not be found or loaded, then a NUCXLOAD operation on the repository module failed.
- The repository corresponding to the default language, defined as the first entry in the ARISNLSC MACRO Q file, could not be accessed.
- The repository corresponding to American English (LANGID: AMENG, LANKEY: S001) could not be accessed.

It contains the language id associated with the last language that failed.

SQLERRMT(1:40) has the following meaning:

It contains the language id associated with the last language that failed.

SQLERRD1 has the following meaning:

Code Number	Explanation
1	The module containing the message repository could not be found for a NUCXLOAD.
2	The module containing the message repository could not be NUCXLOADed because of a storage error.
3	The shared segment containing the message repository could not be found/loaded.
4	The language id corresponding to the CMS language could not be found in the link-edited LANGBLK table.

SQLSTATE 57024.

System Action: Initialization is aborted with the message ARI2902E.

User Response: Contact your system programmer.

System Programmer Response: Ensure that the repository corresponding to the CMS language is available, or that the ARISNLSC MACRO Q's first entry has a corresponding available repository, or that the repository for AMENG is available (repository name for the NUCXLOAD is ARIMS001 MODULE). **Available** implies that a discontinuous shared-segment can be accessed and is specified in the file ARISNLSC MACRO Q or a NUCXLOADable module can be found on one of the disks.

-721 **The message file associated with the string or ID specified as an argument to a SET LANGUAGE command cannot be accessed.**

Explanation: While issuing the SET LANGUAGE command (ISQL and operator), code -721 will be returned if the repository associated with the SET LANGUAGE command argument cannot be accessed.

SQLERRD1 has the following meaning:

Code Number	Explanation
1	The SET LANGUAGE argument's corresponding message file module could not be found for a NUCXLOAD.
2	The SET LANGUAGE argument's corresponding message file module could not be NUCXLOADed because of a storage error.
3	The shared segment containing the message repository could not be found/loaded.
4	The SET LANGUAGE argument could not be found in the SQLDBA.SYSLANGUAGE table as a LANGUAGE or LANGID entry or if the entry was found in the preceding table, then the argument's corresponding language id could not be found in the link-edited LANGBLK table.

SQLSTATE 57024.

System Action: Processing of the command or statement has ended with the message ARI2900E.

User Response: Ensure that the SET LANGUAGE argument is valid. For example, you require a language string (LANGUAGE) or language ID (LANGID) that is less than or equal to 40 characters in length. The language string or ID is present in the SQLDBA.SYSLANGUAGE catalog table. If there is no entry in this catalog table for your language, contact the Database Administrator.

System Programmer Response: Ensure that the SET LANGUAGE argument that you have been given, exists in the SQLDBA.SYSLANGUAGE catalog table. If the language string or ID are missing or incorrect in SQLDBA.SYSLANGUAGE, update the catalog table accordingly.

-751 **A stored procedure has been placed in MUST_ROLLBACK state due to SQL operation** *op-name.*

Explanation: A stored procedure issued an SQL operation that forced the DB2 server to roll back the unit of work.

op-name
The SQL operation that caused the DB2 server to be placed in the MUST_ROLLBACK state is one of the following:

CALL
COMMIT
CONNECT
ROLLBACK

SQLSTATE: 38003

System Action: The statement cannot be executed. All further SQL statements are rejected until the SQL application that issued the SQL CALL statement rolls back the unit of work.

User Response: If this message is received by a stored procedure, the stored procedure must terminate. When control returns to the SQL application that issued the SQL CALL statement, the application must roll back the unit of work. No other SQL statement will be accepted.

The stored procedure should be deactivated by issuing STOP PROC with the REJECT option, so that the disallowed statement can be removed from the procedure. After the statement has been removed, issue the START PROC command to activate the stored procedure.

-752 **You cannot specify an SQL CONNECT statement within a logical unit of work.**

Explanation: You attempted to issue a CONNECT statement before issuing a COMMIT or ROLLBACK statement.

SQLSTATE 0A001.

System Action: Processing of the SQL statement has ended.

User Response: Submit a COMMIT or ROLLBACK statement before the CONNECT statement.

-753 **Internal system error starting a logical unit of work.**

Explanation: While attempting to start a logical unit of work an internal processing error was encountered.

SQLSTATE 58004.

System Action: Processing of the SQL statement has ended.

User Response: Notify your system programmer.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

-754 **Database manager internal error while attempting to log user data.**

Explanation: While attempting to log user data, a database error occurred. An internal field contained an incorrect value on the log request.

SQLSTATEs:

58008 DRDA protocal error

58029 internal error while logging user data error

System Action: Processing of the SQL statement has ended.

User Response: Notify your system programmer.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

-756 **You are permitted only one CREATE PACKAGE statement in a logical unit of work.**

Explanation: The program contains more than one CREATE PACKAGE statement in the same logical unit of work.

SQLSTATE 56042.

System Action: Processing of the SQL statement has ended.

User Response: Conclude this logical unit of work with a COMMIT or ROLLBACK statement. Remove the excess CREATE PACKAGE statements. Try again.

-757 **A CREATE PACKAGE statement is not permitted following a statement that modifies a package in the same logical unit of work.**

Explanation: The program contains a CREATE PACKAGE statement in a logical unit of work. In the same logical unit of work, extended dynamic PREPARE or DROP STATEMENT statements have been processed to modify an existing package.

SQLSTATE 56042.

System Action: Processing of the SQL statement ends.

User Response: Roll back the logical unit of work, then correct the placement of the CREATE PACKAGE statement. Try again.

-758 **You cannot use the current statement if a program has been created or changed in the same logical unit of work.**

Explanation: The current statement may not follow in the same logical unit of work, if a DROP STATEMENT statement or Extended PREPARE statement is used.

SQLSTATE 56042.

System Action: Terminate processing of the statement.

User Response: Commit or ROLLBACK the current logical unit of work before trying to execute the current statement.

-759 **The database manager has not committed package** *package-name* **because it is not a modifiable package and it contains no statements.**

Explanation: The current logical unit of work contains a CREATE PACKAGE. No statements were added to this package so that the COMMIT did not cause the package to be created. This will occur for any CREATE PACKAGE when using the DRDA protocol, or if the NOMODIFY option was specified with the CREATE PACKAGE statement.

SQLSTATE 42943.

System Action: The current logical unit of work was rolled back.

User Response: The package created by the CREATE PACKAGE must have statements added to it by executing Extended PREPARE statements.

-760 **Statement cannot be executed in the same logical unit of work where a package is being updated because the authority to execute the statement did not exist when the package was processed again.**

Explanation: The database manager has processed the subject package again. When this processing ended, the authority to execute the current statement did not exist. The resulting limitation on the user is that the current statement might not execute in the same logical unit of work in which the system is updating the package. A previous DROP STATEMENT or Extended PREPARE against the subject package in the current logical unit of work started the update activity.

SQLSTATE 56044.

System Action: Processing of the SQL statement ends.

User Response: Drop the current statement. Replace it after the authority needed to execute the statement has been granted.

-761 **The language preprocessor being run cannot replace package** *package-name***.**

Explanation: The Assembler, C, PL/I, and COBOL preprocessors cannot be used to replace a package created using the FORTRAN preprocessor or the CREATE PACKAGE statement. A package created by using a CREATE PACKAGE statement can be replaced only by using a CREATE PACKAGE statement with the REPLACE option. It cannot be replaced by the language preprocessor now being executed.

Note: This SQLCODE will also be issued if you preprocessed a FORTRAN program in a Distributed Relational Database Architecture (DRDA) environment and then repreprocess it in a non-DRDA environment.

To avoid this situation, you must either first drop the package or consistently preprocess the program in a DRDA or non-DRDA environment.

SQLSTATE Spaces.

System Action: Processing of the SQL statement has ended.

User Response: Either use the appropriate preprocessor or change the preprocessor parameters PREPNAME or USERID and then rerun the preprocessor.

-762 **Your request is incompatible with the target application server. Reason Code =** *reason-code***.**

Explanation: The target application server cannot process your request. The possible reason code is:

01 You are preprocessing a package using a previous release of an SQL/DS or DB2 Server for VM application requester against a remote DB2 Server for VM or SQL/DS application server of a newer release. The two releases are incompatible. If you want to store a package in the target application server, preprocess your program on an application requester of the same release, and then you can run the program from the back release application requester.

SQLSTATE Spaces.

System Action: Execution of the statement has ended.

User Response: See the **Explanation** for details.

-801 **Exception error** *exception-type* **occurred during** *operation-type* **operation on** *data-type* **data.**

Explanation: The database manager had to materialize a view. In doing so, an arithmetic operation in the subselect clause of the view definition has caused the program exception identified by the **exception-type** value.

The explanation for SQLCODE -802 describes the exception-type and data-type values possible.

SQLSTATEs:

22003 Arithmetic overflow or underflow

22012 Division by zero

22502 Anything else.

System Action: Processing of the SQL statement has ended.

User Response: The information shown in the message can be used by the view owner to determine what data used in the view might be causing the program exception.

System Programmer Response: The view owner can use the information shown in the message to determine what data used in the view might cause the program exception. Running just the subselect clause used in defining the view should cause a similar problem, except it produces SQLCODE -802. This information should help you to determine the problem.

-802 **Exception error** *exception-type* **occurred during** *operation-type* **operation on** *data-type* **data, position** *position-number. psw1 psw2***.**

Explanation: An arithmetic operation in the SQL statement has caused the program exception identified by the exception-type value. The two 4-byte fields shown in the message are the contents of the Program Status Word (PSW) when the abend occurred. **exception-type**, **data-type**, **operation-type** **position-number** and **psw1 psw2** may or may not be returned in SQLCA, depending on where the error was detected. The SQLERRD1 field will be -9xx, where xx is the value of the interruption code.

Note: SQLERRD1 would not be set if you are using the Distributed Relational Database Architecture (DRDA) protocol.

The following is a list of **exception-type** values:

- **DATA EXCEPTION** can occur when incorrect operand fields have been detected in a machine code instruction. One of the following has occurred:
  - The sign or digit codes of operands in Decimal Instructions or in CONVERT TO BINARY are not correct.
  - The multiplicand in MULTIPLY DECIMAL has too few high-order zeros.
  - In ADD DECIMAL, COMPARE DECIMAL, DIVIDE DECIMAL, MULTIPLY DECIMAL, or SUBTRACT DECIMAL the operand fields overlap such that the rightmost bytes of the operand fields do not occupy the same place in storage.
  - In ZERO AND ADD DECIMAL, the operand fields overlap, but the rightmost byte of the second operand is to the right of the rightmost byte of the first operand.
- **FIXED POINT OVERFLOW** can occur during any arithmetic operation on either INTEGER or SMALLINT fields.
- **DECIMAL OVERFLOW** exception can occur when one or more non-zero digits are lost because the destination field in any decimal operation is too short to contain the result.
- **DIVIDE EXCEPTION** can occur on a division by zero operation on any numeric field type (DECIMAL, FLOAT, SMALLINT, or INTEGER), on a decimal division operation when the quotient exceeds the specified data-field size, or on a decimal division operation when the conversion of the divisor requires more than 15 significant digits to represent the whole part of the number.
- **MULTIPLY EXCEPTION** can occur on a decimal multiplication operation when the value chosen for conversion requires more than 15 significant digits to represent the whole part of the number. This exception can only occur when both the multiplier and the multiplicand have precision greater than 15.
- **EXPONENT OVERFLOW** can occur when the result characteristic of any floating point operation exceeds the maximum value supported by System/370 architecture (approximately 7.2E+75).
- **EXPONENT-UNDERFLOW EXCEPTION** can occur if the resulting value of a double-precision floating-point arithmetic operation was less than the minimum value supported by System/370 architecture.
- **SIGNIFICANCE EXCEPTION** can occur when an error has occurred in floating-point addition or subtraction. The error is that the fraction resulting from the arithmetic operation in the SQL statement is zero.

The possible **data-types** are INTEGER, SMALLINT, DECIMAL, and FLOAT. The data type displayed in the message indicates the data type of the temporary internal copy of the data which may differ from the actual column or literal data type due to conversions by the database manager. The possible **operation-types** are ADDITION, SUBTRACTION, MULTIPLICATION, and DIVISION. (Unary minus is treated as a multiply by -1.) If the error was in the select-list of an outer SELECT statement then **position-number** will indicate the position of the expression in the select list. If the error was not in the select-list, **position-number** will be zero.

SQLSTATEs:

22003 Arithmetic overflow or underflow

22012 Division by zero

22502 Anything else.

System Action: Processing of the SQL statement ends.

User Response: Check the arithmetic operation and the columns within the statement to determine what data may be causing the program exception. If you cannot determine the cause of the error, contact your system programmer.

Note: If the error occurred in the select-list of an outer subselect, indicator variables can be added to each of the expressions to allow you to get values not in error and retrieve subsequent rows. In that case, the +802 SQLCODE would be issued.

System Programmer Response: Check the arithmetic operation and the columns within the statement to determine what data may be causing the program exception. The exception-type and associated PSW shown in the message should aid in problem determination. Also refer to the *DB2 Server for VSE & VM Database Administration* manual for information on arithmetic operations.

-803 **The statement failed because the INSERT or UPDATE statement causes a duplicate value for a column defined in either a UNIQUE INDEX, PRIMARY KEY, or UNIQUE CONSTRAINT. Index identifier =** *index-id* **Dbspace number =** *dbspace-number* **Reason code =** *reason code*

Explanation: A column was defined with a unique index, primary key, or unique constraint. An attempt was made to update such a column with a value that already exists in that column. This is not allowed.

If a column has a type of VARCHAR or VARGRAPHIC, two column values which differ only by trailing blanks cause duplication.

The index name corresponding to index-id can be found in the SYSINDEXES catalog table as follows:

```
SELECT A.DBSPACENO,B.II.INAME,B.ICREATOR,A.TNAME
   FROM SYSTEM.SYSCATALOG A, SYSTEM.SYSINDEXES B
      WHERE A.TNAME = B.TNAME AND A.CREATOR = B.CREATOR
          AND A.DBSPACENO = dbspace-number
          AND B.IID = index-id
      ORDER BY A.DBSPACENO,B.IID
```

The reason code provides additional information about the error.

0 The statement executed was one of the following:

- An INSERT
- A positioned UPDATE
- A searched UPDATE where at most one row can satisfy the search condition.

1 The statement executed was a searched UPDATE on a table in a recoverable storage pool whose search condition could be satisfied by more than one row. Uniqueness checking was deferred until all rows were updated but all duplicates were not resolved by the end of statement processing.

2 The statement executed was a searched UPDATE on a table in a nonrecoverable storage pool whose search condition could be satisfied by more than one row. Uniqueness checking was not deferred and a duplicate was detected before all rows were updated.

SQLSTATE 23505.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-804 **Either the structure or the section number in the RDIIN control block with call type** *call-type* **is incorrect.**

Explanation: The database manager found an incorrect call type in the RDIIN structure.

SQLSTATE 07002.

System Action: The processing of this call to the database manager ends.

User Response: If your program generates its own RDIIN structure or modifies an RDIIN structure that DB2 Server for VM preprocessors generated, ensure that a valid call type is in the RDIIN.

If your program does not generate its own RDIIN or modify an existing RDIIN, a database manager error has occurred. Notify your system programmer.

System Programmer Response: Record the error and contact the designated support group for your installation.

-805 **The package** *owner.package-name* **is not in the SYSTEM.SYSACCESS catalog table.**

Explanation: When the system tried to run the program, it could not find the package in the catalog table SYSTEM.SYSACCESS.

SQLSTATEs:

42704 On a GRANT, REVOKE, or DROP statement (for example, when directly naming the package)

51002 On any other statement (for example, when indirectly attempting to access a package that is not at the application server).

System Action: Processing of the SQL statement has ended. The application can issue subsequent SQL statements.

User Response: Be sure that the program has been preprocessed with no errors. If not, correct any errors which caused the preprocess to fail, preprocess the program and rerun it.

If you are using multiple databases, you may use the CONNECT statement to switch to a different database to do some other work.

-806 **The package** *owner.package-name* **is unusable.**

Explanation: The package is unusable because it depends on some name (table, view, index, etc.) not now defined, or some authorization is required which is not granted now.

SQLSTATE Spaces.

System Action: Processing of the SQL statement has ended. The logical unit of work (if any) is rolled back.

SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: Determine the name or authorization that is required and submit necessary statements to correct the error. If a change is made to the program, it must be preprocessed.

If you are using multiple databases, you may use the CONNECT statement to switch to a different database to do some other work.

If you are using CMS work units, you may switch to another work unit to do some work. If the CMS work unit in which the error occurred is reestablished at any time, errors may result and the work unit status will be unknown. If a severe error has occurred, only a CONNECT statement will be accepted.

-807 **The maximum number of active packages for an agent is exceeded. Maximum allowed number is** *npackage*.

Explanation: The user has link edited more than the permissible number of programs together and has executed an SQL statement from each of them without an intervening COMMIT or ROLLBACK. This means that the user has more than the permissible number of packages active at the same time. The maximum number of active packages allowed is the same as the value of the initialization parameter NPACKAGE.

SQLSTATE 54017.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response:

1. Redesign or restructure your programs so that no more than the maximum permissible number of packages are involved during execution; or ask your database administrator to increase the value of the NPACKAGE parameter on the SQLSTART EXEC.

   If the value of the NPACKAGE parameter changes, using the new value, re-run the SQLSTART EXEC.
2. Issue a COMMIT at an appropriate point during execution, before the program which causes this error executes.

If you are using multiple databases, you may use the CONNECT statement to switch to a different database to do some other work.

If you are using CMS work units, you may switch to another work unit to do some work. If the CMS work unit in which the error occurred is reestablished at any time, errors may result and the work unit status will be unknown.

-809 **The statement failed because an attempt was made to insert a row longer than allowed.**

Explanation: The row was longer than allowed for a table. Only 4080 bytes of data can be placed in a row, unless a long string is involved. A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(*n*) where *n* is greater than 254 but less than or equal to 32767, or VARGRAPHIC(*n*) where *n* is greater than 127 but less than or equal to 16383.

This error is not detected at the time the table is created. Rather, the error is issued the first time that an attempt is made to insert data into the table.

SQLSTATE 54018.

System Action: Processing of the SQL statement has ended.

User Response: Drop the table. Recreate the table so that the total length of a row is reduced. Try again.

-810 **The statement failed because more than one row was returned for a SELECT clause when only one is allowed. (Subquery or non-cursor SELECT).**

Explanation: A cursor SELECT is required to return more than one row.

The contents of any output host variables are unpredictable.

SQLSTATE 21000.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-811 **The use of this statement is invalid against a section created by a PREPARE SINGLE ROW statement.**

Explanation: The only statements that can be executed against a section created by the PREPARE SINGLE ROW form of the EXTENDED PREPARE statement are Extended EXECUTE statements specified with the OUTPUT DESCRIPTOR clause and Extended DESCRIBE statements.

SQLSTATE 07007.

System Action: Processing of the SQL statement has ended.

Programmer Response: Correct the statement. Try again.

-813 You cannot lock a view.

Explanation: An attempt was made to lock a view. A user may only lock a DBSPACE or a table.

SQLSTATE 42809.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-814 You cannot lock a system dbspace or catalog table.

Explanation: A dbspace with owner of SYSTEM or a catalog table cannot be locked.

SQLSTATE 42832.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-815 CONNECT denied by accounting user exit routine.

Explanation: An attempt to connect to the application server was rejected by the installation's accounting user exit routine. Note that this can occur on either an explicit or implicit

SQLSTATE 42505.

System Action: The application server denies the connection and returns to the application. SQLERRD1 contains the return code from the installation's accounting user exit routine.

User Response: Contact the database administrator to verify the authorization name which was rejected.

-816 Error converting a date/time value in the local exit routine *exit-name***.**

Explanation: The local date/time exit **exit-name** encountered an error when converting a date/time value from/to the local format.

SQLERRD5 contains the function code with the following values.

- 4 - Convert from local format to ISO format.
- 8 - Convert from ISO format to local format.

SQLERRD1 contains the return code set by the local exit.

SQLSTATE 58002.

System Action: The statement is not executed.

User Response: Contact your system administrator about this problem.

-818 The preprocessor-generated consistency token in the load module is different from the consistency token in the package.

Explanation: At preprocessing time, the database manager creates a consistency token and stores it, for run time use, in the source program (to be stored later in the object program) and in the package. At run time, the consistency token within the package is compared with the consistency token in the object program being executed. If the two consistency tokens do not match, the package and the program were not the result of the same preprocessing invocation.

This problem can occur if the owner of an application changes the program (in other words, adds or deletes SQL statements), repreprocesses and recompiles the application, and someone else tries to run the program using the old object code.

SQLSTATE 51003.

System Action: Processing of the SQL statement ends.

User Response: Preprocess and compile the program again and rerun it, or rerun the program using the object code that matches the package.

-821 The PREPARE or EXECUTE IMMEDIATE statement failed because the associated SQL statement has an incorrect length.

Explanation: The SQL statement that was sent over to the database manager on a PREPARE or EXECUTE IMMEDIATE statement at execution time contained an incorrect length. The SQL statement was expected to be in the form of a variable length character string with length of from 0 to 8 192 bytes.

SQLSTATE 22501.

System Action: Processing of the SQL statement has ended.

User Response: Ensure that the SQL statement that is being used by the PREPARE or EXECUTE IMMEDIATE statement is a variable length character string with a length of from 0 to 8 192.

For a host variable declared as a fixed length character string variable in PL/I or in FORTRAN, ensure that the length is correct.

-823 The statement failed because you are not permitted to perform this operation on a catalog table.

Explanation: This code is issued when you attempt to do any of the following:

- Delete rows from certain catalog tables.
- Update rows in certain catalog tables.
- Insert rows into certain catalog tables.

- Alter any catalog table other than SYSTEM.SYSLANGUAGE.
- Reference a catalog table other than SYSTEM.SYSLANGUAGE as the parent table of your foreign key.
- Reorganize a system-created index on any catalog table other than SYSTEM.SYSLANGUAGE using the DBS Utility REORGANIZE INDEX command.

Only someone with DBA authority may issue UPDATE, INSERT and DELETE statements against catalog tables whose columns may be altered. For a list of alterable catalog tables, see the *DB2 Server for VSE & VM SQL Reference* manual.

SQLSTATE 42832.

System Action: Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

-824 Resource Adapter internal error.

Explanation: This is a DB2 Server for VM system error. An incorrect internal modify cancel request was made, or an internal error occurred during blocking of a FETCH or PUT request. SQLERRD1 contains further information about the error. The possible values are:

-4 A pointer used to obtain addressability to an internal control block is zero.

-5 The calltype of the statement passed is not P (for PUT) or F (for FETCH).

24 Invalid parameter list passed in the CMS IMMCMD macro.

44 The immediate command being cleared was not found by CMS.

48 The specified immediate command is a nucleus extension and cannot be cleared.

60 The Modify Cancel Request CALLTYPE is not S or R.

80 Modify cancel support processing found that the pointer to the command name was zero.

SQLSTATE 58004.

System Action: Processing of the SQL statement has ended.

If the value of SQLERRD1 is less than zero, a severe error has occurred. The logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: Notify your system programmer. After a severe error has occurred, only a CONNECT statement will be accepted.

System Programmer Response: Make a record of what went wrong. Ensure that the SQLERRD1 value is included in the information. Contact the designated support group for your installation.

-825 An error occurred while processing an operator command.

Explanation: An error occurred while processing an operator command. SQLERRD1 contains further information about the errors. The possible values are:

-65 Operator commands are not valid in a LUW.

-120 Operator commands are not valid in SUM.

+61 Length of operator command or parameter exceeds 8 characters. Refer to message ARI0061E for detailed information.

+63 Operator command has not been specified. Refer to message ARI0063I for detailed information.

+64 Operator command is invalid. Refer to message ARI0064E for detailed information.

+66 Unexpected error such as: invalid parameter or I/O error. Refer to message ARI0066E for detailed information.

+67 Invalid SHOW command. Refer to message ARI0067E for detailed information.

+68 Invalid SET command. Refer to message ARI0068E for detailed information.

SQLSTATE 51010.

System Action: database operator command processing ends.

User Response: Ensure that the command is correct and try again. However, if the SQLERRD1 is +66 and the problem persists, notify your system programmer.

System Programmer Response: Refer to the error messages returned to the user and the operator, and take the appropriate corrective action.

-826 The statement identifier *statement-id* **in package** *owner.package-name* **is not valid because:** *reason-code***.**

Explanation: The **statement-id** of an Extended DESCRIBE, DECLARE cursor, or EXECUTE is not valid for one of the following reasons:

1. **statement-id** does not exist in the package named.
2. **statement-id** is not executable because it was deleted with a DROP STATEMENT.
3. **statement-id** is not a SELECT, but a query operation is implied by the current statement. (The current statement is an extended DESCRIBE or DECLARE CURSOR).

4. **statement-id** is a SELECT, but a query operation is not implied by the current statement. (The current statement is an extended EXECUTE).
5. **statement-id** is not a SELECT statement, but a SELECT statement is required. (The current statement is an extended PREPARE SINGLE ROW...).
6. **statement-id** is not a SINGLE ROW SELECT (prepared with a PREPARE SINGLE ROW form of PREPARE). (The current statement is an extended EXECUTE...USING OUTPUT DESCRIPTOR).

SQLSTATEs for reason codes 1 and 2:

07003 EXECUTE

26501 DECLARE CURSOR, DESCRIBE, DROP STATEMENT, PREPARE – Filling empty section.

SQLSTATEs for other reason codes:

07005 Reason code 3

07003 Reason code 4

26507 Reason code 6

26508 Reason code 5.

System Action: Execution of the SQL statement has ended.

User Response: Rollback the logical unit of work, then correct the error. Try again.

-827 **Extended dynamic statements are permitted only for package created by the CREATE PACKAGE statement. Package** *owner.package-name* **was not created by CREATE PACKAGE.**

Explanation: You can use extended dynamic statements only to execute or modify a package that a CREATE PACKAGE statement has created.

SQLSTATE 42862.

System Action: Processing of SQL statement ends.

User Response: Roll back the logical unit of work, correct the error, and rerun the program.

-828 **Package** *owner.package-name* **was created using the NODESCRIBE option of the CREATE PACKAGE statement. Therefore, you cannot use the DESCRIBE statement for it.**

Explanation: You used the NODESCRIBE option when creating the package with the CREATE PACKAGE statement.

SQLSTATE Spaces.

System Action: Processing of the SQL statement has ended.

User Response: Roll back the logical unit of work, then correct the error. Try again.

-829 **You cannot use a CREATE PACKAGE statement with the REPLACE option to replace package** *owner.package-name* **which was created with the MODIFY option.**

Explanation: When the package was originally created, the MODIFY option was specified. The current CREATE PACKAGE statement is incorrect because it contains the REPLACE option. A modifiable package may not be replaced.

SQLSTATE 56046.

System Action: Processing of SQL statement has ended.

User Response: Rollback the logical unit of work, then correct the error. Try again.

-830 **You cannot use an Extended PREPARE or DROP STATEMENT to change package** *owner.package-name***.**

Explanation: You are not allowed to add or delete a statement from the package either because:

- The package was created with the NOMODIFY option, or
- The package was created with the MODIFY option but you are running the program under the DRDA protocol which does not support these two statements.

SQLSTATE 42859.

System Action: Processing of the SQL statement ends.

User Response:

- If the NOMODIFY option was specified on the CREATE PACKAGE, roll back the logical unit of work, correct the error, and rerun the program.
- If the MODIFY option was specified, but the application server is not a DB2 Server for VM database, you need to rewrite your program to not add or delete a statement from a package. If the application server is a DB2 Server for VM database, then the PROTOCOL(AUTO) or PROTOCOL(SQLDS) option should not be used. You should reinvoke SQLINIT with other PROTOCOL values.

-831 **A package created by the CREATE PACKAGE statement can be the target of the following Extended Dynamic Statements only: EXECUTE, DECLARE CURSOR, OPEN, FETCH, PUT, CLOSE, DROP STATEMENT, and DESCRIBE.**

Explanation: The only SQL extended dynamic statements that might reference or modify a package

created using the CREATE PACKAGE statement are:
- Extended DECLARE CURSOR
- Extended DROP STATEMENT
- Extended OPEN, FETCH, PUT, and CLOSE
- Extended EXECUTE
- Extended DESCRIBE.

Any other SQL statements referencing a CREATE PACKAGE package are invalid.

SQLSTATE Spaces.

System Action: Processing of the SQL statement has ended.

User Response: Roll back the current logical unit of work, correct the problem, and rerun the preprocessor.

-832 **A CREATE PACKAGE statement has been issued in the current logical unit of work. The database manager cannot execute the current statement until you commit or roll back the current logical unit of work.**

Explanation: After you have issued a CREATE PACKAGE, rules apply to the execution of SQL statements that can follow in the same logical unit of work. Those rules are:

- If the CREATE PACKAGE contains the NOMODIFY option, only Extended PREPARE SQL statements can follow.
- If the CREATE PACKAGE contains the MODIFY option, the SQL statements that follow can only reference the package named in the CREATE PACKAGE statement, and must be one or a combination of the following SQL statements:
  Extended PREPARE
  Extended EXECUTE
  Extended DECLARE Cursor
  Extended OPEN
  Extended PUT
  Extended FETCH
  Extended CLOSE
  Extended DESCRIBE
  DROP STATEMENT.

After a COMMIT or ROLLBACK statement is issued, these rules no longer apply.

SQLSTATE 56042.

System Action: Processing of the SQL statement has ended.

User Response: Modify the user program so that the statement in error is executed after the package has been committed or rolled back. A package is committed with the COMMIT statement and rolled back with the ROLLBACK statement.

-833 **The package or view, *name*, cannot be rebound because the character set under which it was preprocessed, *charset1*, does not coincide with the character set under which the database manager is running, *charset2*.**

Explanation: The database manager is rebinding package or view *name*. The package or view was preprocessed with the character set *charset1*. The database manager is currently running using character set *charset2*, which is different from the preprocessed character set.

Note that if the value of the column 'VALID' in SYSTEM.SYSACCESS is manually changed to 'N' by a user for a view package, this error will be received on all subsequent invocations of the view.

SQLSTATE 51016.

System Action: Rebinding is terminated and processing of the SQL statement has ended.

User Response:

1. Preprocess the package again.If it is the DBS Utility package (SQLDBA.ARIDSQL), use the ARISDBMA EXEC to recreate it. See the *DB2 Server for VM Program Directory* for details on the ARISDBMA EXEC.

2. Contact the database administrator. End the application server and restart, supplying a character set name in the CHARNAME parameter that is identical to the character set that was used to first preprocess the package.

Packages preprocessed using either the ENGLISH or the INTERNATIONAL character set never encounter this error. They are always rebound in ENGLISH or INTERNATIONAL, respectively.

-834 **The NULL form of the Extended PREPARE statement does not follow a CREATE PACKAGE...USING NOMODIFY statement in the current logical unit of work.**

Explanation: The user application is executing an Extended PREPARE statement using the NULL option. In the current logical unit of work, this statement does not follow a CREATE PACKAGE statement of the form: CREATE PACKAGE...USING NOMODIFY... . Therefore, the database manager does not allow it to be executed. The database manager enforces an execution sequence rule for the NULL form of the Extended PREPARE: it must be in the same logical unit of work as CREATE PACKAGE of the form: CREATE PACKAGE...USING NOMODIFY... ; and it must follow the CREATE PACKAGE in the execution sequence.

SQLSTATE 56047.

System Action: Processing of the SQL statement has ended.

User Response: Arrange the execution sequence of the NULL form of the Extended PREPARE so that it follows, in the same logical unit of work, a CREATE PACKAGE of the form: CREATE PACKAGE... USING NOMODIFY... .

-835 **The database manager cannot rebind the view** *owner.view-name* **because this view was created in a previous version of the database system.**

Explanation: When the database manager is running an application for the first time, and the application uses a view that was created before the current version of the application, the database manager rebinds the CREATE VIEW statement of the view.

An unexpected error can occur during the rebinding. In most instances, this is because the view definition included a SELECT * and the table on which the view is based had columns added to it since the view was originally created. In this instance, SQLERRD1 is equal to -110. Otherwise, SQLERRD1 is set to the SQLCODE that indicates the error detected during view repreprocessing.

When the application server default CHARNAME value has been changed and a view has a dependency on a system table, the view is no longer valid. In this case, SQLERRD1 is set to -833.

SQLSTATE 56049.

System Action: Processing of the SQL statement has ended.

User Response: Drop the view and recreate it before rerunning the application. Contact your database administrator to have this done.

-836 **The current statement cannot be executed using the DRDA protocol.**

Explanation: The current statement is not defined in the DRDA protocol. The following is a list of the statements that are not supported when the DRDA protocol is used.

- Extended PREPARE SINGLE ROW (In FORTRAN this is SELECT INTO.)
- Positioned UPDATE or positioned DELETE in FORTRAN.
- Extended EXECUTE...USING OUTPUT DESCRIPTOR
- CONNECT (userid) IDENTIFIED BY (password)
- DB2 Server for VM Operator Commands
- Extended Dynamic statements for modifiable packages

SQLSTATE 42612.

System Action: Processing of the SQL statement has ended. The logical unit of work (if any) has been rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: Preprocess and execute the program against the database application server and do not use the DRDA protocol. If you must execute against the non-DB2 Server for VM application server, a program should be rewritten to avoid these statements. For example, if the current statement is a PREPARE SINGLE ROW, replace it with a Basic PREPARE and use cursor operations to retrieve the required data.

-841 **A communications directory error has occurred. Reason Code= X'***reason-code***'.**

Explanation: The database manager detected an error with the communications directory when it tried to resolve the server name.

The **reason-code** value is the 2-byte hexadecimal error code. Possible reasons for the error:

- X'0001' - System communications directory is undefined.
- X'0002' - System communication directory is inactive.
- X'0003' - System communication directory does not have a file type of NAMES.
- X'0004' - User communications directory is undefined.
- X'0005' - User communication directory is inactive.
- X'0006' - User communications directory does not have a file type of NAMES.
- X'0007' - DBNAME is not found in system communications directory.
- X'0008' - DBNAME is not found in user communications directory.
- X'0009' - Invalid nickname is found in communications directory.
- X'000A' - Invalid address information found in communications directory. Check LUNAME or TPN if attempting an SNA connection, check SERVICE or HOST if attempting a TCP/IP connection.
- X'000B' - System communications directory file is not found.
- X'000C' - User communications directory file is not found.
- X'0012' - User communications directory file is unloaded.
- X'0013' - System communications directory file is unloaded.
- X'0014' - A DBNAME greater than 8 characters was not found in the communications directory when it tried to resolve the DBNAME, or the database manager in a VM environment does not support DBNAME longer than 8 characters.
- X'0015' - SNA and TCP/IP tags are mutually exclusive. *tpn* and *luname* are SNA tags. *host* and *service* are TCP/IP tags. If *tpn* or *luname* are specified, then *host* or *service* cannot be specified. If *host* or *service* are specified, then *tpn* or *luname* cannot be

specified. Determine whether SNA or TCP/IP communications should be used and correct the user communications directory.

- X'0016' - SNA and TCP/IP tags are mutually exclusive. *tpn* and *luname* are SNA tags. *host* and *service* are TCP/IP tags. If *tpn* or *luname* are specified, then *host* or *service* cannot be specified. If *host* or *service* are specified, then *tpn* or *luname* cannot be specified. Determine whether SNA or TCP/IP communications should be used and correct the system communications directory.
- X'0017' - TCP/IP Service name greater than 16 characters in the user communications directory.
- X'0018' - TCP/IP Service name greater than 16 characters in the system communications directory.
- X'001E' - System error occurred when processing communications directory.

SQLSTATE:

42705 For reason code 14.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: After a severe error has occurred, only a CONNECT statement will be accepted.

Correct the communications directory problem and try to connect again.

-844 **An Extended PREPARE can only be executed using the DRDA protocol if it has an input SQLDA.**

Explanation: You cannot PREPARE an SQL statement with host parameter markers without specifying input SQLDA when using the DRDA protocol.

SQLSTATE 56041.

System Action: Processing of the SQL statement has ended.

User Response: Execute the program using PROTOCOL (AUTO) when the database manager is accessed. If you must execute against a non-DB2 Server for VM database, the program should be rewritten to include the SQLDA information for the parameter markers specified in the PREPARE statement.

-860 **The SQL statement was not processed because the application server** *server-name* **is not accepting connections from DRDA-only application requesters.**

Explanation: An implicit or explicit connect request has failed. An application requester that uses the DRDA protocol cannot be connected to an application server that only supports the DB2 Server for VM protocol.

SQLSTATE 56077.

System Action: Processing of this SQL statement has ended. The resource adapter returns to the application with SQLCA fields SQLWARN0 and SQLWARN6 both set to S, indicating a serious system error.

User Response: If the application does not require that the DRDA protocol be used, then issue either SQLINIT PROTOCOL(AUTO) or SQLINIT PROTOCOL(SQLDS) to change the application requester's protocol to one that the application server may accept.

-861 **Neither protocol option AUTO nor DRDA can be specified because the DRDA facility has not been installed for the application requester.**

Explanation: The SQLINIT parameter PROTOCOL is set to either AUTO or DRDA. However, the DRDA facility has not been installed for the application requester.

SQLSTATE 56079

System Action: Processing of the SQL statement ends. The resource adapter returns to the application with the SQLCA fields SQLWARN0 and SQLWARN6 both set to S, indicating a serious system error.

User Response: Change the SQLINIT parameter PROTOCOL to SQLDS if you do not need to use either AUTO or DRDA. Otherwise, contact your database administrator to install the DRDA facility.

Database Administrator Response:

Determine whether or not you require your application requester to initiate DRDA conversation. If so, install the DRDA facility following the steps in the *DB2 Server for VM System Administration* manual. If not, specify PROTOCOL(SQLDS) on the SQLINIT EXEC.

-900 **The SQL statement cannot be executed because the previous error has left the application in an unconnected condition.**

Explanation: A previous failure has placed the application process in a connectable and unconnected state. The only SQL statement that can be executed from the connectable and unconnected state is CONNECT.

SQLSTATE 08003.

System Action: Processing of the SQL statement ends and SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: After a severe error occurs, enter a CONNECT statement only.

-901 **The SQL statement failed because of a system error. You can continue to use the database manager.**

Explanation: An error has occurred which is due to system error.

SQLSTATE 58004.

System Action: Processing of the SQL statement has ended.

User Response: Verify that your SQL statement is valid. If you cannot determine the cause of the error, contact your system programmer.

SQLERRD2 in message ARI0504I or in the SQLCA is the DBSS return code which may further define the error that occurred. The following SQLERRD2 codes are considered to be limit or user errors, not system errors:

- -63 = 'SCHEDULE' (for example, CONNECT) operation requested from an active LUW
- -64 = Database log space exhausted
- -65 = Operation not valid in an active LUW
- -92 = Checkpoint in progress, try later
- 137 = Duplicate found during sort of unique index, unique constraint, or primary key

For a complete list of possible SQLERRD2 values, see "Chapter 18. DBSS Return Codes" on page 271.

System Programmer Response: Use SQLERRP and SQLERRD1 in message ARI0504I to determine more information about the error. Verify that the database catalog has not been destroyed. If this is a system error: Make a record of what went wrong and contact the designated support group for your installation.

-902 **Serious system error. Discontinue further use of the database manager.**

Explanation: An unrecoverable error occurred.

SQLSTATE 58005.

System Action: Processing of the SQL statement has ended. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred. If a severe error has occurred, only a CONNECT statement will be accepted.

User Response: Notify your system programmer.

If you are using multiple databases, you may use the CONNECT statement to switch to a different database to do some other work.

If you are using CMS work units, you may switch to another work unit to do some work. If the CMS work unit in which the error occurred is reestablished at any time, errors may result and the work unit status will be unknown.

System Programmer Response: Use SQLERRP and SQLERRD1 in message ARI0504I to determine more information about the error. Make a record of what went wrong and contact the designated support group for your installation.

-903 **The current logical unit of work has been rolled back because of a system failure.**

Explanation: An error has occurred that may be due to a user error or a real system error. The catalog may be destroyed. Refer to **User Response**.

SQLSTATE 40003.

System Action: Processing of the SQL statement has ended. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred. If a severe error has occurred, only a CONNECT statement will be accepted.

User Response: Notify your system programmer.

If you are using multiple databases, you may use the CONNECT statement to switch to a different database to do some other work.

SQLERRD2 in message ARI0504I or in the SQLCA is the DBSS return code which may further define the error that occurred. The following SQLERRD2 codes are considered to be limit or user errors, not system errors:

- -63 = 'SCHEDULE' (for example, CONNECT) operation requested from an active LUW
- -64 = Database log space exhausted
- -65 = Operation not valid in an active LUW
- -92 = Checkpoint in progress, try later
- 137 = Duplicate found during sort of unique index, unique constraint, or primary key

For a complete list of possible SQLERRD2 values, see "Chapter 18. DBSS Return Codes" on page 271.

System Programmer Response: SQLERRD2 in message ARI0504I is the DBSS return code if SQLERRD2 is not zero. For more information about the error, see "Chapter 18. DBSS Return Codes" on page 271. If SQLERRD2 is zero, a database system error has occurred. In that case, make a record of what went wrong and contact the designated support group for your installation.

-906 **The current program was cancelled because of a system error. The program exception is as follows:** *exception-type psw1 psw2*

Explanation: While the system was executing a package, a program exception occurred.

SQLSTATE 58023.

System Action: Processing has ended.

User Response: Notify your system programmer.

System Programmer Response: The exception-type and associated PSW shown in the message should aid in problem determination. If a database system error occurred, make a record of what went wrong and contact the designated support group for your installation.

-911 **The current logical unit of work was rolled back because of a deadlock or timeout. It was waiting for a** *lock-level* **lock in dbspace =** *dbspace-number* **held by user** *sql-userid***. Reason code =** *code***.**

Explanation: The current unit of work was involved in an unresolved contention for use of an object and had to be rolled back.

The reason codes are as follows:

2 transaction rolled back due to deadlock.

68 transaction rolled back due to lock timeout.

The value of *lock-level* can be any one of DBSPACE, TABLE, INDEX PAGE, PAGE, INDEX KEY, or ROW.

SQLSTATE 40001.

System Action: Processing has ended for this logical unit of work and all changes to the database have been rolled back.

User Response: If a deadlock occurred, resubmit your current job at a later time.

If a lock timeout occurred, determine if the lock timeout value is too short or resubmit your current job at a later time.

Operator Response: If a lock timeout occurred and it is determined that the lock timeout value is too short, the lock timeout value can be increased by using the SET LTIMEOUT *new-time* command.

-912 **The current logical unit of work was rolled back because of excessive (system-wide) lock requests.**

Explanation: The system has encountered too many system lock requests and has ended processing for this logical unit of work.

SQLSTATE 57028.

System Action: Processing has ended for this logical unit of work and all changes to the database have been rolled back.

User Response: Resubmit your current SQL application at a later time.

-913 **Unsuccessful execution caused by timeout. Reason code** *reason-code***, type of resource** *resource-type***, and resource name** *resource-name***.**

Explanation: A CALL statement has timed out while waiting for a stored procedure server. The startup parameter PTIMEOUT determines the maximum time the database manager will wait for a stored procedure server to be assigned for an SQL CALL statement.

Possible reason codes are:

1 An agent has exceeded the allowed time to wait for a stored procedure server to become available.

2 A stored procedure server failed to start within the allowed time.

SQLSTATE: 57033

System Action: The statement terminates.

User Response: Run the application again when the stored procedure servers are less busy. Consider rebalancing the stored procedure server workload, or defining additional stored procedure servers.

-914 **A ROLLBACK of the current logical unit of work was scheduled because of a user CANCEL request.**

Explanation: A CANCEL request has been issued. The CANCEL could be initiated by the user or by the system. The CANCEL initiated by the system is a result of reaching the limit on lock request blocks.

SQLSTATE 57014.

System Action:

- The ROLLBACK is scheduled.
- The SQLCODE is returned to the application.
- A new communication path to the database manager will be connected on the next SQL request. If an explicit SQL CONNECT was issued prior to the CANCEL request, the CANCEL process causes that explicit connect to be reset to the default user ID, password, and server name values.

User Response: You may continue to use SQL.

If the user did not issue a CANCEL request, then retry at a later time and if the operation is still receiving an SQLCODE of -914, contact your system programmer about increasing the NLRBU or the NLRBS parameters when starting the application server.

-915 **The current logical unit of work was rolled back because of excessive locks held for this logical unit of work.**

Explanation: The user has issued too many lock requests within the logical unit of work.

SQLSTATE 57029.

System Action: Processing has ended and all changes to the data base have been rolled back.

User Response: Issue a COMMIT statement between lock requests and rerun.

-916 **The current logical unit of work was rolled back because of operator action.**

Explanation: The database virtual machine operator entered the FORCE command to cause ROLLBACK of the current logical unit of work.

SQLSTATE 57027.

System Action: ROLLBACK logical unit of work.

User Response: You may continue to use SQL.

-917 **The package was not created and the logical unit of work was rolled back due to a system error.**

Explanation: During preprocessing or the creation of a nonmodifiable extended dynamic package that was processed with the ERROR bind option, an error occurred that prevents the package from being created. The logical unit of work was rolled back (SQLWARN0 = W, SQLWARN6 = W). Examples of these types of errors are invalid section numbers, internal limitations of the system, or a system error (-9XX).

SQLSTATEs:

42969 Statement too long or invalid section number

58027 Earlier system error

System Action: Processing has ended for this logical unit of work and all changes to the database have been rolled back.

User Response: Correct the cause of the problem and try again.

-921 **The system catalog table** *table-name* **has an incorrect column type** *column-type***.**

Explanation: The catalog table in the message has an incorrect column type as shown.

SQLSTATE 58025.

System Action: Processing of the SQL statement has ended.

User Response: Notify your system programmer.

System Programmer Response: This is a system error. Make a record of what went wrong and contact the designated support group for your installation.

-923 **The index was marked invalid.**

Explanation: One of the following may have happened during ROLLBACK, UNDO or REDO processing:

- The database manager cannot obtain a logical index page. The shortage of index pages can result from the ROLLBACK or UNDO of either a DROP INDEX, a REORGANIZE INDEX, a deletion of a row, or an update of a row. It can also result from the REDO of either a CREATE INDEX, an insertion of a row, or an update of a row.
- The database manager cannot obtain a physical index page. The shortage of index pages can result from the ROLLBACK or UNDO of either a DROP INDEX, a REORGANIZE INDEX, a deletion of a row, or an update of a row. It can also result from the REDO of either a CREATE INDEX, an insertion of a row, or an update of a row.
- Not enough virtual storage or not enough room in a temporary DBSPACE exists to recreate an index. The shortage can result from the ROLLBACK or UNDO of either a DROP INDEX or REORGANIZE INDEX, or from the REDO of a CREATE INDEX or REORGANIZE INDEX.

SQLSTATE 51012.

System Action: The database manager completed the ROLLBACK, UNDO, or REDO without updating the index and marked the index invalid. You may now have more than one invalid index. If the database manager performed an implicit ROLLBACK, the actual reason for the ROLLBACK (DBSS Return Code) is in SQLERRD2.

User Response: Issue the SHOW INVALID operator command to find the invalid indexes. You cannot use an invalid index until you have dropped and recreated it. Drop invalid indexes that you own, and inform your system programmer about any other invalid indexes.

System Programmer Response: Contact the owners of the invalid indexes and have them drop and recreate the indexes.

-928 **An error has occurred while connecting with the CMS GLOBALV Command Processor when trying to access GLOBAL variable** *name***.**

Explanation: An error occurred when the application server tried to access the LASTING GLOBALV file. The SQLINIT EXEC and the CMS GLOBALV command write parameters to the LASTING GLOBALV file. Possible causes are:

- the SQLINIT EXEC has not been invoked
- an "SQL/DS" value has been deleted from the LASTING GLOBALV file

- an incorrect parameter was found in the LASTING GLOBALV file.

The group name for all values specified by the database manager is "SQL/DS".

The system tried to access the global variable **name**.

SQLERRD1 contains further information about the error. The possible values are:

0 When **reading** the global variable.

1 When **writing** the global variable.

SQLSTATE 55028.

System Action: Processing of the SQL statement ends.

User Response: Issue the SQLINIT EXEC with the correct parameters, or issue SQLINIT RESET to reset to DB2 Server for VM default values. If the error persists, notify your system programmer.

System Programmer Response: Verify the parameters that were specified in the SQLGLOB EXEC during initialization.

-930 **Sufficient storage is not available to process the statement.**

Explanation: The virtual storage necessary to execute the command is not available now.

SQLSTATE 57011.

System Action: Processing of the SQL statement has ended.

User Response: Resubmit the statement later. If the problem persists, notify your system programmer.

System Programmer Response: One of the following has occurred:

- All available virtual storage has been used.
- An internal database error has occurred.

This error is associated with message ARI0001E. Refer to the explanation of message ARI0001E for more information.

Use a larger virtual machine for the database manager. If the problem persists, make a record of what went wrong and contact the designated support group for your installation. If the problem persists, make a record of what went wrong and contact the designated support group for your installation.

-931 **Failure while freeing dynamically allocated storage.**

Explanation: While attempting to free storage during processing, an error occurred. This is a DB2 Server for VM or VM error.

SQLSTATE 58004.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: This is a DB2 Server for VM or VM error.

Notify your system programmer. After a severe error has occurred, only a CONNECT statement will be accepted.

If you are using CMS work units, you may switch to another work unit to do some work. If the CMS work unit in which the error occurred is reestablished at any time, errors may result and the work unit status will be unknown.

System Programmer Response: Make a record of what went wrong and contact the designated support group for your installation.

-932 **An error occurred when the Resource Adapter attempted to get or to free dynamic storage.**

Explanation: The resource adapter encountered an error in either getting or freeing storage needed for control blocks and work areas. SQLERRM will be set to GET for a failure when trying to obtain storage. SQLERRM will be set to FREE, MBXFREE, CURSFREE, or SQLDAFRE for a failure when trying to free up storage.

SQLSTATEs:

57011 Dynamic storage could not be obtained.

58004 Dynamic storage could not be released.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: If possible, retry with a larger virtual machine size. Otherwise, notify your system programmer.

After a severe error has occurred, only a CONNECT statement will be accepted. If the resource adaptor is initialized, only a CONNECT statement will be accepted. If the resource adaptor is not initialized, any subsequent statement will result in termination of the application.

If you are using CMS work units, you may switch to another work unit to do some work. If the CMS work unit in which the error occurred is reestablished at any time, errors may result and the work unit status will be unknown.

System Programmer Response: SQLERRD1 contains the DMSFREE/DMSFRET or CMS CMSSTOR macro return code. If SQLERRD1 is 1, it indicates that not

enough space was available to satisfy the request for additional storage. SQLERRD2 indicates the amount of space required. If SQLERRD1 is not 1, make a record of what went wrong and contact the designated support group for your installation.

-933 **The communications path to** *dbname* **is disabled. No further access to this path is possible. Request Code =** *request-code*. **PROTOCOL:** *protocol* **FUNCTION:** *function* **REG15: X'***reg15***' CC:** *condition-code* **IPRCODE: X'***iprcode***' IPCODE: X'***ipcode***' IPWHATRC: X'***ipwhatrc***' IPAUDIT: X'***ipaudit***'**

Explanation: The application server detected a communication error. This usually indicates that the communication path to the application server was disconnected. This disconnection may have resulted from:

- The operator terminating the application server
- The operator issuing a FORCE command
- Exceeding the maximum number of connections to the database machine.
- A serious system or sub-system error

Request Code = *request-code* is the communication request code.

If the SQLSTATE is 57027, then the operator has terminated the application server or issued a FORCE command.

If the SQLSTATE is 57030, then the maximum number of connections allowed to the database machine has been exceeded. This limit (MAXCONN) is established in the DB2 Server for VM database machine directory. Try your SQL request later when the system activity is lighter. If this condition occurs frequently, ask your system programmer to increase the value of MAXCONN.

If the SQLSTATE is 40003, then a system error has occurred.

For further information on determining the cause of the error, see "Chapter 22. Debugging Communication Errors" on page 403.

SQLSTATEs:

57027	Operator action.
57030	Maximum number of connections to virtual machine exceeded.
40003	System error.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: Report this problem to your system programmer.

After a severe error has occurred, only a CONNECT statement will be accepted.

System Programmer Response: Take corrective action based on the Explanation and the User Response.

-934 **An error occurred referencing** *object*. **The action being performed was** *action*.

Explanation: An error occurred when the Resource Adapter attempted to perform the specified *action* on the specified *object*.

object is the program or control block on which the action is performed.

action is the action performed.

Note: If the application is linked with SQL/DS Version 3 Release 3 or earlier code, the *action* may not be available.

SQLSTATE 58021.

System Action: Processing has ended. The following fields are set in the SQLCA:
- SQLWARN0 and SQLWARN6 are set to S to signal that a severe error has occurred
- SQLERRD1 is set to the error code returned by the failing action
- SQLERRP contains the name of the module where the error occurred.

User Response: If the error occurs when you are running under VM with virtual machine size larger than 16 megabytes and your application has been link-edited with an old version of the database manager, your action would be one of the following:
- Relink and retry your application.
- Set your machine mode to 370 and retry your application.
- Reduce the amount of virtual storage to less than 16 megabytes and retry your application.

In all other cases, or if the VM error persists, notify your system programmer.

If you are using CMS work units, you may switch to another work unit to do some work. If the CMS work unit in which the error occurred is reestablished at any time, errors may result and the work unit status will be unknown.

System Programmer Response: The problem may be that the resource adapter bootstrap module (ARISRMBT) was not found because the SQLINIT EXEC was not run prior to preprocessing or running the application.

Review your installation procedures to determine if you executed all the steps successfully. At this time, you should have added all necessary packages to your

DB2 Server for VM service and production minidisks. If no steps were left out and all ran successfully, you have a database system error. Make a record of what went wrong and contact the designated support group for your installation.

-935 **The address of an output variable is outside the partition (VSE), or virtual machine boundaries (VM).**

Explanation: An address pointing to application output variables was found to be outside the application virtual machine boundaries.

SQLSTATE 51004.

System Action: The resource adapter returns to the calling program. Some output variables may have been moved to the user area if they contained valid addresses. No moving takes place after the incorrect addresses are found. Processing of the SQL statement has ended. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred. If a severe error has occurred, only a CONNECT statement will be accepted.

In the SQLCA, SQLERRD1 is set to 5, meaning an addressing error has occurred, or SQLERRD1 is set to 4, meaning a protection exception has occurred.

SQLERRD2 is set to the host variable ordinal position number of the address in error, in the host variable list or SQLDA descriptor.

User Response: Notify your system programmer.

If you are using CMS work units, you may switch to another work unit to do some work. If the CMS work unit in which the error occurred is reestablished at any time, errors may result and the work unit status will be unknown.

System Programmer Response: Check the RDIIN structure and verify that it is the same as after preprocessing. If the program looks valid, make a record of what went wrong and contact the designated support group for your installation.

-939 **An error occurred with the CMS MACRO** *name* **while accessing the file** *filename filetype***.**

Explanation: During Resource Adapter initialization, an error occurred while accessing the file *filename filetype* that is required for CCSID support using the DRDA protocol.

SQLSTATE 58030.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: Report this problem to your system programmer.

After a severe error has occurred, only a CONNECT statement will be accepted.

System Programmer Response: To correct the problem, refer to SQLERRD1 for the return code from the CMS MACRO.

-940 **The SQL statement was not processed because the application server resource** *server-name* **was not ready or was not found.**

Explanation: An implicit or explicit connect request failed due to the following possible reasons:

- If you issued a CONNECT statement with the TO *server-name* option, either no DB2 Server for VM application servers were started with the application server resource name you desired, or you specified an incorrect server name.
- Otherwise, it is likely that your application server has not been started with the server name you specified on the SQLINIT exec.
- SNA sessions between the AVS gateway and the target LU may not be established when the user attempts to connect to a remote application server through VTAM.
- Intermediate communications server, like TSAF, or AVS may not be active.
- The COMDIR has been unloaded by the VM system after the application program or the application server abended.

If the SQLSTATE is 57011, then a communications directory entry was found with the specified *server-name*, but the connection to the target application server failed.

If the SQLSTATE is 42705, no communications directory entry was found or the directories were not used. The attempt to connect to the target application server failed.

SQLSTATEs:

57011 **server-name** resource not available.

42705 **server-name** resource not available.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: After a severe error has occurred, only a CONNECT statement will be accepted.

Contact your system administrator to start up the desired application server, communications server, or SNA sessions, and restart your application.

-941 **An error occurred while the database manager was building a system communication message.**

Explanation: An error was detected while building an IUCV or APPC/VM message. The cause of the error is indicated by the return code in SQLERRD1.

The possible return codes are:

12 Not enough virtual storage for the message.

16 Incorrect storage area being freed.

36 CALLTYPE in RDIIN specifies a function which is not supported.

40 The specified RDICTYPE requires an input length and RDIMBLEN is not specified in RDIIN.

44 SQLTYPE in SQLDA specifies a data type which is not supported.

48 RDIMBLEN in RDIIN does not specify a length which is equal to the total input.

52 Free area pointer is not equal to the start of the Mailbox plus our current offset to it.

56 Space used plus space left not equal to 8K for blocked output.

60 Address of blocked PUT Mailbox is less than or equal to zero.

64 No pointer to Mailbox found in RDIIN for blocked PUT.

For error codes 12 and 16 the amount of storage required by the message is placed in SQLERRD2.

For error code 44, the ordinal number of the host variable with the unsupported datatype in the SQLDA is placed in SQLERRD2.

Either the application program overlaid some of the input structure, or this is a system error.

SQLSTATE 58004.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: Notify your system programmer. After a severe error has occurred, only a CONNECT statement will be accepted.

If you are using CMS work units, you may switch to another work unit to do some work. If the CMS work unit in which the error occurred is reestablished at any time, errors may result and the work unit status will be unknown.

System Programmer Response: Verify that the desired database machine has been started and that the application does not mistakenly overlay any of the DB2 Server for VM input structures. If neither of these is the case, then this is a DB2 Server for VM system error. Make a record of what went wrong and contact the designated support group for your installation.

-942 **Because of insufficient storage, the database manager cannot process the SQL statement.**

Explanation: The DB2 Server for VM virtual machine does not have enough storage to build internal control blocks necessary for receiving the SQL statement. SQLERRD1 contains the length of the storage required.

Note: This code will be reported for DB2 Server for VM protocol.

SQLSTATE 57011.

System Action:
- The SQLCODE is returned to the user.
- The logical unit of work is not rolled back.

User Response: Re-enter the SQL statement after the DB2 Server for VM virtual machine is redefined with more virtual storage (or when the virtual machine is less active).

-943 **User attempted to connect to server** *server-name* **on Virtual Machine** *sql-vmid***. Virtual Machine is not accessing this server.**

Explanation: The user virtual machine has on its A-disk a CMS file named ARISRMBT MODULE. This file specifies to the resource adapter the particular application server and the particular DB2 Server for VM virtual machine to be accessed when an EXEC SQL statement is issued. This file is initialized by the SQLINIT EXEC.

SQLCODE -943 is issued when the specifications in the ARISRMBT MODULE file do not match the situation in the system. That is, the database manager in the virtual machine specified is accessing an application server other than the one specified in the ARISRMBT MODULE file.

server-name Application server specified in ARISRMBT MODULE.

sql-vmid Id of the DB2 Server for VM virtual machine specified in ARISRMBT MODULE.

SQLSTATE 57031.

System Action: Processing of the SQL statement has ended. Any further attempts to access the database manager will cause a CMS abend.

User Response: Determine if the server name

identified in the message is the correct application server

- If the server name is not correct, run the SQLINIT EXEC, specifying the application server to be accessed; then rerun the application.
- If the server name is correct, an application server must be started in the virtual machine. Notify the system operator of this need. After that application server has been started, rerun the application.

  Note: If the application server has been started in a virtual machine other than the one specified in the ARISRMBT MODULE file, you must first rerun the SQLINIT EXEC, specifying that virtual machine, and then rerun the application.

-944 **The system dbspace SYS0002 does not exist.**

Explanation: The database manager was attempting to store a package on dbspace SYS0002 during preprocessing; however, SYS0002 does not exist. This dbspace should have been created at DB2 Server for VM installation time during database generation.

SQLSTATE 57026.

System Action: Preprocessing will end.

User Response: Contact your system programmer to find out why dbspace SYS0002 was not created or why it no longer exists. When SYS0002 becomes available, rerun the preprocessor job.

System Programmer Response: A user with DBA authority may have deleted the SYS0002 entry in the SYSTEM.SYSDBSPACES catalog table. Use the INSERT command on the SYSTEM.SYSDBSPACES catalog table with the following values for the associated column names:

```
DBSPACENAME - SYS0002
DBSPACENO - 2
OWNER - PUBLIC
DBSPACETYPE - 1
NTABS - 0
NPAGES - (value specified at database
          generation for the second
          PUBLIC DBSPACE)
NRHEADER - 8
PCTINDEX - 0
FREEPCT - 1
LOCKMODE - P
NACTIVE - -1
POOL - (value specified at database
        generation for the second
        PUBLIC DBSPACE)
```

If this fails or does not correct the problem, you have a system error. A database generation may be necessary. Make a record of what went wrong and contact the designated support group for your installation.

-945 **All dbspaces for packages are full. Acquire another dbspace for additional packages.**

Explanation: Because all dbspaces for packages are full, another package dbspace must be acquired before CREATE VIEW, CREATE PACKAGE, RELOAD PACKAGE, or a preprocessor can run successfully. A dbspace might be full for one of the following reasons:

- All the available packages in the dbspace are used.
- No more space is available in the dbspace.

A FREEPCT column value of zero in a row for a package dbspace in SYSTEM.SYSDBSPACES catalog table indicates the second condition. Issuing the following SELECT indicates (by a count of 255) that all packages in a dbspace are used:

```
SELECT DBSPACENO,COUNT(*)
       FROM SYSTEM.SYSACCESS
WHERE TNAME NOT LIKE '%AVAILABLE'
GROUP BY DBSPACENO
```

SQLSTATE 57025.

System Action: Preprocessing or statement execution ends. If dbspaces became full during the current logical unit of work, the current logical unit of work is rolled back, the entry or entries in the SYSTEM.SYSDBSPACES table for the filled dbspaces is updated to reflect the full condition (the FREEPCT columns are updated), and the entries in the SYSTEM.SYSACCESS table for the filled dbspaces are also updated to reflect the full condition (the TNAME columns are updated).

User Response: Notify your system programmer or database administrator.

System Programmer Response: Acquire a public dbspace named **SYSyyyy** as follows:

```
ACQUIRE PUBLIC DBSPACE NAMED SYSyyyy
(PAGES = xxx, PCTINDEX = 0, PCTFREE = 1)
```

where SYSyyyy is the name of the new package dbspace (yyyy being a value 0003-9999 that has not already been used) and xxx is the number of dbspace pages for the packages in this new dbspace. This size is installation dependent, as determined by the size of the programs at your installation. (The recommended size for the initial dbspace is 1024 pages).

Refer to the *DB2 Server for VM System Administration* manual for exact instructions on acquiring dbspaces for packages.

-946 **There is insufficient space remaining for packages in dbspace number** *dbspace-number.* **Someone with DBA authority may need to acquire another dbspace for packages if all existing package dbspaces are full.**

Explanation: While attempting to insert a row into a package table residing in dbspace number *dbspace-number*, the database manager found that the dbspace was full.

SQLSTATE 57025.

System Action: The current logical unit of work is rolled back. The entry in the SYSTEM.SYSDBSPACES catalog table for the filled dbspace is updated to reflect the full condition. (The FREEPCT column is updated.) The entries in the SYSTEM.SYSACCESS table for the filled dbspace are also updated to reflect the full condition. (The TNAME columns are updated.) Preprocessing will end.

User Response: Notify your system programmer or database administrator.

System Programmer Response: If other package dbspaces are not yet full, retry the command. If all package dbspaces are now full, you need to acquire additional package dbspaces. A dbspace might be full for one of the following reasons:
- All the available packages in the dbspace are used.
- No more space is available in the dbspace.

A FREEPCT column value of zero in a row for a package dbspace entry in the SYSTEM.SYSDBSPACES catalog table indicates the second condition. Issuing the following SELECT indicates (by a count of 255) that all packages in a dbspace are used:

```
SELECT DBSPACENO,COUNT(*)
        FROM SYSTEM.SYSACCESS
WHERE TNAME NOT LIKE '%AVAILABLE'
GROUP BY DBSPACENO
```

Acquire a PUBLIC DBSPACE named **SYSyyyy** as follows:

```
ACQUIRE PUBLIC DBSPACE NAMED SYSyyyy
(PAGES = xxx, PCTINDEX = 0, PCTFREE = 1)
```

where SYSyyyy is the name of the new package DBSPACE (yyyy being a value 0003-9999 that has not already been used) and xxx is the number of dbspace pages for the packages in this new dbspace. This size is installation dependent, as determined by the size of the programs at your installation. (The recommended size for the initial dbspace is 1024 pages.)

Refer to the *DB2 Server for VM System Administration* manual for exact instructions on acquiring dbspaces for packages.

-947 **This environment does not support** *string***.**

Explanation: See each Format, below:

Format 1

If *string* is **DATABASE SWITCHING**, the Explanation, SQLSTATE value, System Action and User Response are as follows:

This environment does not support database switching.

Environment may be:
- SINGLE USER MODE
- DBMODE=N

You tried to connect to a database other than the default one while running in single user mode. Database Switching Support is only available in multiple user mode with DBMODE=L or DBMODE=G.

SQLSTATE 56038.

System Action: The statement is not executed.

User Response: If you were operating in single user mode, rerun the program in multiple user mode.

Format 2

If *string* is **CMS WORK UNITS**, the Explanation, SQLSTATE value, System Action and User Response are as follows:

Explanation:

You tried to switch to a new CMS work unit while running in single user mode. Support is only available in multiple user mode.

SQLSTATE 56038.

System Action: Processing of this SQL statement ends.

User Response: If you were operating in single user mode, rerun the program in multiple user mode.

Format 3

This format is not used in VM.

Format 4

If *string* is **RESULT SET PROCESSING**, the Explanation, SQLSTATE value, System Action and User Response are as follows:

Explanation: You tried to process result sets over private protocol. The following is a list of statements that are not supported when private protocol is in use :
- ASSOCIATE LOCATORS
- ALLOCATE CURSOR
- DESCRIBE PROCEDURE
- DESCRIBE CURSOR

SQLSTATE 56038.

System Action: Processing of this SQL statement ends.

User Response: Preprocess and execute the program against the database application server and do not use private protocol.

-948 **The current logical unit of work was rolled back, and the connection to the application server was severed because of operator action.**

Explanation: The DB2 Server for VM virtual machine operator entered the FORCE command to cause ROLLBACK of the current logical unit of work and to sever the connection to DB2 Server for VM The user ID, password, and database name are all reset to their default values.

SQLSTATE 57027.

System Action: When you enter the next DB2 Server for VM command, you are automatically connected as the default user ID. The default user ID is the VM user ID

User Response: You may continue to use DB2 Server for VM. Any explicit CONNECTs you have issued have been lost. DB2 Server for VM treats your DB2 Server for VM requests as originating with a user ID that is the same as that of your virtual machine ID. You must issue another explicit CONNECT request, using the same user ID, to re-establish your original connection to DB2 Server for VM.

-949 **A system error has occurred. Reason Code =** *reason-code***.**

Explanation: The database manager has encountered an Operating System error. SQLERRD1 contains the return code from the Operating System if one exists. The possible Reason Codes are:

01 An error occurred while the Resource Adapter was attempting to EXTRACT the current work unit id.

SQLSTATE 58024.

System Action: The statement is not executed.

User Response: Notify your system programmer.

System Programmer Response: Make a record of what went wrong. Ensure that both the Reason Code and the return code are included in this information. Contact the designated support group for your installation.

-963 **You cannot run this version of DB2 Server for VM in your present VM environment.**

Explanation: To execute, Version 5 Release 1 of the DB2 Server for VM product requires VM/ESA. However, if you are running a DB2 Server for VM system, the application program may have been link-edited with an incompatible resource adapter stub, ARIRVSTC.

SQLSTATE Spaces.

System Action: Processing terminates. The resource adapter returns to the application with SQLCA fields SQLWARN0 and SQLWARN6 both set to S, indicating a serious system error.

User Response: If the error occurs when you are running SQL/DS Version 3 under VM/SP 5, contact your system administrator to set up a VM/ESA environment and rerun your application.

If the error occurs when you are running a SQL/DS Version 2 Release 2, relink-edit your application with the SQL/DS Version 2 Release 2 version of ARIRVSTC, and rerun the application.

-965 **FORMAT 1**
Stored procedure *proc-name*
terminated abnormally.
FORMAT 2
Stored procedure *proc-name*
terminated abnormally for AUTHID
authid.

Explanation: DB2 received an SQL CALL statement for a stored procedure. The stored procedure terminated abnormally while attempting to execute the statement.

proc-name
The name of the stored procedure that terminated abnormally.

authid The authorization ID associated with the stored procedure.

SQLSTATE: 51021

System Action: The statement cannot be executed. If the procedure has exceeded the maximum abend count, which is determined by the startup parameter PROCMXAB, then the procedure *proc-name* is stopped and further requests to call it are rejected.

User Response: Correct the errors in the procedure, and then issue START PROC command to activate the stored procedure.

-993 **The file** *filename filetype* **cannot be found.**

Explanation: During Resource Adapter initialization, the file *filename filetype* that is required for CCSID support using the DRDA protocol cannot be found on the system production minidisk.

SQLSTATE 57044.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: Report this problem to your system programmer. After a severe error has occurred, only a CONNECT statement will be accepted.

System Programmer Response: Create the file *filename filetype*. Refer to the *DB2 Server for VM System Administration* manual for the steps to follow.

-994 **The row with charname = *charname* does not exist in the file *file*.**

Explanation: During Resource Adapter initialization, the character translation table for *charname* cannot be loaded because the required row does not exist in the file ARISSCR MACRO.

SQLSTATE 57045.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: Report this problem to your system programmer. After a severe error has occurred, only a CONNECT statement will be accepted.

System Programmer Response: Create the file ARISSCR MACRO to include the required translation table. Refer to the *DB2 Server for VM System Administration* manual for the steps to follow.

-8001 **Product key validation failed.**

Explanation: A valid product license key is not found or is invalid. A valid key is required to run DB2 Server for VSE. SQLERRD1 contains the return code from the validation. The possible values are:

8 The key is not found or the key that was found is invalid.

20 The key validation routine is not loaded.

SQLSTATE 42968.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: Report this error to your system programmer.

System Programmer Response: Obtain and install a valid key for the product. A valid key can be obtained by contacting your IBM representative. Instructions for installing the key are found in the *DB2 Server for VSE Program Directory*.

-15000 **The *object-type object-name* must have a status of *status*, and not be in use, before it can be altered or dropped.**

Explanation: A stored procedure must have been stopped with the REJECT option, giving a status of STOP-REJ, and not be running in a stored procedure server, before an ALTER or DROP of that procedure can be processed. In addition, no agents can be running the procedure. A stored procedure server must have been stopped with IMPL=NO before an ALTER or DROP of that stored procedure server can be processed.

Note: If the CREATE PROCEDURE or CREATE PSERVER statement is not yet committed, the PROCEDURE or PSERVER cannot be stopped since the entry is not put in the cache until the unit of work is committed. However, in this case, as ALTER or DROP is allowed if it is issued in the same unit of work as the CREATE statement.

SQLSTATE 55044.

System Action: Processing of the SQL statement ends.

User Response: Issue the appropriate STOP operator command. If the ALTER or DROP PROCEDURE is rejected because an agent is running the procedure, the SHOW PROC command can be used to monitor activity and determine when the agent(s) have finished running the procedure. Once no agents are running the procedure, issue the ALTER or DROP statement again.

-30000 **Execution failed because of a DRDA protocol error that does not affect the successful execution of subsequent commands or SQL statements. Reason Code = *reason-code*(*sub-code*).**

Explanation: The *reason-code* value is the 2-byte hexadecimal code point for the DDM reply message that reported the error and is one of the following:

- X'1254' - CMDCHKRM
- X'220A' - DSCINVRM
- X'220E' - DTAMCHRM
- X'1245' - PRCCNVRM
- X'2202' - QRYNOPRM
- X'220F' - QRYPOPRM
- X'2207' - RDBACCRM
- X'2204' - RDBNACRM
- X'124C' - SYNTAXRM

When the error is reported at the application requester, the reason-code is the 2-byte hexadecimal code point of the reply message that most closely defines the error condition.

The *sub-code* is a 2-byte hexadecimal value represented as a hexadecimal string constant. The first byte of the *sub-code* is defined as:

- X'01' which indicates that the error was discovered at the application requester.
- X'02' which indicates that the error was discovered at the application server.

The second byte of the *sub-code* should be X'00', unless the reply message has an error code defined. The error code value should be placed in the second byte. The reply messages with error codes defined are listed below:

DSCINVRM
The *sub-code* value is a 1-byte hexadecimal value that defines the type of description error. Refer to the DDM term DSCERRCD in the *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4* manual for the 1-byte hexadecimal values.

PRCCNVRM
The *sub-code* value is a 1-byte hexadecimal value that defines the type of conversation protocol error. Refer to the DDM term PRCCNVCD in the *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4* manual for the 1-byte hexadecimal values.

SYNTAXRM
The *sub-code* value is a 1-byte hexadecimal value that defines the type of syntax error. Refer to the DDM term SYNERRCD in the *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4* manual for the 1-byte hexadecimal values.

Note: This code will be reported for DRDA protocol when an insufficient storage condition is encountered.

SQLSTATE 58008.

System Action: The SQL statement is not processed.

User Response: Discontinue further processing. Notify your system programmer.

-30020 **Execution failed because of a DRDA protocol error that caused deallocation of the conversation. Reason Code =** *reason-code*(*sub-code*)

Explanation: The *reason-code* value is the 2-byte hexadecimal code point for the DDM reply message that reported the error and is one of the following:

- X'1232' - AGNPRMRM
- X'1254' - CMDCHKRM
- X'220A' - DSCINVRM
- X'220E' - DTAMCHRM
- X'1218' - MGRDEPRM
- X'1245' - PRCCNVRM
- X'2202' - QRYNOPRM
- X'220F' - QRYPOPRM
- X'2207' - RDBACCRM
- X'2204' - RDBNACRM
- X'124C' - SYNTAXRM

When the error is reported at the application requester, the reason-code is the 2-byte hexadecimal code point of the reply message that most closely defines the error condition.

The *sub-code* is a 2-byte hexadecimal value represented as a hexadecimal string constant. The first byte of the *sub-code* is defined as:

- X'01' which indicates that the error was discovered at the application requester.
- X'02' which indicates that the error was discovered at the application server.

The second byte of the *sub-code* should be X'00', unless the reply message has an error code defined. The error code value should be placed in the second byte. The reply messages with error codes defined are listed below:

DSCINVRM
The *sub-code* value is a 1-byte hexadecimal value that defines the type of descriptor error. Refer to the DDM term DSCERRCD in the *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4* manual for the 1-byte hexadecimal values.

MGRDEPRM
The *sub-code* value is a 1-byte hexadecimal value that defines the manager dependency that was not met. Refer to the DDM term DEPERRCD in the *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4* manual for the 1-byte hexadecimal values. No FFDC dump.

PRCCNVRM
The *sub-code* value is a 1-byte hexadecimal value that defines the type of conversation protocol error. Refer to the DDM term PRCCNVCD in the *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4* manual for the 1-byte hexadecimal values.

SYNTAXRM
The *sub-code* value is a 1-byte hexadecimal value that defines the type of syntax error. Refer to the DDM term SYNERRCD in the *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4* manual for the 1-byte hexadecimal values.

SQLSTATE 58009.

System Action: Processing of the SQL statement ends

and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: After a severe error has occurred, only a CONNECT statement will be accepted.

System Programmer Response: This is a DRDA protocol error. Make a record of what went wrong and contact the designated support group.

-30021 **Execution failed because of a DRDA protocol error that will affect the successful execution of subsequent DDM commands or SQL statements. Manager** *manager* **at level** *level* **is not supported.**

Explanation: The *manager* value is the 2-byte hexadecimal code point for the DDM manager class identified as not supported in the EXCSATRD that reported the error. Refer to the DDM term MGRLVL in the *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4* manual for the 2-byte hexadecimal values.

The manager *level* is the decimal value for the manager class identified as not supported in the EXCSATRD that reported the error. Refer to the DDM term MGRLVL in the *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4* manual for more information.

SQLSTATE 58010.

System Action: The attempt to connect to the database is not successful. The resource adapter returns to the application with SQLCA fields SQLWARN0 and SQLWARN6 both set to S, indicating a serious system error.

User Response: Do not continue processing. Notify your system programmer.

-30040 **Execution failed because of unavailable resources that do not affect the successful execution of subsequent commands and SQL statements: Application server** *server-name***, type of resource** *resource-type***, reason** *reason-code***(**s*ub-code***).**

Explanation: The application server has problems obtaining or freeing resources that are needed to process the current request before or after the request is executed by the target relational database manager.

server-name identifies the target application server.

resource-type indicates the type of resource involved, such as storage; *reason-code* may or may not contain product-specific information, depending on the product ID involved. The *sub-code* is a 2-byte hexadecimal value represented as a hexadecimal string constant. The first byte of the *sub-code* is defined as:

- X'01' indicates that the error was discovered at the application requester.
- X'02' indicates that the error was discovered at the application server.

The second byte of the *sub-code* is X'00.

SQLSTATE 57012.

System Action: The SQL statement is not processed.

User Response: Resubmit the statement later. If the problem persists, notify your system programmer.

-30041 **Execution failed because of unavailable resources that affect the successful execution of subsequent commands or SQL statements: Application server** *server-name***, type of resource** *resource-type***, reason** *reason-code* **(**s*ub-code***).**

Explanation: The application server has problems obtaining or freeing resources that are needed to process the current request before or after the request is executed by the target relational database manager. The *server-name* identifies the target application server.

The *resource-type* indicates the type of resource involved, such as storage; *reason-code* may or may not contain product specific information, depending on the product-id involved. The *sub-code* is a 2-byte hexadecimal value represented as a hexadecimal string constant. The first byte of *sub-code* is defined as:

- X'01' indicates that the error was discovered at the application requester.
- X'02' indicates that the error was discovered at the application server.

The second byte of the *sub-code* is X'00.

SQLSTATE 57013.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: After a severe error has occurred, only a CONNECT statement will be accepted.

System Programmer Response: Use a larger virtual storage size for the application server. If the problem persists, make a record of what went wrong and contact the designated support group.

-30050 **The** *ddm-command-codepoint* **command is invalid while the bind process is in progress.**

Explanation: An attempt was made to execute a specific DDM command that is invalid while a bind process is in progress. BNDSQLSTT, RDBCMM, RDBRLLBCK, and ENDBND are the only legal DDM commands that you can issue while a bind process is in

progress. The *ddm-command-codepoint* command is a two byte hexadecimal codepoint.

SQLSTATE 58011.

System Action: The SQL statement is not processed. The resource adapter returns to the application with SQLCA fields SQLWARN0 and SQLWARN6 both set to S, indicating a serious system error.

User Response: Do not continue processing. Notify your system programmer of this DRDA protocol error.

-30051 **The bind process with specified package name and consistency token is not active.**

Explanation: An attempt was made to execute a DDM BNDSQLSTT or ENDBND command for a bind process that was not active.

SQLSTATE 58012.

System Action: The preprocessing of the program ends. No package is created. The resource adapter returns to the application with SQLCA fields SQLWARN0 and SQLWARN6 both set to S, indicating a serious system error.

User Response: Do not continue further processing. Notify your system programmer of this DRDA protocol error.

-30052 **Program preparation assumptions are incorrect.**

Explanation: An SQL statement that is incompletely understood by the preprocessor is sent to the target application server with the following assumptions:

- All host variables are assumed to be input variables.
- The statement is assigned a unique section number.
- The section is executed through the DDM EXCSQLSTT command.

At least one of these assumptions is incorrect according to the target application server.

SQLSTATE 42932.

System Action: The preprocessing of the program has ended. No package is created.

User Response: Do not continue processing. Notify your system programmer and database administrator of this DRDA protocol error.

-30053 **Owner** *pkgownid* **authorization failure.**

Explanation: An authorization error occurred associated with *pkgownid* when preprocessing a program. The following are some possible scenarios where this error may occur:

- The person who preprocesses the program does not have the authority to specify the value *pkgownid* for the OWNER parameter.
- When the target application server is a DB2 Server for VM server, the value of the OWNER parameter (of SQLPREP EXEC or CREATE PACKAGE) is not the same as that of:
  - the COLLECTION-ID portion for the PREPNAME parameter
  - the USERID parameter when the COLLECTION-ID portion for the PREPNAME parameter is not specified
  - the connected user's id as it is known to the target application server when the USERID parameter is not specified and the COLLECTION-ID portion for the PREPNAME parameter is not specified

SQLSTATE 42506.

System Action: The SQL statement is not processed, or the preprocessing of the program has ended with no package created.

-30060 **Application Server** *server-name* **authorization failure.**

Explanation: The user is not authorized to access the application server *server-name*.

SQLSTATE 08004.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: After a severe error has occurred, only a CONNECT statement will be accepted.

-30061 **Application server** *server-name* **is not found.**

Explanation: The user attempted to connect to an application server *server-name* which cannot be found.

SQLSTATE 08004.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: After a severe error has occurred, only a CONNECT statement will be accepted.

Contact your system administrator to start up the desired application server and restart your application.

-30070 *ddm-command-codepoint* **DDM command is not supported.**

Explanation: The target does not support a particular DDM command. The error causes the command processing to end, but does not affect the processing of subsequent DDM commands and SQL statements issued by the application program. The *ddm-command-codepoint* command is a two byte hexadecimal codepoint.

SQLSTATE 58014.

System Action: The SQL statement is not processed.

User Response: Do not continue processing. Notify your system programmer and database administrator of this DRDA protocol error.

-30071 **There is no support for the DDM object** *ddm-object-type***.**

Explanation: The application server does not support a particular DDM object. The error causes current DDM command processing to end, but does not affect the processing of subsequent DDM commands and SQL statements issued by the application program. The *ddm-command-type* command is a two byte hexadecimal codepoint.

SQLSTATE 58015.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: After a severe error has occurred, only a CONNECT statement will be accepted.

-30072 **There is no support for the DDM command parameter** *ddm-parameter***.**

Explanation: The target does not support a particular DDM command parameter. The error causes command processing to end, but does not affect the processing of subsequent DRDA commands and SQL statements issued by the application program. The *ddm-command-type* command is a two byte hexadecimal codepoint.

SQLSTATE 58016.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

System Programmer Response: Make a record of what went wrong and contact the designated support group.

-30073 **Value not supported for** *ddm-parameter* **parameter.**

Explanation: The target does not support a particular DDM command parameter value. The error causes termination of processing, but does not affect the processing of subsequent DDM commands and SQL statements issued by the application program. The *ddm-command-type* command is a two byte hexadecimal codepoint.

SQLSTATE 58017.

System Action: The SQL statement is not processed.

User Response: Do not continue processing. Notify your system programmer and database administrator of this DRDA protocol error.

-30074 **There is no support for the reply message with a codepoint of** *codepoint***(***svrcod***)**

Explanation: The application server has returned a reply message that is not known to the application requester. DDM architecture defines the valid *codepoint*s and *svrcod* values for reply messages.

SQLSTATE 58018.

System Action: The SQL statement has produced unpredictable results.

User Response: Do not continue processing. Notify your system programmer and database administrator of this DRDA protocol error.

-30080 **Communication error occurred.**

PROTOCOL: *protocol*
FUNCTION: *function*
REG15: X'*reg15***'**
CC: *condition-code*
IPRCODE: X'*iprcode***'**
IPCODE: X'*ipcode***'**
IPWHATRC: X'*ipwhatrc***'**
IPAUDIT: X'*ipaudit***'**

Explanation: This SQLCODE reports communications errors that are visible to an application requester or application server.

One possible reason is that a FORCE command was issued by the DB2 Server for VM operator. See "Chapter 22. Debugging Communication Errors" on page 403 for further information on determining the cause of the error.

SQLSTATE 08001.

System Action: Processing of the SQL statement ends and the logical unit of work (if any) is rolled back. SQLWARN0 and SQLWARN6 of the SQLCA are set to S to signal that a severe error has occurred.

User Response: After a severe error has occurred, only a CONNECT statement will be accepted.

System Programmer Response: Take corrective action based on the **Explanation** and the **User Response**.

-30090 **The remote operation is invalid for this application execution environment.**

Explanation: You are trying to update a read-only application server, or to dynamically prepare or execute a COMMIT or ROLLBACK statement against a read-only application server.

SQLSTATE 25000.

System Action: The SQL statement is not processed.

User Response: Issue other non-update SQL statements or notify your database administrator.

Chapter 20. SQLSTATEs

If you have the Remote unit of work feature, the SQLSTATE value can provide application programs with common codes for common error conditions that can occur in any of the DB2 family of database systems including DB2 Server for VM.

An SQLSTATE field is returned in the SQLCA for all warnings and errors that can be returned from executing SQL statements. The SQLCA contains both an SQLCODE and a corresponding SQLSTATE value, if applicable. If, however, you are running an SQL/DS Version 2 Release 2 or earlier database that is connected to a DB2 Server for VM Version 7 Release 1 user machine, the DB2 Server for VM system has no control of the SQLSTATE value returned to you.

The application programs, when SQLSTATEs are fully implemented, can test for classes of errors that are common to all products, or for specific errors that are unique to a product.

SQLSTATE values are comprised of a two-character class code value, followed by a three-character subclass code value. Class code values represent classes of successful (class codes 00 and 01) and unsuccessful (all other class codes) execution conditions.

The SQLSTATE class codes defined by IBM SQL and ISO/ANSI are:

00 Unqualified Successful Completion

01 Warning
- **01H** External Function Warning

02 No Data

07 Dynamic SQL Error

08 Connection Exception

09 Triggered Action Exception

0A Feature Not Supported

0F Invalid Token

21 Cardinality Violation

22 Data Exception

23 Constraint Violation

24 Invalid Cursor State

25 Invalid Transaction State

26 Invalid SQL Statement Identifier

27 Triggered Data Change Violation

28 Invalid Authorization Specification

2B Dependent Privilege Descriptors Still Exist

2C Invalid Character Set Name

2D Invalid Transaction Termination

2E Invalid Connection Name

33 Invalid SQL Descriptor Name

34 Invalid Cursor Name

35 Invalid Condition Number

37 Syntax Error (Discontinued)

38 External Function Exception

39 External Function Call Exception

3C Ambiguous Cursor Name

3D Invalid Catalog Name

3F Invalid Collection (Schema) Name

40 Transaction Rollback (Serialization Failure)

42 Syntax Error or Access Rule Violation
- **425** Access Violation
- **426** Syntax Error
- **427** Duplicate or Undefined Name
- **428** Invalid Operand or Inconsistent Specification
- **429** Miscellaneous SQL or Product Restriction

44 WITH CHECK OPTION Violation

51 Invalid Application State

52 Duplicate or Undefined Name (Discontinued)

53 Invalid Operand or Inconsistent Specification (DB2 and SQL/DS only)

54 SQL or Product Limit Exceeded

55 Object Not in Prerequisite State

56 Miscellaneous SQL or Product Error

57 Resource Not Available or Operator Intervention

© Copyright IBM Corp. 1987, 2000

579	Request Suspended - Intervention Required
58	System Error
HY	Call Level Interface
HZ	Remote Database Access
S1	ODBC Interface

Programmers who want to use SQLSTATE as the basis of their application's return codes may define SQLCODE numbers above 32767, SQLSTATE classes beginning with the characters '7' through '9' or 'I' through 'Z' and subclasses for any class beginning with the characters 'I' through 'Z'. These value ranges will not be used by IBM SQL or ISO/ANSI.

The SQLSTATE values have been standardized in SQL92 the *(ISO IS 9075:1992, Database Language SQL* manual). If the SQLSTATE values of all databases comply with SQL92, customers who are writing applications which are either portable across SQL implementations or have to connect to more than one SQL implementation can thus rely on a consistent SQLSTATE being returned to report the same diagnostic condition.

IBM standardized diagnostic reporting across its various databases by complying with the SQLSTATE mechanism proposed in the ISO/ANSI SQL2 the *(DIS 9075:199x(E) Database Language SQL2)* draft a few years ago. However, the SQLSTATE values in the final published version of SQL92 deviate in many ways from the scheme that SQL/DS was planning to follow a few years ago. As a result, SQL/DS is implementing those changes in Version 3 Release 5 to comply with SQL92.

Here is a summary of the changes:

01501 changed to 01004
01502 changed to 01003

22509 changed to 42806
22515 changed to 23515
22517 changed to 22021
22518 changed to 22021
22520 changed to 23520
23501 changed to 44000
26503 changed to 07005
26504 changed to 07003
26506 changed to 07007

37501 changed to 42601
37502 changed to 42602
37503 changed to 42603
37504 changed to 42604
37505 changed to 42605
37506 changed to 42606
37507 changed to 42607
37509 changed to 42609
37510 changed to 42610
37511 changed to 42611
37512 changed to 42612
37514 changed to 42614
37515 changed to 42615
37517 changed to 42617
37518 changed to 42618

40000 changed to 40001
42507 changed to 08004

51001 changed to 07002
51011 changed to 0A001
51018 changed to 08003
52001 changed to 42701
52002 changed to 42702
52003 changed to 42703
52004 changed to 42704
52005 changed to 42705
52006 changed to 42706
52007 changed to 42707
52008 changed to 34000
52009 changed to 42709
52010 changed to 42710
52011 changed to 42711
52012 changed to 42712
52013 changed to 3C000
52017 changed to 08004
52019 changed to 42719
53002 changed to 42802
53003 changed to 42803
53005 changed to 42805
53007 changed to 42807
53008 changed to 42808
53009 changed to 42809
53010 changed to 42810
53011 changed to 42811
53013 changed to 42813
53015 changed to 42815
53016 changed to 42816
53018 changed to 42818
53019 changed to 42819
53020 changed to 42820
53021 changed to 42821
53023 changed to 42823
53024 changed to 42824
53025 changed to 42825
53026 changed to 42826

53027 changed to 42827
53028 changed to 42828
53029 changed to 42829
53031 changed to 42831
53032 changed to 42832
53034 changed to 42834
53042 changed to 42842
53052 changed to 42852
53053 changed to 42853
53056 changed to 28000
53059 changed to 42859
53062 changed to 42862
53067 changed to 42867
54003 changed to 42622
55008 changed to 42888
55013 changed to 42889
56001 changed to 42901
56002 changed to 42902
56003 changed to 42903
56005 changed to 42905
56006 changed to 42906
56007 changed to 42907
56008 changed to 42908
56009 changed to 42909
56011 changed to 42911
56012 changed to 42912
56013 changed to 42913
56014 changed to 42914
56015 changed to 42915
56017 changed to 42917
56026 changed to 25000
56032 changed to 42932
56037 changed to 42937
56039 changed to 42939
56043 changed to 42943
56061 changed to 42961
56069 changed to 42969
58019 changed to 08001
58022 changed to 40003

The SQLSTATEs and their corresponding SQLCODEs are listed below. In some cases, the SQLSTATE occurs when a particular situation caused the SQLCODE, and these situations are also listed. For more information about the corresponding SQLCODEs, a page number is provided with each reference.

SQLSTATE
Meaning and SQLCODE

00000 0 on page 277 for no warnings

0A001 This SQLSTATE was formerly 51011.

The CONNECT statement is invalid, because the process is not in the connectable state.

-752 on page 349

0F001 The locator value does not currently represent any value.

-423 on page 321

01003 This SQLSTATE was formerly 01502.

Null values were eliminated from the argument of a column function.

0 on page 277 for SQLWARN2=W

01004 This SQLSTATE was formerly 01501.

The value of a string column was truncated when assigned to a host variable.

0 on page 277 for SQLWARN1=W

01501 This SQLSTATE has been changed to 01004.

01502 This SQLSTATE has been changed to 01003.

01503 The number or result columns is larger than the number of host variables provided.

0 on page 277 for SQLWARN3=W

01504 The UPDATE or DELETE statement does not include a WHERE clause.

0 on page 277 for SQLWARN4=W

01506 An adjustment was made to a DATE or TIMESTAMP value to correct an invalid date resulting from an arithmetic operation.

0 on page 277 for SQLWARN7=W

01507 One or more non-zero digits were eliminated from the fractional part of a number used as the operand of a multiply or divide operation.

0 on page 277 for SQLWARN7=Z

01508 The statement was disqualified for blocking for reasons other than storage.

0 on page 277 for SQLWARN8=W

01509 Blocking was cancelled for a cursor, because there is insufficient storage in the user virtual machine.

0 on page 277 for SQLWARN9=W

01510 Blocking was canceled for a cursor, because a blocking factor of at least two rows could not be maintained.

0 on page 277 for SQLWARNA=W

01511 Performance may not be optimum because of the number of predicates specified in the WHERE clause.

0 on page 277 for SQLWARN5=W

01512 The REVOKE operation has no effect on CONNECT privileges.

+150 on page 279

01513 A subsequent commit operation will revoke all EXECUTE privileges on the package except for that of the owner.

+200 on page 280

01515 The null value has been assigned to a host variable, because the non-null value of the column is not within the range of the host variable.

+304 on page 282

01516 An inapplicable WITH GRANT OPTION has been ignored.

+558 on page 286

01517 A character that could not be translated was replaced with a substitute character.

0 on page 277 for SQLWARN8=Z

01519 The null value has been assigned to a host variable, because an arithmetic exception occurred in computing the value of the column.

+802 on page 287

01520 The null value has been assigned to a host variable, because the characters cannot be translated.

+331 on page 282

01525 The number of INSERT values is not the same as the number of columns.

+117 on page 278

01532 An undefined object name was detected.

+204 on page 281

01533 An undefined column name was detected.

+202 on page 280

+205 on page 281

+206 on page 281

01536 During remote bind where existence checking is deferred, the server name specified does not match the current server.

+114 on page 277

01539 Connection is successful, but only SBCS characters should be used.

+863 on page 288

01541 Operator command processing has completed successfully.

+825 on page 288

01542 Authorization ID does not have the privilege to perform the operation as specified.

+552 on page 286

01543 A duplicate constraint has been ignored.

+541 on page 285

01545 An unqualified column name has been interpreted as a correlated reference.

+12 on page 277

01548 Authorization ID does not have the privilege to perform the specified operation on the identified object.

+551 on page 286

01555 Mixed data is invalid and has been truncated according to SBCS rules.

0 on page 277 for SQLWARN1=Z

01570 The bind process detected a character string in an INSERT or UPDATE statement that is too large for the target column.

+404 on page 282

01571 The bind process detected a numeric value that is out of range.

+414 on page 284

01572 The bind process detected an invalid datetime format; that is, an invalid string representation or an invalid value.

+180 on page 280

01573 The bind process detected a null insert or update value is null for a column that cannot contain null values.

+407 on page 283

01574 The bind process detected an INSERT,

UPDATE, or DELETE that is not permitted on this object.

+149 on page 279

01575 The bind process detected a non-updateable column in an INSERT or UPDATE statement.

+151 on page 279

01576 The bind process detected a CREATE INDEX statement for a view.

+154 on page 279

01577 The bind process detected a CREATE VIEW statement that includes an operator or operand that is not valid for views.

+154 on page 279

01578 The bind process detected operands of an operator that are not compatible.

+401 on page 282

01579 The bind process detected a numeric constant that is either too long or has a value that is not within the range of its data type.

+405 on page 283

01580 The bind process detected an update or insert value that is not compatible with the column.

+408 on page 283

01581 The bind process detected incompatible operands of a UNION operation.

+415 on page 284

01582 The bind process detected a string that is too long.

+134 on page 278

+416 on page 284

01583 The bind process detected a decimal divide operation that is invalid because the result would have a negative scale.

+419 on page 285

01584 The bind process detected an insert or update value of a long string column that is neither a host variable or NULL.

+135 on page 278

01585 The bind process detected a table that cannot be accessed because it is inactive.

+668 on page 287

01609 The procedure generated more than the maximum allowed number of result sets. Only the first integer result sets have been returned to the caller.

+464 on page 285

01610 One or more ad hoc results sets were returned from the procedure.

+466 on page 285

01614 There are fewer locations than the number of result sets.

+494 on page 285

02000 One of the following exceptions occurred:

- The result of the SELECT INTO statement or the subselect of the INSERT statement was an empty table.
- The number of rows identified in the searched UPDATE or DELETE statement was zero.
- The position of the cursor referenced in the FETCH statement was after the last row of the result table.

+100 on page 277

07001 This SQLSTATE was formerly 53017.

The number of host variables is not the same as the number of parameter markers.

-313 on page 311

07002 This SQLSTATE was formerly 51001.

An invalid call parameter list or control block was detected.

-804 on page 352

07003 This SQLSTATE was formerly 26504.

The statement identified in the EXECUTE is a select-statement, or it is not in a prepared state.

-160 on page 298 for using EXECUTE

-504 on page 325 when statement not prepared

-516 on page 327 for EXECUTE

-826 on page 355 for reason code 4 and EXECUTE

07004 This SQLSTATE was formerly 53017.

The statement contains parameter markers but the USING clause is not specified.

-313 on page 311

07005 This SQLSTATE was formerly 26503.

The statement-name of the cursor identifies a prepared statement that cannot be associated with a cursor.

-512 on page 327

-826 on page 355 for reason code 3.

07006 This SQLSTATE was formerly 22507.

An input host variable cannot be used because of its data type.

-301 on page 308

07007 This SQLSTATE was formerly 26506.

INTO DESCRIPTOR clause must be specified on an Extended EXECUTE against a statement that was prepared via an Extended PREPARE - Single Row.

-811 on page 353

08001 This SQLSTATE was formerly 58019.

Communication error occurred.

-30080 on page 374

08003 This SQLSTATE was formerly 51018.

An error has left the application in an unconnected state.

-900 on page 359

08004 This SQLSTATE was formerly 42507 or 52017.

The application server rejected establishment of the connection.

-30060 on page 373

-30061 on page 373

2D528 A COMMIT WORK statement cannot be dynamically prepared or executed.

-515 on page 327

2D529 A ROLLBACK WORK statement cannot be dynamically prepared or executed.

-515 on page 327

21000 The result of a SELECT INTO is a table of more than one row, or the result of the subquery of a basic predicate is more than one value.

-810 on page 353

21501 A multiple-row INSERT into a self-referencing table is invalid.

21502 A multiple-row UPDATE of a Primary Key is invalid.

-534 on page 330

21504 A multiple-row DELETE from a self-referencing table with a delete rule of RESTRICT or SET NULL is invalid.

22001 Character data, right truncation occurred; for example, an update or insert value is a string that is too long for the column.

-302 on page 309 when character or graphic string is too long

-304 on page 310 when date or time value is too large for host variable

-404 on page 319.

22002 A null value or the absence of an indicator parameter was detected; for example, the null value cannot be assigned to a host variable, because no indicator variable is specified.

-305 on page 310

22003 A numeric value is out of range.

-302 on page 309 for numeric overflow or underflow

-304 on page 310 for numeric overflow or underflow

-413 on page 320

-425 on page 321

-801 on page 350 for arithmetic overflow and underflow

-802 on page 351 for arithmetic overflow and underflow.

22007 An invalid datetime format was detected; that is, an invalid string representation or an invalid value was specified.

-180 on page 304

-181 on page 304.

22008 Datetime field overflow occurred; for example, an arithmetic operation on a date or timestamp has a result that is not within the valid range of dates, or a datetime value cannot be assigned to a host variable because it is too small.

-183 on page 305

22011 A substring error occurred; for example, an argument of SUBSTR is out of range.

-171 on page 303 when an argument of SUBSTR is out of range

22012 Division by zero is invalid.

-801 on page 350 for division by zero

-802 on page 351 for division by zero.

22019 The LIKE predicate has an invalid escape character.

-131 on page 295

22021 This SQLSTATE was formerly 22517 or 22518.

Character is not in the coded character set.

-330 on page 317

-331 on page 317

22024 This SQLSTATE was formerly 22510.

A C NUL-terminated input host variable with length greater than 1, did not contain a NUL.

-302 on page 309

22025 The LIKE predicate string pattern contains an invalid occurrence of an escape character.

-131 on page 295

22501 The length control field of a variable length string is negative or greater than the maximum.

-311 on page 311

-821 on page 354.

22502 Miscellaneous data exceptions occurred; for example, a floating-point NAN (not a number) or invalid data in a packed decimal field was detected.

-424 on page 321

-801 on page 350 for general errors

-802 on page 351 for general errors.

22504 Mixed data value is improperly formed.

-191 on page 305

22507 This SQLSTATE has been changed to 07006.

22509 This SQLSTATE has been changed to 42806.

22510 This SQLSTATE has been changed to 22024.

22512 A host variable in a predicate is invalid, because its indicator variable is negative.

-309 on page 311

22515 This SQLSTATE has been changed to 23515.

22517 This SQLSTATE has been changed to 22021.

22518 This SQLSTATE has been changed to 22021.

22519 The Primary or Foreign Key cannot be activated.

-667 on page 340

22520 This SQLSTATE has been changed to 23520.

22521 The Foreign Key cannot be defined, because the Primary Key of the parent table is inactive.

-670 on page 341 for reason code 1

22522 CCSID value is not valid at all, not valid for the data type or subtype, or not valid for the encoding scheme.

-189 on page 305

22524 A string conversion resulted in truncation.

-334 on page 318

23501 This SQLSTATE has been changed to 44000.

23502 An insert or update value is null, but the column cannot contain null values.

-407 on page 319

23503 The insert or update value of a foreign key is invalid.

-530 on page 330

23504 The operation is prevented by the RESTRICT update or delete rule.

-532 on page 330

23505 A violation of the constraint imposed by a unique index occurred.

-803 on page 352

23507 A violation of a constraint imposed by a field procedure occurred.

-681 on page 342

23515 This SQLSTATE was formerly 22515.

The unique index could not be created, because the table contains duplicate values of the specified key.

-603 on page 334

23520 This SQLSTATE was formerly 22520.

The Foreign Key cannot be defined, because all of its values are not equal to the primary key of the parent table.

-670 on page 341 for reason code 2

24501 The identified cursor is not open.

-160 on page 298 for using FETCH

-501 on page 325

-507 on page 326.

24502 The cursor identified in an OPEN statement is already open.

-502 on page 325

24503 The cursor identified in the PUT statement is a select cursor, or the cursor identified in the FETCH statement is an insert cursor.

-160 on page 298 for using PUT

24504 The cursor identified in the UPDATE or DELETE statement is not positioned on a row.

-508 on page 326

24505 COMMIT is invalid, because blocking is in effect and an insert cursor is open.

-519 on page 328

24506 The statement identified in the PREPARE is the select-statement of an open cursor.

-519 on page 328

24506 The statement identified in the PREPARE is the select-statement of an open cursor.

24516 A cursor has already been assigned to a result set.

-499 on page 324

25000 This SQLSTATE was formerly 56026.

Operation is invalid for the application execution enviro nment.

-30090 on page 374

26501 The statement identified does not exist.

-516 on page 327 for DESCRIBE or dynamic cursor operation

-826 on page 355 for reason code 1 and 2 DECLARE CURSOR, DESCRIBE, DROP STATEMENT, and PREPARE (Filling empty section)

26503 This SQLSTATE has been changed to 07005.

26504 This SQLSTATE has been changed to 07003.

26505 An extended EXECUTE, DECLARE CURSOR, or DESCRIBE has been issued against an empty section.

-504 on page 325 when trying to execute a NULL section

26506 This SQLSTATE has been changed to 07007.

26507 An extended EXECUTE with an OUTPUT DESCRIPTOR has been issued against a section that is not a Single Row SELECT.

-826 on page 355 for reason code 6

26508 The statement identified in an extended PREPARE Single Row is not a select-statement.

-826 on page 355 for reason code 5

28000 This SQLSTATE was formerly 53056.

Authorization-name or password on CONNECT is missing or invalid.

-564 on page 333

3C000 This SQLSTATE was formerly 52013.

A duplicate cursor name was detected.

-517 on page 327

34000 This SQLSTATE was formerly 52008.

An undefined cursor name was detected.

-504 on page 325 when cursor name not declared

37501 This SQLSTATE has been changed to 42601.

37502 This SQLSTATE has been changed to 42602.

37503 This SQLSTATE has been changed to 42603.

37504 This SQLSTATE has been changed to 42604.

37505 This SQLSTATE has been changed to 42605.

37506 This SQLSTATE has been changed to 42606.

37507 This SQLSTATE has been changed to 42607.

37509 This SQLSTATE has been changed to 42609.

37510 This SQLSTATE has been changed to 42610.

37511 This SQLSTATE has been changed to 42611.

37512 This SQLSTATE has been changed to 42612.

37514 This SQLSTATE has been changed to 42614.

37515 This SQLSTATE has been changed to 42615.

37517 This SQLSTATE has been changed to 42617.

37518 This SQLSTATE has been changed to 42618.

38003 The statement is not allowed in a function or procedure.

-751 on page 348

39002 A null value is not allowed for an IN or INOUT argument when using the SIMPLE CALL calling convention.

-470 on page 323

39501 An output argument value returned from a function or a procedure was too long.

-450 on page 323

40000 This SQLSTATE has been changed to 40001.

40001 This SQLSTATE was formerly 40000.

Deadlock or timeout with automatic rollback occurred.

-911 on page 361

40003 This SQLSTATE was formerly 58022.

A system error has caused the unit of work to be rolled back.

-903 on page 360

-933 on page for a 364 for a system error.

40503 PRIVATE dbspace is in use by another process.

-710 on page 346

42501 Authorization ID does not have the privilege to perform the specified operation on the identified object.

-551 on page 331

-566 on page 333.

42502 Authorization ID does not have the privilege to perform the operation as specified.

-552 on page 331

-554 on page 332

-555 on page 332

-556 on page 332

-606 on page 335

-610 on page 335

-708 on page 346.

42504 A specified privilege cannot be revoked from a specified authorization-name.

-556 on page 332 when revokee does not have privilege

-558 on page 332.

42505 Connection authorization failure occurred.

-560 on page 332

-561 on page 333

-815 on page 354.

42506 Owner authorization failure occurred.

-30053 on page 373

42507 This SQLSTATE has been changed to 08004.

42601 This SQLSTATE was formerly 37501.

A character, token, or clause is invalid or missing.

-104 on page 289 for general errors

-105 on page 290

-106 on page 290 for general errors

-109 on page 291

-110 on page 291

-115 on page 291

-128 on page 294

-130 on page 295

-159 on page 297

-160 on page 298 for using ALL, REFERENCE, STRUCTURE, COST, or PLAN

-162 on page 300 when the name is missing

-524 on page 329

-611 on page 336.

42602 This SQLSTATE was formerly 37502.

An invalid character in a name has been detected.

-162 on page 300 when the name is incorrect

-165 on page 301

-210 on page 308.

42603 This SQLSTATE was formerly 37503.

An unterminated string constant has been detected.

-106 on page 290 for an unterminated string constant

42604 This SQLSTATE was formerly 37504.

An invalid numeric or string constant has been detected.

-103 on page 289

-164 on page 301

-166 on page 301

-170 on page 302.

42605 This SQLSTATE was formerly 37505.

The number of arguments specified for a scalar function is invalid.

-172 on page 304

42606 This SQLSTATE was formerly 37506.

An invalid hexadecimal constant has been detected.

-167 on page 302

42607 This SQLSTATE was formerly 37507.

An invalid operand of a column function has been detected.

-112 on page 291

-409 on page 319.

42609 This SQLSTATE was formerly 37509.

All operands of an operator or predicate are parameter markers.

-417 on page 320

42610 This SQLSTATE was formerly 37510.

An invalid use of parameter markers has been detected.

-184 on page 305

-418 on page 321.

42611 This SQLSTATE was formerly 37511.

An invalid column definition has been detected.

-604 on page 334

42612 This SQLSTATE was formerly 37512.

The statement string is an SQL statement that is not acceptable in the context in which it is presented.

-836 on page 358

42614 This SQLSTATE was formerly 37514.

Invalid duplication of keyword has been detected.

-630 on page 339

42615 This SQLSTATE was formerly 37515.

Invalid alternative has been detected.

-168 on page 302

-705 on page 346

42617 This SQLSTATE was formerly 37517.

The statement string is blank or empty.

-104 on page 289 when statement string is empty or blank

42618 This SQLSTATE was formerly 37518.

Host variable is not allowed.

-163 on page 301

-312 on page 311.

42622 This SQLSTATE was formerly 54003.

A name or label is too long.

-107 on page 290

-108 on page 291

-116 on page 292

-162 on page 300 when the name is too long.

42701 This SQLSTATE was formerly 52001.

A duplicate column name in INSERT or UPDATE statement was detected.

-121 on page 293

42702 This SQLSTATE was formerly 52002.

A column reference is ambiguous because of duplicate names.

-203 on page 306

-209 on page 307

42703 This SQLSTATE was formerly 52003.

An undefined column name was detected.

-201 on page 306

-202 on page 306

-205 on page 307

-666 on page 340.

42704 This SQLSTATE was formerly 52004.

An undefined object or constraint name was detected.

-204 on page 307 for general errors

-206 on page 307

-608 on page 335

-805 on page 352 for a GRANT, REVOKE or DROP statement.

-649 on page 340 for a missing EXPLAIN table.

42705 This SQLSTATE was formerly 52005.

An undefined server name was detected.

-841 on page 358

-940 on page 365

42706 This SQLSTATE was formerly 52006.

Column names in ORDER BY are invalid, because all columns of the result table are unnamed.

-207 on page 307

42707 This SQLSTATE was formerly 52007.

Column name in ORDER BY does not identify a column of the result table.

-208 on page 307

42709 This SQLSTATE was formerly 52009.

A duplicate column name in a PRIMARY or FOREIGN KEY clause was detected.

-537 on page 330

42710 This SQLSTATE was formerly 52010.

A duplicate object name was detected.

-601 on page 334

-618 on page 337

-644 on page 339.

42711 This SQLSTATE was formerly 52011.

A duplicate column name in an object definition was detected.

-612 on page 336

42712 This SQLSTATE was formerly 52012.

A duplicate table designator in a FROM clause was detected.

-211 on page 308

42719 This SQLSTATE was formerly 52019.

The cursor referenced in the extended OPEN, FETCH, PUT, or CLOSE has not been declared in the current unit of work.

-518 on page 327

42724 Unable to access an external program used for a user-defined function or a procedure.

-444 on page 322

42802 This SQLSTATE was formerly 53002.

The number of insert values is not the same as the number of columns.

-117 on page 292

42803 This SQLSTATE was formerly 53003.

A column reference in the SELECT or HAVING clause is invalid, because it is not a grouping column or a column reference in the GROUP BY clause is invalid.

-119 on page 292

42805 This SQLSTATE was formerly 53005.

An integer in the ORDER BY clause does not identify a column of the result table.

-125 on page 293

42806 This SQLSTATE was formerly 22509.

A value cannot be assigned to a host variable, because the data types are not compatible.

-303 on page 310

42807 This SQLSTATE was formerly 53007.

The INSERT, UPDATE, or DELETE is not permitted on this object.

-149 on page 297

42808 This SQLSTATE was formerly 53008.

A column identified in the INSERT or UPDATE statement is not updateable.

-151 on page 297

42809 This SQLSTATE was formerly 53009.

The identified object is not the type of object to which the statement applies.

-154 on page 297 for using CREATE INDEX on a view.

-204 on page 307 for DROP VIEW table or DROP TABLE view

-657 on page 340

-813 on page 354.

42810 This SQLSTATE was formerly 53010.

A view is identified in a FOREIGN KEY clause.

-157 on page 297 for general errors

42811 This SQLSTATE was formerly 53011.

The number of columns specified is not the same as the number of columns in the SELECT clause.

-158 on page 297

42813 This SQLSTATE was formerly 53013.

WITH CHECK OPTION cannot be used for the specified view.

-140 on page 296

42815 This SQLSTATE was formerly 53015.

The data type, length, value, or CCSID of an operand is invalid.

-171 on page 303 for general errors

-713 on page 347.

42816 This SQLSTATE was formerly 53016.

An invalid use of a datetime value or duration in an expression has been detected.

-182 on page 304

-187 on page 305.

42818 This SQLSTATE was formerly 53018.

The operands of an operator are not compatible.

-401 on page 318

42819 This SQLSTATE was formerly 53019.

An operand of an arithmetic operation is not a number.

-402 on page 318

42820 This SQLSTATE was formerly 53020.

A numeric constant is too long, or it has a value that is not within the range of its data type.

-405 on page 319

42821 This SQLSTATE was formerly 53021.

An update or insert value is not compatible with the column.

-408 on page 319

42823 This SQLSTATE was formerly 53023.

The SELECT clause of a subquery specifies multiple columns.

-412 on page 320

42824 This SQLSTATE was formerly 53024.

An operand of LIKE is not a string, or the first operand not a column.

-132 on page 295

-414 on page 320.

42825 This SQLSTATE was formerly 53025.

The operands of UNION do not have compatible columns.

-415 on page 320

42826 This SQLSTATE was formerly 53026.

The operands of UNION do not have the same number of columns.

-422 on page 321 for different number of items in select lists

42827 This SQLSTATE was formerly 53027.

The table identified in the UPDATE or DELETE is not the same table designated by the cursor.

-509 on page 326

42828 This SQLSTATE was formerly 53028.

The table designated by the cursor of the UPDATE or DELETE statement cannot be modified, or the cursor is read-only.

-510 on page 326

-520 on page 328.

42829 This SQLSTATE was formerly 53029.

FOR UPDATE OF is invalid, because the table designated by the cursor cannot be modified.

-126 on page 293

-511 on page 327.

42830 This SQLSTATE was formerly 53030.

The Foreign Key does not conform to the description of the Primary Key.

-538 on page 330

42831 This SQLSTATE was formerly 53031.

A column of a Primary Key cannot allow null values.

-542 on page 331

42832 This SQLSTATE was formerly 53032.

The operation is not allowed on system objects.

-609 on page 335

-709 on page 346

-814 on page 354

-823 on page 354.

42834 This SQLSTATE was formerly 53034.

SET NULL cannot be specified, because the Foreign Key does not allow null values.

-629 on page 338

42842 This SQLSTATE was formerly 53042.

A column definition is invalid, because a specified option is inconsistent with the column description.

-683 on page 343

42852 This SQLSTATE was formerly 53052.

Privileges specified in GRANT or REVOKE are invalid or inconsistent.

-557 on page 332

42853 This SQLSTATE was formerly 53053.

Both alternatives of an option were specified, or the same option was specified more than once.

-161 on page 300

42859 This SQLSTATE was formerly 53059.

An extended PREPARE or DROP STATEMENT cannot be used to change a non-modifiable package.

-830 on page 356

42862 This SQLSTATE was formerly 53062.

An extended dynamic statement cannot be executed against a non-extended dynamic package.

-827 on page 356

42867 This SQLSTATE was formerly 53067.

Conflicting options have been specified.

-160 on page 298 for the inconsistent use of the NEW, CHECK, or ERROR clauses

42878 An invalid function or procedure name was used with the EXTERNAL keyword.

-449 on page 322

42884 No function or procedure was found with the specified name and compatible arguments.

-440 on page 322

42886 The IN, OUT, or INOUT parameter attributes do not match.

-469 on page 323

42888 This SQLSTATE was formerly 55008.

The nominated parent table does not have a Primary Key.

-539 on page 331

42889 This SQLSTATE was formerly 55013.

The table already has a primary key.

-656 on page 340

42893 The object or constraint cannot be dropped, because other objects are dependent on it.

-478 on page 324

42895 For static SQL, an input host variable cannot be used, because its data type is not compatible with the parameter of a procedure or user-defined function.

-302 on page 309

42901 This SQLSTATE was formerly 56001.

A column function does not include a column name.

-111 on page 291

42902 This SQLSTATE was formerly 56002.

The object of the INSERT, UPDATE, or DELETE is also identified in a FROM clause.

-118 on page 292

42903 This SQLSTATE was formerly 56003.

A WHERE clause or SET clause includes an invalid reference such as a column function.

-120 on page 293

42905 This SQLSTATE was formerly 56005.

DISTINCT is specified more than once in a subselect.

-127 on page 294

42906 This SQLSTATE was formerly 56006.

A column function in a subquery of a HAVING clause includes an expression that applies an operator to a correlated reference.

-133 on page 295

42907 This SQLSTATE was formerly 56007.

The string is too long.

-134 on page 295

-416 on page 320.

42908 This SQLSTATE was formerly 56008.

The statement does not include a required column list.

-153 on page 297

42909 This SQLSTATE was formerly 56009.

The CREATE VIEW includes an operator or operand that is not valid for views.

-154 on page 297 for general errors

42911 This SQLSTATE was formerly 56011.

A decimal divide operation is invalid, because the result would have a negative scale.

-419 on page 321

42912 This SQLSTATE was formerly 56012.

A column cannot be updated, because it is not identified in the UPDATE clause of the select-statement of the cursor.

-503 on page 325

42913 This SQLSTATE was formerly 56013.

An invalid UPDATE or DELETE WHERE CURRENT OF has been detected.

-535 on page 330

42914 This SQLSTATE was formerly 56014.

The DELETE is invalid, because a table referenced in a subquery can be affected by the operation.

-536 on page 330

42915 This SQLSTATE was formerly 56015.

One or more invalid referential constraints have been detected.

-632 on page 339

-648 on page 339

42917 This SQLSTATE was formerly 56017.

The object cannot be explicitly dropped.

-669 on page 341

42932 This SQLSTATE was formerly 56032.

Program preparation assumptions are incorrect.

-30052 on page 373

42937 This SQLSTATE was formerly 56037.

Parameter must not have a subtype of mixed.

-192 on page 306

42939 This SQLSTATE was formerly 56039.

The object cannot be created, because the specified qualifier (or an DB2 Server for VM DBSPACE name that starts with SYS) is reserved for system use.

-550 on page 331

-707 on page 346.

42943 This SQLSTATE was formerly 56043.

The database manager has not committed package package-name because it is a modifiable package and it contains no statements.

-759 on page 349

42961 This SQLSTATE was formerly 56061.

The server name specified does not match the current server.

-114 on page 291

42969 This SQLSTATE was formerly 56069.

The package was not created and the unit of work was rolled back because of either an invalid section number or internal limitations.

-917 on page 362

44000 This SQLSTATE was formerly 23501.

The INSERT or UPDATE is not allowed, because a resulting row does not satisfy the view definition.

-141 on page 296

51001 This SQLSTATE has been changed to 07002.

51002 The package corresponding to an SQL statement execution request was not found.

-805 on page 352 for any statement other than GRANT, REVOKE or DROP

51003 Consistency tokens do not match.

-818 on page 354

51004 An address in the SQLDA is invalid.

-935 on page 365

-563 on page

51010 The programmable interface for operator commands is not valid when within a unit of work.

-825 on page 355

51011 This SQLSTATE has been changed to 0A001.

51012 Index has been marked invalid.

-923 on page 362

51013 An attempt has been made to use an index that has been marked as invalid.

-621 on page 337

51015 An attempt was made to execute a section that was found to be in error at bind time.

-525 on page 329

51016 A package or view cannot be rebound, because the character set under which it was originally bound is different than the character set under which the database manager is running.

-833 on page 357

51018 This SQLSTATE has been changed to 08003.

51021 SQL statements cannot be executed until the application process executes a rollback operation.

-965 on page 369

51030 The procedure referenced in a DESCRIBE PROCEDURE statement or an ALLOCATE CURSOR statement has not yet been called within the application process.

-480 on page 324

-482 on page 324

52001 This SQLSTATE has been changed to 42701.

52002 This SQLSTATE has been changed to 42702.

52003 This SQLSTATE has been changed to 42703.

52004 This SQLSTATE has been changed to 42704.

52005 This SQLSTATE has been changed to 42705.

52006 This SQLSTATE has been changed to 42706.

52007 This SQLSTATE has been changed to 42707.

52008 This SQLSTATE has been changed to 34000.

52009 This SQLSTATE has been changed to 42709.

52010 This SQLSTATE has been changed to 42710.

52011 This SQLSTATE has been changed to 42711.

52012 This SQLSTATE has been changed to 42712.

52013 This SQLSTATE has been changed to 3C000.

52017 This SQLSTATE has been changed to 08004.

52019 This SQLSTATE has been changed to 42719.

53002 This SQLSTATE has been changed to 42802.

53003 This SQLSTATE has been changed to 42803.

53005 This SQLSTATE has been changed to 42805.

53007 This SQLSTATE has been changed to 42807.

53008 This SQLSTATE has been changed to 42808.

53009 This SQLSTATE has been changed to 42809.

53010 This SQLSTATE has been changed to 42810.

53011 This SQLSTATE has been changed to 42811.

53013 This SQLSTATE has been changed to 42813.

53015 This SQLSTATE has been changed to 42815.

53016 This SQLSTATE has been changed to 42816.

53017 This SQLSTATE has been changed to 07001 and 07004.

53018 This SQLSTATE has been changed to 42818.

53019 This SQLSTATE has been changed to 42819.

53020 This SQLSTATE has been changed to 42820.

53021 This SQLSTATE has been changed to 42821.

53023 This SQLSTATE has been changed to 42823.

53024 This SQLSTATE has been changed to 42824.

53025 This SQLSTATE has been changed to 42825.

53026 This SQLSTATE has been changed to 42826.

53027 This SQLSTATE has been changed to 42827.

53028 This SQLSTATE has been changed to 42828.

53029 This SQLSTATE has been changed to 42829.

53030 This SQLSTATE has been changed to 42830.

53031 This SQLSTATE has been changed to 42831.

53032 This SQLSTATE has been changed to 42832.

53034 This SQLSTATE has been changed to 42834.

53042 This SQLSTATE has been changed to 42842.

53043 Columns with different field procedures cannot be compared.

-686 on page 343

53044 The columns have a field procedure, but the field types are not compatible.

-687 on page 343

53052 This SQLSTATE has been changed to 42852.

53053 This SQLSTATE has been changed to 42853.

53056 This SQLSTATE has been changed to 28000.

53059 This SQLSTATE has been changed to 42859.

53060 Public dbspaces must be acquired from a recoverable storage pool.

-706 on page 346

53062 This SQLSTATE has been changed to 42862.

53067 This SQLSTATE has been changed to 42867.

54001 The statement is too long or too complex.

-101 on page 288

54002 A string constant is too long.

-102 on page 289

54003 This SQLSTATE has been changed to 42622.

54004 The statement has too many table names or too many items in a SELECT or INSERT list.

-129 on page 294 for too many table names or columns

54005 The sort key is too long, or it has too many columns.

-124 on page 293

54006 The result of concatenation is too long.

-137 on page 296

54008 The key is too long, or it has too many columns.

-614 on page 336

-617 on page 336.

54011 Too many columns were specified for a table or view.

-602 on page 334

54012 The FIELDPROC literal list is too long.

-684 on page 343

54013 The statement has too many host variables.

-129 on page 294 for too many variables or too many characters in variable names

54014 Too many cursors are open.

-522 on page 329

54015 Section was not created as a result of executing the null form of an extended dynamic PREPARE or preprocessing a PREPARE statement.

-523 on page 329

54016 Too many tables were detected in a DBSPACE.

-607 on page 335

54017 The maximum number of active packages for a unit of work has been exceeded.

-807 on page 353

54018 The row is too long.

-809 on page 353

54019 The maximum number of late descriptions has been exceeded, probably because too many different CCSIDs were used.

-129 on page 294

54020 Too many indexes were detected for a table.

-605 on page 335

55002 The explanation table is not defined properly.

-619 on page 337

55008 This SQLSTATE has been changed to 42888.

55013 This SQLSTATE has been changed to 42889.

55021 Change of data type or length of host variable is invalid, because blocking is in effect.

-521 on page 328

55023 An error occurred calling a procedure.

-471 on page 324

55028 Parameter in the LASTING GLOBALV file is either missing or incorrect.

-928 on page 362

55044 The PROCEDURE must have a status of STOP-REJ, or the PSERVER must be stopped with IMPL=N, before it can be altered or dropped.

-15000 on page 370

56001 This SQLSTATE has been changed to 42901.

56002 This SQLSTATE has been changed to 42902.

56003 This SQLSTATE has been changed to 42903.

56004 The statement failed, because the Invalid Entities table is full.

-698 on page 344

56005 This SQLSTATE has been changed to 42905.

56006 This SQLSTATE has been changed to 42906.

56007 This SQLSTATE has been changed to 42907.

56008 This SQLSTATE has been changed to 42908.

56009 This SQLSTATE has been changed to 42909.

56011 This SQLSTATE has been changed to 42911.

56012 This SQLSTATE has been changed to 42912.

56013 This SQLSTATE has been changed to 42913.

56014 This SQLSTATE has been changed to 42914.

56015 This SQLSTATE has been changed to 42915.

56017 This SQLSTATE has been changed to 42917.

56026 This SQLSTATE has been changed to 25000.

56031 The clause or scalar function is invalid, because mixed and graphic data are not supported on this system.

-640 on page 339

56032 This SQLSTATE has been changed to 42932.

56033 The insert or update value of a long string column must be a host variable or NULL.

-135 on page 296

56034 ALLUSERS can only be used in GRANT CONNECT without a password.

-562 on page 333

56035 Referential constraints cannot cross DBSPACEs resident in different types of storage pools.

-714 on page 347

56037 This SQLSTATE has been changed to 42937.

56038 Requested feature is not supported in this environment.

-947 on page 368

56039 This SQLSTATE has been changed to 42939.

56041 An Extended PREPARE can only be executed using the DRDA protocol if it has an input SQLDA.

-844 on page 359

56042 Only one package can be created or modified in a unit of work, and, while that package is being created or modified, all statements in that unit of work must be issued against that package. If the package is non-modifiable, only Extended PREPARE statements can be issued.

-565 on page 333

-756 on page 349

-757 on page 349

-758 on page 349

-832 on page 357.

56043 This SQLSTATE has been changed to 42943.

56044 An attempt was made to execute a section that has been marked as invalid in a modifiable package that is undergoing modification.

-760 on page 350

56046 CREATE PACKAGE with the REPLACE option cannot be issued against a modifiable package.

-829 on page 356

56047 PREPARE Adding Empty Section was not preceded by a CREATE PACKAGE with the NOMODIFY option.

-834 on page 357

56048 Three part package name is not supported.

-136 on page 296

56049 An unexpected error occurred when attempting to rebind a view with a new version of the database manager. The view must be dropped and recreated.

-835 on page 358

56058 This SQLSTATE has been changed to 2D528 and 2D529.

56061 This SQLSTATE has been changed to 42961.

56069 This SQLSTATE has been changed to 42969.

56076 An DB2 Server for VM application requester that uses the DRDA-only protocol cannot be connected to an application server that only supports the SQLDS protocol.

-860 on page 359

56079 Neither protocol option AUTO nor DRDA can be specified because the DRDA facility has not been installed for the application requester.

-861 on page 359

57008 The date or time local format exit has not been installed.

-185 on page 305

57010 A field procedure could not be loaded.

-682 on page 342

57011 Virtual storage or database resource is not available.

-674 on page 341

-701 on page 344

-702 on page 345

-703 on page 345

-704 on page 345

-711 on page 346

-712 on page 347

-930 on page 363

-932 on page 363 when obtaining dynamic storage

-940 on page 365 when server-name resource is not available.

-942 on page 366.

57012 A non-database resource is not available. This will not affect successful execution of subsequent statements.

-30040 on page 372

57013 A non-database resource is not available. This will affect the successful execution of subsequent statements.

-30041 on page 372

57014 Processing was canceled as requested.

-914 on page 361.

57016 Table cannot be accessed, because it is inactive.

-668 on page 341

57017 Character conversion is not defined.

-332 on page 318

57022 Table could not be created, because the authorization ID of the statement does not own any suitable dbspaces.

-620 on page 337

57024 No appropriate message repository can be accessed.

-720 on page 347

-721 on page 348.

57025 There is not enough room in the dbspaces allocated to hold packages.

-945 on page 367

-946 on page 368.

57026 The system dbspace SYS002 does not exist. This dbspace is used to store packages.

-944 on page 367

57027 Connection to the application server has been severed by the operator.

-916 on page 362

-933 on page 364 for operator action

-948 on page 369 for operator action

57028 Unit of work has been rolled back due to an excessive number of system-wide lock requests.

-912 on page 361

57029 Unit of work has been rolled back due to an excessive number of lock requests by the unit of work.

-915 on page 362

57030 Connection to the application server would exceed the installation-defined limit.

-933 on page when maximum number of connections exceeded

57031 Connection to the application server is not possible, because the DB2 Server for VM virtual machine does not have access to that application server.

-943 on page 366

57033 Deadlock or timeout occurred without automatic rollback.

-913 on page 361

57037 The ACQUIRE DBSPACE command failed, because all storage pools for available dbspaces are full.

-699 on page 344

57038 No space is available in the storage pool.

-700 on page 344

57040 The communications directory was either not found, or it has the wrong file type.

-841 on page 358

57044 Resource adapter cannot find file.

-993 on page 369

57045 The Resource Adapter cannot find an entry for the character set in the ARISSCR MACRO file.

-994 on page 370

58002 An exit has returned an error or invalid data.

-685 on page 343

-688 on page 343

-816 on page 354

58003 An invalid section number was detected.

-144 on page 296

58004 A system error (that does not necessarily preclude the successful execution of subsequent SQL statements) occurred.

-753 on page 349

-824 on page 355

-901 on page 360

-931 on page 363

-932 on page 363 when freeing dynamic storage

-941 on page 366.

58005 A system error (that prevents the successful execution of subsequent SQL statements) occurred.

-902 on page 360

-754 on page 349

58009 Execution failed due to a distribution protocol error that caused deallocation of the conversation.

-30020 on page 371

58010 Execution failed due to a distribution protocol error that will affect the successful execution of subsequent DDM commands or SQL statements.

-30021 on page 372

58011 The DDM command is invalid while the bind process in progress.

-30050 on page 372

58012 The bind process with the specified package name and consistency token is not active.

-30051 on page 373

58014 The DDM command is not supported.

-30070 on page 373

58015 The DDM object is not supported.

-30071 on page 374

58016 The DDM parameter is not supported.

-30072 on page 374

58017 The DDM parameter value is not supported.

-30073 on page 374

58018 DDM reply message is not supported.

-30074 on page 374

58019 This SQLSTATE has been changed to 08001.

58021 A system error occurred while loading a program or control block.

-934 on page 364

58022 This SQLSTATE has been changed to 40003.

58023 System error has caused the current program to be canceled.

-906 on page 360

58024 An error occurred in the underlying operating system.

-949 on page 369

58025 A column in a catalog table has the wrong data type.

-921 on page 362

58027 The package was not created and the unit of work was rolled back due to an earlier system error.

-917 on page 362

58029 An internal error has occurred while attempting to log user data.

-754 on page 349

58030 An I/O error occurred.

-939 on page 365

Chapter 21. DB2 Server for VM Reason Codes

The DB2 Server for VM reason codes are printed in message ARI0042I. The reason codes are the VM return codes that were returned to the database manager. Message ARI0042I is usually accompanied by other database messages; for example: ARI0026E, ARI0027E, ARI0039E, ARI0040E, ARI0041E, etc. The return codes fall into four categories:

1. VM System Services (such as NUCXLOAD or free storage).
2. VM Abnormal Termination Codes.
3. VM Inter-User Communication Vehicle (IUCV) return codes.
4. APPC/VM Return Codes.

VM System Services Return Codes

The VM System Services return codes are found in the following manuals:

- *VM/ESA: CMS Command Reference*
- *VM/ESA: System Messages and Codes*

The accompanying database messages should assist in determining the system service that was being invoked.

VM Abnormal Termination Codes

The return codes from VM Abnormal Termination are found in *VM/ESA: System Messages and Codes* under the discussions of the CP Abend Codes and CMS Abend Codes.

The database manager does not get control when a CP abnormal termination condition occurs. The database manager does get control when a Conversational Monitor System (CMS) abnormal termination occurs. Because many of the CMS Abend Codes apply to use of VSAM and OS functions under CMS, they are not listed here.

For certain abnormal termination conditions, the database manager issues the CMS DMSABN abnormal termination macro with a unique DB2 Server for VM abnormal termination code. These codes are listed below:

Code Value Dec (Hex)	VM Abnormal Termination Code Meaning
242 (F2)	Unable to free the resource adapter code when in the saved segment area and single user mode. (CP PURGESYS diagnose failed).
243 (F3)	Unable to free the relational data system code when in the saved segment area. (CP PURGESYS DIAGNOSE failed).
244 (F4)	Unable to free the database storage subsystem/data system control code when in the saved segment area via the CMSSTACK macro.
245 (F5)	FILEDEF failed for ARILARC or ARIARC. During initialization of the DB2 Server for VM system, it was discovered that the block size for DDNAME ARILARC or ARIARC was invalid. An attempt to re-issue a FILEDEF for this DDNAME has failed.
246 (F6)	Unable to free the ARICMOD control block from the free storage storage area. (CMS NUCXDROP command failed.)
247 (F7)	Unable to free the resource adapter code from the free storage storage area. (CMS NUCXDROP command failed.)
248 (F8)	Unable to free the user program when in the saved segment area and single user mode. (CP PURGESYS DIAGNOSE failed.)
249 (F9)	Unable to free the user program when in the free storage storage area. (CMS NUCXDROP command failed.)
250 (FA)	The DB2 Server for VM system detected a recursive entryination condition.
251 (FB)	DB2 Server for VM external interrupt handler encountered a system error condition. A message will identify the system error point.

252 (FC)	May be caused by not having enough virtual storage available. A dump is not performed, but the following diagnostic messages are issued:
	`ARI0059E Program cancelled due to insufficient virtual storage`
	`ARI0059E Module name = xxxxxxxx`
	`ARI0059E Storage requested = n bytes`
	`ARI0042I DB2 Server for VM Reason Code is n`
	`ARI0043I DB2 Server for VM Return Code is 512`

© Copyright IBM Corp. 1987, 2000

DB2 Server for VM Reason Codes

Code Value Dec (Hex)	VM Abnormal Termination Code Meaning

253 (FD)	This condition results in either a dumpof the registers at the time of the abend or a dump of the registers and virtual storage. Messages will accompany the dump. For instance, if a request for virtual storage could not be honored because none was available, the system performs a CP DUMP of the virtual machine after issuing these messages:
	`ARI0059E Program cancelled due to insufficient virtual storage`
	`ARI0059E No diagnostic possible - dump function invoked`

254 (FE)	DB2 Server for VM DBSS/DSC/RDS bootstrap module found an existing copy of either DBSS/DSC, or RDS code, or user exit router when trying to load the code into the free storage storage area with the CMS NUCXLOAD command. (Storage was not reset properly.)
255 (FF)	ISQL in not supported in VM/XA mode.

VM APPC/VM Communication Codes

The following is a list of VM APPC/VM Communication Codes. They are issued with message ARI0042I, when Remote Relational Access is used.

Code Value Dec (Hex)	Code Meaning
101 (65)	An I/O error occurred while CP was reading the CP directory. *IDENT was checking for authorization to identify or revoke the resource.
102 (66)	The CONNECT parameter list has an invalid function code in FCODE of IPUSER; it is not equal to 1 for IDENTIFY or equal to 2 for REVOKE. The database manager produces a dump.
103 (67)	The CONNECT parameter list has an invalid parameter PRMDATA.
104 (68)	The directory does not contain an IUCV *IDENT entry for the database being used, or there is no IUCV *IDENT entry for RESANY. The virtual machine is not authorized to connect to *IDENT for the specified resource.
105 (69)	An attempt has been made to identify the database as a global resource (DBMODE=G) when it should be identified as a local resource. The virtual machine is not authorized to identify the resource as a global resource.
106 (6A)	The virtual machine is not authorized to revoke the specified resource. The database manager produces a dump.
107 (6B)	The virtual machine is not authorized to revoke the specified resource globally. The database manager produces a dump.
108 (6C)	CP cannot identify the resource because the system resource table currently contains the maximum of 500 entries.
109 (6D)	A virtual machine already owns the resource being identified. The virtual machine trying to identify the resource could be the same one that already owns the resource.
110 (6E)	A virtual machine revoked the resource. The virtual machine that owns the resource may have revoked the resource.
111 (6F)	The resource to be revoked does not exist. The database manager produces a dump.
112 (70)	The resource is pending identification by a virtual machine and cannot be identified or revoked.
113 (71)	The resource is pending a revoke by a virtual machine and cannot be identified or revoked. An authorized virtual machine is revoking the resource, or the resource-owner virtual machine is severing its path to *IDENT.
114 (72)	The connect parameter list has a resource name specified as all blanks. The database manager produces a dump.
115 (73)	The directory does not contain any IUCV *IDENT entries.

VM IUCV Communication Codes

The VM IUCV communication codes are usually displayed as returned to the DB2 Server for VM system. In some cases, they are translated by the database manager. These codes are displayed by the database manager and the Resource Adapter virtual machines.

The message ARI0504I is issued by the Resource Adapter and displays the request code, return code, and reason code. If the reason code = 06 or 07 with a return code of 04, accompanying messages will be issued by the DB2 Server for VM virtual machine. The accompanying messages must be considered to determine the condition that existed and the action to be taken.

The following is a list of the reason codes, return codes, and their meanings as contained in message ARI0504I. The list is presented in four parts:

1. Codes common to the DB2 Server for VM and Resource Adapter virtual machines.
2. Codes of the DB2 Server for VM virtual machine.
3. Codes of the Resource Adapter virtual machine.
4. Codes arising during Block I/O operations on the database.

Codes Common to the DB2 Server for VM and Resource Adapter Virtual Machines

Reason Code Value Dec (Hex)	Return Code Value Dec (Hex)	Reason Code Meaning
04 (04)	12 (0C)	User with this name previously issued HNDIUCV SET
06 (06)	04 (04)	Path was severed normally
07 (07)	04 (04)	Path was severed abnormally
08 (08)	08 (08)	Running single user mode
08 (08)	12 (0C)	HNDIUCV SET has not been issued for this application
12 (0C)	12 (0C)	The application does not own the path
16 (10)	12 (0C)	NAME parameter not specified, or its address is 0
20 (14)	12 (0C)	EXIT parameter not specified, or its address is 0
24 (18)	12 (0C)	PARMLIST parameter not specified, or its address is 0
28 (1C)	12 (0C)	An IUCV SEVER with the IPALL bit on is not allowed
32 (20)	12 (0C)	DECLARE BUFFER already issued by a non-CMSIUCV user; CMSIUCV support cannot be initialized
36 (24)	12 (0C)	Errors occurred while reading the directory for the virtual machine during IUCV initialization
40 (28)	12 (0C)	Unrecognized function name (not SET, REP, CLR, CONNECT, ACCEPT, or SEVER)
101 (65)	12 (0C)	Path id not valid
108 (6C)	12 (0C)	Message id found, message class or path id not valid
110 (6E)	12 (0C)	Message length is negative
118 (76)	12 (0C)	Value in message limit exceeds 255
119 (77)	12 (0C)	DECLARE BUFFER previously issued
2xx (--)	12 (0C)	Error occurred while acquiring or freeing CMS free storage; xx is return code from the CMS free storage or DMSFRET macro.

Codes of the DB2 Server for VM Virtual Machine

Reason Code Value Dec (Hex)	Return Code Value Dec (Hex)	Reason Code Meaning
07 (07)	04 (04)	Connection cannot be completed; originator severed
17 (11)	04 (04)	Message has been purged
105 (69)	12 (0C)	Receive or answer buffer is too short to contain message
125 (7D)	12 (0C)	Fetch protection on receive buffer
126 (7E)	12 (0C)	Addressing exception on receive buffer
127 (7F)	12 (0C)	Fetch protection on reply buffer
128 (80)	12 (0C)	Addressing exception on reply buffer
129 (81)	12 (0C)	No message found
254 (FE)	08 (08)	External interrupt not valid

Codes of the Resource Adapter Virtual Machine

Reason Code Value Dec (Hex)	Return Code Value Dec (Hex)	Reason Code Meaning
01 (01)	04 (04)	Message was rejected by target
10 (0A)	04 (04)	Target communicator has not logged on
11 (0B)	04 (04)	Target communicator has not declared buffer
12 (0C)	04 (04)	Maximum number of connects for target exceeded; no connect by partner
23 (17)	08 (08)	Partner issued a SHUTDOWN
102 (66)	12 (0C)	Path quiesced; no sends allowed
103 (67)	12 (0C)	Message limit exceeded
104 (68)	12 (0C)	Priority messages not allowed on this path

DB2 Server for VM Reason Codes

Reason Code Value Dec (Hex)	Return Code Value Dec (Hex)	Reason Code Meaning
113 (71)	12 (0C)	Maximum number of connects for originator exceeded
115 (73)	12 (0C)	No authorization found
116 (74)	12 (0C)	CP system service name not valid
117 (75)	12 (0C)	Function code in IPFCNCD not valid
121 (79)	12 (0C)	Protection exception on send buffer
122 (80)	12 (0C)	Addressing exception on send buffer
123 (81)	12 (0C)	Protection exception on answer buffer
124 (82)	12 (0C)	Addressing exception on answer buffer

Codes Arising during Block I/O Operations

Reason Code Value Dec (Hex)	Return Code Value Dec (Hex)	Reason Code Meaning
04 (04)	12 (0C)	High-order byte of R1 was not zero
08 (08)	12 (0C)	HNDIUCV SET has not been issued for this application
12 (0C)	12 (0C)	DASD was not reserved with the RESERVE command (from DISKID)
12 (0C)	12 (0C)	The application does not own the path (from CONNECT)
16 (10)	12 (0C)	NAME parameter not specified, or its address is 0
24 (18)	12 (0C)	PARMLIST parameter not specified, or its address is 0
28 (1C)	12 (0C)	ddname not defined (from DISKID)
28 (1C)	12 (0C)	An IUCV SEVER with the IPALL bit on is not allowed (from CONNECT)
36 (24)	12 (0C)	Disk not accessed
40 (28)	12 (0C)	Unrecognized function name (not SET, REP, CLR, CONNECT, ACCEPT, or SEVER)
100 (64)	12 (0C)	Disk not attached
1xx (--)	12 (0C)	I/O error occurred, where xx is the return code from DIAGNOSE 18 (Count-Key-Data devices)
108 (6C)	12 (0C)	Message id found, message class or path id not valid
113 (71)	12 (0C)	Maximum number of connects for originator exceeded
116 (74)	12 (0C)	CP system service name not valid
117 (75)	12 (0C)	Function code in IPFCNCD not valid
118 (76)	12 (0C)	Value in message limit exceeds 255
151 (97)	12 (0C)	Virtual device not defined
152 (98)	12 (0C)	Virtual device not supported
153 (99)	12 (0C)	Blocksize not supported
154 (9A)	12 (0C)	IUCV path already exists
155 (9B)	12 (0C)	Connection not using PRMDATA=YES
175 (AF)	04 (04)	Virtual device is read only
2xx (C-)	12 (0C)	I/O error occurred, where xx is the return code from DIAGNOSE 20 (Fixed Block Architecture devices) (from DISKID)
2xx (C-)	12 (0C)	Error occurred while getting or freeing CMS free storage; xx is the return code from free storage or DMSFRET (from CONNECT)

Chapter 22. Debugging Communication Errors

This section describes causes and suggests corrective action for communications errors reported via an SQLCODE of -933 or -30080 or via the message ARI0197E. It includes possible reasons and solutions for the errors most likely to be encountered. It is **NOT** a complete list of errors, and does not attempt to include errors that are not user-correctable.

The following is the descriptive text for SQLCODEs -933 and -30080 and for message ARI0197E:

-933	**The communications path to** *dbname* **is disabled. No further access to this path is possible. Request Code =** *request-code*. **PROTOCOL:** *protocol* **FUNCTION:** *function* **REG15: X'***reg15***' CC:** *condition-code* **IPRCODE: X'***iprcode***' IPCODE: X'***ipcode***' IPWHATRC: X'***ipwhatrc***' IPAUDIT: X'***ipaudit***'**
-30080	**Communication error occurred. PROTOCOL:** *protocol* **FUNCTION:** *function* **REG15: X'***reg15***' CC:** *condition-code* **IPRCODE: X'***iprcode***' IPCODE: X'***ipcode***' IPWHATRC: X'***ipwhatrc***' IPAUDIT: X'***ipaudit***'**
ARI0197E	**A communication error occurred.** **Function:** *function-code* **Return Code:** *return-code* **Reason Code:** *reason-code* **CMSIUCV Reg15:** *X'reg15'* **CC:** *condition-code* **IPRCODE:** *X'iprcode'* **IPCODE:** *X'ipcode'* **IPWHATRC:** *X'ipwhatrc'* **IPAUDIT:** *X'audit'*

Each of these SQLCODES includes a set of error indicator fields — PROTOCOL, FUNCTION, REG15, CC, IPRCODE, IPCODE, IPWHATRC, and IPAUDIT. The error indicator fields may or may not have valid values depending on the nature of the error reported. An error indicator field is considered invalid or not applicable if the field contains only dashes ('-' is a single dash character). Anything else is considered a valid value.

DB2 Server for VM uses IUCV, APPC/VM, or TCP/IP to communicate between the application requester and the application server. The **PROTOCOL** field in an SQLCODE -933 or -30080 refers to the communications protocol in effect when the function was invoked. This field will always have a valid value of either IUCV, APPC, or TCP/IP. Message ARI0197E is displayed if an error occurs while DB2 Server for VM is communicating with CRR. PROTOCOL is not displayed in this case because PROTOCOL is always APPC.

While communications functions are generally handled using direct invocation of IUCV and APPC/VM function macros, some functions are handled using the CMS macros CMSIUCV and HNDIUCV. CMSIUCV is used for invoking IUCV or APPC/VM functions through CMS. HNDIUCV is used for initiating or terminating a program's IUCV or APPC/VM environment. When an error is reported via an SQLCODE -933 or -30080 or message ARI0197E, the contents of the **FUNCTION** field will always be the name of the last IUCV, APPC/VM, CMSIUCV, or HNDIUCV communication function invoked.

If the **REG15** field contains a valid value, then the function named in the FUNCTION field is either a CMSIUCV or HNDIUCV function. IF REG15 contains dashes, then the function is an IUCV or APPC/VM function. Message ARI0197E shows this field in the "CMSIUCV Reg15".

All fields following the REG15 field (**CC, IPRCODE, IPCODE, IPWHATRC,** and **IPAUDIT**) are IUCV or APPC/VM error indicator fields. The individual fields may or may not have valid values depending on the nature of the error reported. For instance, if the error is returned when using a CMSIUCV function, the error may be detected either by CMS or by the IUCV or APPC/VM function it invokes. If the error is detected by CMS, only REG15 would have a valid value. On the other hand, if the error is detected at the IUCV or APPC/VM level, the CC field and at least some of the other error indicator fields would have valid values as well as REG15.

If both the CC field and the REG15 field contain invalid values (dashes), then the error reported to

© Copyright IBM Corp. 1987, 2000

the user is due to the link being SEVERed by either the application requester or an intermediate server (TSAF, AVS, VTAM). If the protocol is APPC/VM, only IPCODE and IPWHATRC will have valid values. IPCODE will be the SEVER code issued by your communications partner. If the protocol is IUCV, none of the error indicators will contain a valid value. For SQLCODEs -933 and -30080, the SQLERRD1 and SQLERRD2 fields may contain information about the cause of the SEVER. These fields are reported in the error message ARI0504I or can be found in the SQLCA. SQLERRD1 will contain the return code and SQLERRD2 will contain the reason code. For message ARI0197E, the **return code** and **reason code** values are displayed in the message. For additional information on diagnosing these SEVER conditions, refer to "VM IUCV Communication Codes" on page 400.

If the **PROTOCOL** field is TCP/IP, the **Request Code** indicates which TCP/IP function was being performed. The remaining fields are not used and contain dashes. The SQLERRD1 field of the SQLCA contains the TCPERRNO that was returned by TCP/IP. The TCPERRNO values can be found in the TCPERRNO H file on the TCP/IP client disk. The following table shows the possible request codes. TCP/IP can report the same TCPERRNO for different functions. The interpretation of the TCPERRNO depends on the function that was performed. The functions and the errors that can be reported can be found in the *IBM TCP/IP V2R4 for VM: Programmer's Reference* manual.

Table 1. Request Codes - TCP/IP

Request Code	TCP/IP Function Performed
1	socket
2	connect
3	accept
4	send
5	recv
6	listen
7	selectex
8	close
9	getservbyname
10	gethostbyaddr
11	gethostbyname
12	gethostid
13	maxdesc
14	getdtablesize
15	ioctl
16	bind
17	inet_addr
18	inet_ntoa
19	FDSET
20	writev

Possible Reasons for TCP/IP Errors

The following table gives possible reasons for and corrective actions to solve some of the TCP/IP errors.

Table 2. Errors Related to a Valid TCPERRNO Request Code

Request Code	TCPERRNO	Possible Reason	Corrective Action
9 - getservbyname	67 - EPROCLIM	TCP/IP Client disk is not available.	Link and access the TCP/IP client disk for your installation. Usually, it is the TCPMAINT 592 disk.
16 - bind	48 - EADDRINUSE	(1) Another server on the same system is already running and has taken the port number you want. (2) The listener socket was not cleaned up properly the last time the server was shutdown.	(1) Use a different port number by updating the ETC SERVICES file or by using the TCPPORT parameter. (2) Re-IPL the server machine to force a proper cleanup.
2 - connect	61 - ECONNREFUSED	The target server is not enabled to handle TCP/IP requesters.	Start the target server with TCP/IP support enabled.

Possible Reasons for Communication Errors

The following tables give possible reasons for and corrective actions to solve some of the errors reported via SQLCODEs -933 and -30080 and via message ARI0197E. They include errors where either the IPCODE field or IPRCODE field contains a valid value for the error. Note that this is simply a list of errors which the user may encounter most frequently, and which are potentially correctable by the user. It is NOT a complete list of errors, and does not attempt to list errors which are not user-correctable.

If the error you encounter is reported using different fields, if the error code is not found in the following tables, or if the corrective action suggested in the tables does not solve the problem, refer to the CMS and CP documentation of the CMSIUCV, HNDIUCV, IUCV or APPC/VM function named in the FUNCTION field. See your system programmer and the bibliography to determine the appropriate VM reference manual for your system. Review the discussion in the previous section. Then use the values from the error indicator fields (REG15, CC, IPRCODE, IPCODE, IPWHATRC and IPAUDIT) and the CMS or CP documentation, to determine the cause of the error. The error you encounter may not be user-correctable. In that case, make a record of what went wrong and contact the designated IBM support group for your installation.

Table 3. Errors Related to a Valid IPCODE

IPCODE	Possible Reason	Corrective Action
x'0111'	No AVS sessions are available for connection.	Ask the AVS operator to activate another session and retry the connection.
x'0130'	Attempted connection with unlike database server while AR is running with SQLINIT P(SQLDS).	Issue SQLINIT with P(AUTO) or P(DRDA) to communicate in DRDA flows with unlike database.
x'0141'	Either the target server or VTAM has exceeded the maximum number of connections.	Try again when the system is not so busy. If the condition persists, then ask your system administrator to increase the maximum number of connections for either the target server or VTAM as appropriate.
x'0142'	TPN specified in user or system communications directory can not be found, or the server is not logged on.	Ensure that TPN specified for the target server is correct, and that the server is logged on. Reload the communication directory.
x'0160'	Invalid userid/password in user or system communication directory entry for the application server named.	Ensure all security information and passwords are valid, and that any mapping performed by AVS is correct. Reload the communication directory.
x'0210'	Database has severed the link.	Attempt to reconnect and continue processing.
x'0220'	Database has severed the link.	Attempt to reconnect and continue processing.
x'0301'	Invalid LU name specified in the user or system communication directory for the target server.	Ensure LU name in the communication directory is correct. Reload the communication directory.
x'0302'	Invalid mode name specified in user or system communication directory for the target server.	Ensure mode name in the communication directory is correct. Reload the communication directory.
x'0610'	The communications link has been severed.	Attempt to reconnect and continue processing.

Table 4. Errors Related to a Valid IPRCODE

IPRCODE	Possible Reason	Corrective Action
x'0B'	Invalid TPN or application server is not logged on.	Ensure TPN is correct and that the application server is logged on.
x'0D'	Your virtual machine already has the maximum number of connections.	Increase the MAXCONN parameter in the VM directory.

Table 4. Errors Related to a Valid IPRCODE (continued)

IPRCODE	Possible Reason	Corrective Action
x'0E'	Your application server or intermediate server already has the maximum number of connections.	Try again when the system is not so busy. If the condition persists, then ask your system administrator to increase the maximum number of connections for either the target server or VTAM as appropriate.
x'0F'	Your machine is not authorized to connect to the resource.	Ensure that your AVS machine's VM directory and your application's server VM directory are set up correctly.
x'28'	Invalid locally known LU name.	Ensure LU name in the communication directory is correct. Reload the communication directory.
x'29'	Invalid mode name specified in user or system communication directory for the target application server.	Ensure mode name in the communication directory is correct. Reload the communication directory.
x'34'	There is no APPCPASS directory statement, and your connection attempted COMDIR resolution requiring either userid or password from the APPCPASS directory statement.	Ensure combination of communication directory and APPCPASS is valid. Reload communication directory.

Chapter 23. Sequential File Errors

This section describes the causes and suggests corrective actions for error messages due to errors encountered by the DB2 Server for VM system while processing sequential input or output files in a VM system. The error messages are displayed on the virtual machine console or in a print file.

Problem Determination Notes:

- The primary error code value is the return code value from entry point ARISYSD5 in module ARISYSEC.
- The secondary error code value is the field EREASON in parameter structure ARIFDESC used in ARISYSD5 calls.

Primary-Secondary Error Code Pairs

In the following list, the Primary Error Code is listed first, and the Secondary Error Code is listed second. For any pair of codes not listed, see the "Other" item at the end of the list.

8 0 CMS/OS QSAM was not able to successfully open the file. A CMS DMS error message describing the error condition has been displayed on the virtual machine console. The DMS message includes either the ddname or the device address (cuu) of the file (or both).

Corrective Action: Refer to *VM/ESA: System Messages and Codes* for the displayed DMS message. Take appropriate corrective action and rerun the DB2 Server for VM program.

8 4 A DB2 Server for VM program was not able to open a CMS/OS QSAM file because no CMS FILEDEF command was submitted for the file. The database manager requires a CMS FILEDEF command for every sequential file except for preprocessor workfiles (files with ddnames SYS001, SYS002, and SYS003).

Corrective Action: When starting the program that encountered the error, either

- Supply a CMS FILEDEF command for the file identified by the error message, or
- Supply parameters to the CMS EXEC which invokes the program to cause the CMS EXEC to generate the CMS FILEDEF command for the file identified by the error message.

8 36 A DB2 Server for VM preprocessor attempted to open a workfile. The action failed because an A-disk was not accessed by the virtual machine.

Corrective Action: Ensure that the virtual machine has write access to an A-disk and rerun the failing DB2 Server for VM preprocessor.

12 0 or 3
An uncorrectable I/O error occurred during access of the CMS/OS file identified in the error message. If the secondary error code is a 3, the error is an uncorrectable CMS file read error on your A-disk for a preprocessor workfile. Such errors can be caused by:

- Virtual device not accessed by the virtual machine
- Virtual device not accessed for writing, if the file is an output file
- Hardware-detected input/output errors.

Corrective Action: If a CMS error message for the file (ddname or virtual device address) is displayed on the virtual machine console, take the corrective action suggested or recommended for that message.

For Tape Files:

- If the error occurs during opening of a standard label input or output file, or during reading (GET) of an unlabeled input file, you may be reading the tape volume at the wrong density, or you may be attempting to open or read the file with an LRECL, BLOCKSIZE or RECFM that does not match those that the file was created with. See the CMS FILEDEF command, and specify the proper options.

© Copyright IBM Corp. 1987, 2000

- If the error occurs during opening of a standard label output file, or during writing (PUT) of an unlabeled output file, the tape volume may have been mounted file protected.
- If you get no CMS error message for the file:
  1. Use the CMS QUERY FILEDEF command to get the virtual device address of the file.
  2. Use the CMS QUERY TAPE command to get the real address of the tape unit.
  3. Query the CP operator (using the CP MSG OP command) for an I/O error message with your virtual machine ID and the real address of the tape unit.

For DASD Files: If no CMS error message has been displayed:

- If your virtual machine is reading from a shared disk, make sure that your virtual machine is not reading a CMS file while another virtual machine is updating that file.
- If you know only the ddname of the CMS file:
  1. Use the CMS QUERY FILEDEF command to get the CMS file name and file type of the file.
  2. Use the CMS LISTFILE command to see if that file is on a particular disk. Specifying the file mode as * causes all accessed disks to be searched for the file. You can also use the CMS STATE command to see if a CMS file exists.

     Note: If the file is a preprocessor workfile (ddname is SYS001, SYS002, or SYS003), it has no FILEDEF. The CMS file created is: ddname SQLWORK A3.
- Use the CMS QUERY DISK access-letter command to get the minidisk volume serial number and the virtual cuu of the disk volume. If the required disk is not accessed, that is the problem.
- Use the CP QUERY DASD command to get the virtual device address, the real device type, the real volume serial number, and the access type (read or write). If the file is an output file and your disk volume is not accessed for writing, that is the problem.
- Query the CP operator (using the CP MSG OP command) for an I/O error message for your virtual machine ID and the real volume serial number and the virtual device address.

For Files Assigned to a Virtual Reader, Printer, or Punch:

- Use the CP QUERY cuu command to verify that the virtual device is ready. (Reader is device C, punch is device D, and Printer is device E.) If the virtual device is not ready, that is the problem. Use the CP READY cuu command to ready the virtual device.
- For the virtual reader, use a CP QUERY RDR ALL command to:
  1. Verify that the reader is not empty. If it is empty, that is the problem.
  2. Verify that the file in the reader is not held. If it is held, that is the problem.
  3. Verify that the class of the reader file is the same as the spool class of your virtual reader. If the spool classes do not match (except for * spool class), that is the problem.
- Otherwise, save all error messages and refer the problem to your VM system programmer.

20 0 DB2 Server for VM system error.

Corrective Action: Save the error message and any other error messages displayed, and give the material to your system programmer.

System Programmer Response: Determine the service level of the database manager, and report this problem to the designated support group for your installation.

24 1, 22, or 25
The virtual machine does not have enough virtual storage to process the file.

Corrective Action: Either run the failing program in a larger virtual machine or change the program parameters so that it requires less virtual storage. If you are running a program with the database

manager in single user mode, you may be able to run it in a smaller virtual machine while the database manager is running in multiple user mode. Consult with your system programmer to help you. (You can use the CP QUERY STORAGE command to display the virtual storage size of your virtual machine. You can set the size of your virtual machine by using the CP DEFINE STORAGE command before you use the CP IPL CMS command.)

24 2, 3, 4, or 9
An DB2 Server for VM or VM system error occurred in a virtual storage request during the process of opening a file.

Corrective Action: Save the error message and any other error messages displayed. For secondary error code 9, give the material to your VM system programmer. For other secondary error codes, give the material to your database manager programmer.

System Programmer Response: The secondary error code is the CMS free storage macro instruction return code. If a user program is running with the DB2 Server for VM program, try to verify that it is not damaging CMS storage pointers (secondary error codes 2 and 3).

For secondary error code 9, determine the service level of VM and report the problem to the designated VM support group for your installation.

For other secondary error codes, determine the service level of the database manager and report the problem to the designated DB2 Server for VM support group for your installation.

28 0 A DB2 Server for VM program detected a wrong-length record (Channel Status Word status bit for incorrect length) while reading a CMS/OS QSAM file (using a QSAM GET macro instruction).

Corrective Action: Ensure that the program is reading the file it expects. Ensure that (where applicable) the correct blocksize, record format, and logical record length were specified for the file. If the program requires a specific blocksize, logical record length, maximum logical record length, or record format, ensure that the input file meets these specifications. Check the program that created the file to ensure that it used the correct blocksize, (maximum) logical record length, and record format. If you find no inconsistencies, request the system programmer to determine the service level of the database manager and report this problem to the designated support group for your installation.

28 4 An end of file was detected before the last segment of a spanned record was obtained while reading a spanned record file (using a CMS/OS QSAM GET macro instruction).

Corrective Action: The program that created the file created an erroneous record. If that program was a DB2 Server for VM program, refer this problem to your system programmer.

System Programmer Response: Determine the service level of the database manager and report the problem to the designated support group for your installation.

32 0 DB2 Server for VM system error.

Corrective Action: Save the error message and any other error messages displayed, and give the material to your system programmer.

System Programmer Response: Determine the service level of the database manager, and report this problem to the designated support group for your installation.

44 10 The maximum number of files supported by CMS on a minidisk (your A-disk) has been exceeded. The DB2 Server for VM preprocessor workfile cannot be created. The maximum number of files supported by CMS on a minidisk depends on the minidisk physical blocksize and device type. Refer to the discussion of the CMS file system in the *VM/ESA: CMS User's Guide*.

Corrective Action: Do one of the following and rerun the DB2 Server for VM preprocessor.

- Delete several non-essential CMS files from your A-disk.

- Reconfigure your virtual machine to use a different CMS minidisk as your A-disk.
- Redefine/reformat your A-disk to use a different physical blocksize or device type. You may need help from your VM system programmer.

44 13 Your A-disk is full. No more records can be written on it by the DB2 Server for VM preprocessor.

Corrective Action: Do one of the following and rerun the DB2 Server for VM preprocessor.

- Delete several non-essential CMS files from your A-disk.
- Reconfigure your virtual machine to use a different CMS minidisk as your A-disk.
- Increase the size of the CMS minidisk you are using as your A-disk. You may need help from your VM system programmer.

The CMS QUERY DISK A command displays how much used and unused space exists on your A-disk.

44 19 The maximum number of data blocks or logical records permitted for a CMS file has been exceeded. This can occur only on an 800-byte block minidisk where the limit is 16060 data blocks and 65533 logical records.

Corrective Action: To preprocess this particular program, use an A-disk that has a CMS file blocksize other than 800 bytes. Do one of the following and rerun the DB2 Server for VM preprocessor.

- Reconfigure your virtual machine to use a different CMS minidisk as your A-disk. Use the CMS QUERY DISK A command to verify that the CMS file blocksize is other than 800 bytes.
- Reformat your A-disk to use a blocksize other than 800 bytes for CMS files. You may need help from your VM system programmer.

48 0 An error occurred while processing a DB2 Server for VM READ FILE statement in a SYSIN file. Message ARI0069E describing the error condition has been displayed on the virtual machine console. Refer to that message for further explanation.

Corrective Action: Refer to the corrective action for message ARI0069E.

Other DB2 Server for VM or VM system error.

Corrective Action: Save the error message and any other error messages displayed and give the material to your system programmer.

System Programmer Response: Determine the service level of the database manager and report this problem to the designated support group for your installation.

Chapter 24. Include File Access Errors

This section describes the causes and suggests corrective actions for error messages due to errors encountered by the DB2 Server for VM system while sequentially reading card image CMS files. The error messages are displayed on the virtual machine console or in a SYSPRINT print file. These error messages can occur while reading:

- Files requested using the preprocessor SQL INCLUDE command.
- Files requested using the database initialization parameter PARMID=name.
- Files requested using the database SYSIN file READ FILE statement 25
- Files containing the input for database catalog generation (during database generation and catalog index reorganization).

Problem Determination Notes:

- The primary error code value is the return code value from entry point ARISYSD8 in module ARISYSFC.
- The secondary error code value is the field REASON in the parameter structure used in ARISYSD8 calls.
- FIND calls determine if the file exists (or can be accessed) and if the format is valid; FIND calls perform "OPEN-like" functions for the file. GET (or READ) calls sequentially retrieve the records in the file.

Primary-Secondary Error Code Pairs

In the following list, the Primary Error Code is listed first, and the Secondary Error Code is listed second. For any pair of codes not listed, see the "Other" item at the end of the list.

8 28 The requested CMS file could not be found on the disk specified by the file mode letter (or on any accessed CMS disk, if a file mode of * was specified). This error occurs only on a FIND call.

Corrective Action: Verify that you correctly specified the CMS file name (and file type and file mode, if applicable). You can use the CMS LISTFILE command to tell you if a particular CMS file exists on an accessed disk. You can use the CMS QUERY DISK command to display the VOLID (LABEL field) and access letters (M field) of all the disks accessed by your virtual machine. If necessary, access the missing CMS disk or create the missing CMS file before rerunning the database manager program.

12 3 An I/O error occurred while reading a CMS file with a GET call. A CMS error message (prefix is DMS) may be displayed on your virtual machine console to give you additional information.

Corrective Action: If you received a CMS message about the I/O error, take the corrective action suggested by that message. If your virtual machine is reading a CMS file while another virtual machine is updating that file, your virtual machine may get an I/O error. If your virtual machine is reading from a shared disk, make sure that this situation has not occurred. Otherwise:

- If you do not know the file mode letter for the file, use the CMS LISTFILE command to obtain it.
- Use the CMS QUERY DISK access-letter command to obtain the minidisk volume serial number (LABEL field) and virtual cuu.
- Use the CP QUERY DASD command to obtain the real device type and real volume serial number of the virtual cuu.
- Use the CP MSG OP command to query the CP operator for an I/O error message with your virtual machine ID and the real volume serial number and the virtual cuu. Take the corrective action suggested by the CMS or CP error message. You may need help from your VM system programmer.

24 1 or 25
The virtual machine does not have enough virtual storage to process the file. Secondary Error Code 1 occurs for a FIND call. Secondary Error Code 25 occurs for a GET call.

© Copyright IBM Corp. 1987, 2000

Corrective Action: Either run the failing program in a larger virtual machine or change the program parameters so that it requires less virtual storage. If you are running a program with the database manager in single virtual machine mode, you may be able to run it in a smaller virtual machine while the database manager is running in multiple virtual machine mode. Consult with your system programmer to help you. (You can use the CP QUERY STORAGE command to display the virtual storage size of your virtual machine. You can set the size of your virtual machine by using the CP DEFINE STORAGE command before you use the CP IPL CMS command.)

24 2, 3, 4, or 9
A DB2 Server for VM or VM system error occurred in a virtual storage request during the FIND call for the file.

Corrective Action: Save the error message and any other error messages displayed. For secondary error code 9, give the material to your VM system programmer. For other secondary error codes, give the material to your database manager programmer.

System Programmer Response: The secondary error code is the CMS free storage macro instruction return code. If a user program is running with the DB2 Server for VM program, try to verify that it is not damaging CMS storage pointers (secondary error codes 2 and 3).

For secondary error code 9, determine the service level of VM and report the problem to the designated VM support group for your installation.

For other secondary error codes, determine the service level of the database manager and report the problem to the designated DB2 Server for VM support group for your installation.

28 0 The requested CMS file either contains variable-length records or fixed-length records that are not 80 bytes long. For this function, only fixed-length 80-byte records can be used. This error occurs only on a FIND call.

Corrective Action: Verify that you have specified the correct CMS file. If you have not, correct the file specification. Otherwise, re-create the CMS file with fixed-length 80-byte records. (You can use the CMS LISTFILE command with the LABEL option to display the record format and logical record length of the specified file.)

36 4 The specified CMS file name contains an asterisk as its first character. You are not permitted to use the '*' default for the CMS file name; file names must contain only letters and numbers (up to 8 characters). Alphabetic characters must be upper case; if they are supplied from your terminal, VM folds them to upper case. This error occurs only on a FIND call.

Corrective Action: Correct the file name specification and rerun the DB2 Server for VM program.

36 8 The specified CMS file type contains an asterisk as its first character. You are not permitted to use the '*' default for the CMS file type; file types must contain only letters and numbers (up to 8 characters). Alphabetic characters must be upper case; if they are supplied from your terminal, VM folds them to upper case. This error occurs only on a FIND call.

Corrective Action: Correct the file type specification and rerun the DB2 Server for VM program.

If the DB2 Server for VM program supplies the CMS file type (or optionally does so if you omit it, and you did omit it), this is a DB2 Server for VM system error. Gather all program error messages and report this problem to your system programmer.

System Programmer Response: If a DB2 Server for VM system error has occurred, determine the service level of the database manager and report the problem to the designated support group for your installation.

36 20 The specified CMS file name or file type contains a character that is not valid. File names and file types must contain only letters and numbers (up to 8 characters). Alphabetic characters must be upper case; if they are supplied from your terminal, VM folds them to upper case. This error occurs only on a FIND call.

Corrective Action: Correct the file name or file type specification and rerun the DB2 Server for VM program.

If your specification is valid and the DB2 Server for VM program supplies the file type (or both), this is a DB2 Server for VM system error. Gather all program error messages and report this problem to your system programmer.

System Programmer Response: If a DB2 Server for VM system error has occurred, determine the service level of the database manager and report the problem to the designated support group for your installation.

36 24 The specified CMS file mode is not valid. It must be one of these:

- '*', or
- An uppercase alphabetic letter (A-Z), or
- An uppercase alphabetic letter (A-Z) and a digit (no blank between) from 0 to 5.

Alphabetic characters must be upper case; if they are supplied from your terminal, VM folds them to upper case. This error occurs only on a FIND call.

Corrective Action: Correct the file mode specification and rerun the DB2 Server for VM program.

If the DB2 Server for VM program supplies the CMS file mode (or optionally does so if you omit it, and you did omit it), this is a DB2 Server for VM system error. Gather all program error messages and report this problem to your system programmer.

System Programmer Response: If a DB2 Server for VM system error has occurred, determine the service level of the database manager and report the problem to the designated support group for your installation.

40 36 The CMS minidisk identified by the file mode letter is not accessed by your virtual machine. This error occurs only on a FIND call.

Corrective Action: If you have specified the wrong file mode to the DB2 Server for VM program, rerun the DB2 Server for VM program with the correct file mode specification. Otherwise, ensure that the required CMS minidisk is defined to your virtual machine (using the CP LINK or DEFINE command), and ensure that it is defined to CMS with the desired access (file mode) letter (using the CMS ACCESS command) before rerunning the DB2 Server for VM program.

Other DB2 Server for VM or VM system error.

Corrective Action: Save the error message and any other error messages displayed and give the material to your system programmer.

System Programmer Response: Determine the service level of the database manager and report this problem to the designated support group for your installation.

Include File Access Errors

Chapter 25. Limit Errors

When limit errors are detected, message ARI0039E is issued. The text of the message is:

ARI0039E Application server limit error occurred - ARIxxxx *nn*

Explanation: The following are the conditions that cause the message to be issued where **ARIxxxx nn** is:

ARICCRA *nn*
(See note 1.)

- **01** Get working storage error (DSCAREA)
- **02** Get working storage error (YTABLE1)
- **04** Get working storage error (SCANS)
- **06** Working storage initialization error (DSCAREA)
- **07** Get working storage error (ARIYMBB)
- **08** Get working storage error (Communication Manager Parameter List)
- **09** Get working storage error (OMB Parameter List)
- **10** Get working storage error (RDAREA)
- **11** Get working storage error (SQLCA)
- **12** Get working storage error (TBASE)
- **14** Get working storage error (RI area)
- **15** Get working storage error (default input buffer)
- **16** Get working storage error (default output mailbox buffer)
- **17** Get working storage error (Parser)
- **18** Get working storage error (Generator)
- **19** Get working storage error (Conversion)
- **20** Get working storage error (DRRMSTR)
- **21** Get working storage error (DCAREA)
- **22** Get working storage error (ASPAREA)
- **25** Get working storage error (TRANS)
- **26** Get working storage error (PCAREA)

ARICDSP *nn*

- **03** All agents waiting for buffers, and SQL is out of page buffers

ARICEXP *nn*
(See note 1.)

- **01** Insufficient storage
- **02** Insufficient storage

ARICIP2 *nn*
(See note 1.)

- **13** Get working storage error (translation table)

ARICIP3 *nn*
(See note 1.)

- **15** Error loading program ARIGCAT or ARISEGB (not enough virtual storage or program not found)
- **18** Error loading the user application program in single user mode (not enough virtual storage or program not found)
- **19** Error loading the program. The application program was loaded above the 16MB line in single user mode when the database manager was running AMODE 24. See message ARI0021E for corrective action.

ARICIPB *nn*
(See note 1.)

- **01** Working storage initialization error (DSCAREAP)
- **02** Get working storage error (WAITM Table)
- **03** Get working storage error (DCES)
- **04** Get below working storage error (HSDS, CID)
- **05** Get below working storage error (TRACE FDESC)
- **06** Get working storage error (GLOBAL Tables)
- **10** Get working storage error (Communication Block Queue Elements and VMCBLOCKs)
- **14** Get working storage error (Communication Block Queue Header)
- **17** Get below working storage error (WAITECB Parameter List)
- **19** Get working storage error (accounting records)
- **21** Get working storage error (Guest in-doubt list)
- **22** Get working storage error (Accounting storage queue header)

© Copyright IBM Corp. 1987, 2000

23 Get working storage error (Transproc control)

31 Get working storage error (Accounting record write buffer)

ARICJAC *nn*
(See note 1.)

01 Get working storage error (Accounting records)

ARICSHO *nn*
(See note 1.)

01 Insufficient storage

02 Insufficient storage

ARICSTK *nn*
(See note 1.)

01 Not enough virtual storage to allocate DB2 Server for VM stack storage for agents

ARISDBR *nn*
(See note 1.)

01 Not enough virtual storage for database generation value table

02 Not enough virtual storage for database generation input file control block

ARISDSK *nn*
(See note 1.)

01 Insufficient storage

02 Insufficient storage

ARISFDB *nn*
(See note 1.)

01 Not enough virtual storage for loading directory block during database generation

ARISFMG *nn*
(See note 1.)

01 Insufficient storage

13 Insufficient storage

ARISFM1 *nn*
(See note 1.)

01 Not enough virtual storage for input/output buffer to format database files

02 Not enough virtual storage for VMCBLOCK to format database files

03 Same as 02

04 Same as 02

ARISIIO *nn*
(See note 1.)

01 Not enough virtual storage for directory file buffer during database generation

03 Not enough virtual storage for directory file blocks during database generation

04 Same as 03

05 Same as 03

06 Same as 03

07 Same as 03

08 Same as 03

09 Same as 03

10 Same as 03

ARISRMG *nn*
(See note 1.)

01 Insufficient storage

10 Insufficient storage

ARISUXP *nn*

16 DASD storage assigned to the storage pool is full (See message ARI0201E)

ARIXERO *nn*
(See note 1.)

13 No working storage

21 Same as 13

22 Same as 13

23 Same as 13

ARIYC11 *nn*
(See note 1.)

09 Not enough virtual storage for log buffer

13 Same as 09

17 Not enough virtual storage for log information

24 Same as 17

27 Same as 09

30 Same as 17

35 Not enough virtual storage for log record

52 Same as 17

58 Not enough virtual storage for log

ARIYD48 *nn*

12 DASD storage assigned to the storage pool is full (See message ARI0201E)

13 DASD storage assigned to the storage pool is full (See message ARI0201E)

ARIYD69 *nn*
(See note 1.)

01 Request for working storage for building new row during row update recovery (UNDO/BACKUP or REDO) failed

02 Same as 01

05 Same as 01

ARIYD81 *nn*
(See note 1.)

06 Request for working storage to assemble data manipulation call parameter values for DBSS entry or exit level 2 trace output failed

07 Same as 06

08 Same as 06

09 Same as 06

10 Same as 06

ARIYI14 *nn*

01 Directory buffer pool is exhausted (NDIRBUF parameter) (See message ARI0203E)

ARIYI19 *nn*

01 Page buffer pool is exhausted (NPAGBUF parameter) (See message ARI0203E)

ARIYI22 *nn*
(See note 1.)

03 Not enough storage to allocate VMCBLOCKs for database files

04 Not enough storage to allocate VMCBLOCK pointer array for database files

06 Same as 03

07 Same as 04

08 Same as 03

09 Not enough storage to allocate VMQ pointer array for database files

16 Get working storage error (EXTMAP)

ARIYI29 *nn*

01 DASD storage assigned to the storage pool is full (See message ARI0201E)

02 DASD storage assigned to the storage pool is full (See message ARI0201E)

ARIYI36 *nn*
(See note 1.)

02 Not enough storage to allocate bit map table for DBSPACE directory file allocation blocks

ARIYI63 *nn*

01 Invalid entities table is in new format. The directory must be reset, but there are either transient or invalid indexes.

ARIYI66 *nn*
(See note 1.)

03 Get working storage error (DS2MOD)

ARIYI68 *nn*
(See note 1.)

02 Get working storage error (DS2MOD)

ARIYI70 *nn*
(See note 1.)

01 Get working storage error (YDATSPC)

ARIYI71 *nn*
(See note 1.)

01 Get working storage error (YDBSPMAP)

02 Get working storage error (YDATSPC)

03 Get working storage error (YDATSPC)

ARIYI72 *nn*
(See note 1.)

01 Get working storage error (YPOOLMAP)

ARIYI73 *nn*
(See note 1.)

01 Get working storage error (XLD block)

02 Get working storage error (XLD block)

ARIYI74 *nn*

01 XCONFIG ADDRspace VM directory limit reached, update the VM directory XCONFIG ADDRspace statement

02 XCONFIG ACCEsslist VM directory limit reached, update the VM directory XCONFIG ACCEsslist statement

03 Get working storage error (DSMAP) (See note 1.)

ARIYI84 *nn*
(See note 1.)

01 Get working storage error (SLD blocks)

02 Get working storage error (MLD blocks)

ARIYK12 *nn*
(See note 1.)

03 Not enough storage for deadlock cycle tables

04 Same as 03

05 Same as 03

ARIYK30 *nn*
(See note 1.)

01 Get working storage error (Lock List)

Limit Errors

ARIYK41 *nn*

01 No more LRBs available

ARIYK53 *nn*
(See note 1.)

01 Not enough virtual storage space for SHOW LOCK WANTLOCK command processing.

Insufficient storage for stack storage.

ARIYK54 *nn*
(See note 1.)

01 Not enough virtual storage for SHOW LOCK USER command processing

ARIYK55 *nn*
(See note 1.)

01 Not enough virtual storage for SHOW LOCK DBSPACE command processing

ARIYK56 *nn*
(See note 1.)

01 Not enough virtual storage for SHOW LOCK ACTIVE command processing

04 Same as 01

ARIYK57 *nn*
(See note 1.)

01 Not enough virtual storage for SHOW LOCK GRAPH command processing

04 Same as 01

ARIYL00 *nn*
(See note 1.)

04 Number of log pages exceeds limit. (Note 1 does not apply to this limit error.)

50 Insufficient storage for checkpoint record

51 Insufficient storage for checkpoint record

52 Insufficient storage for log page

54 Insufficient storage for checkpoint record

55 Insufficient storage for log page

ARIYL01 *nn*

05 Database manager log file is full (See message ARI0276E)

ARIYL08 *nn*
(See note 1.)

01 Not enough virtual storage for checkpoint record

ARIYL09 *nn*
(See note 1.)

60 Insufficient storage for checkpoint record

65 Insufficient storage for checkpoint record

ARIYL14 *nn*
(See note 1.)

08 Insufficient storage for checkpoint record

ARIYL17 *nn*
(See note 1.)

01 Get working storage error (File Descriptor Block)

04 Same as 01

ARIYL18 *nn*
(See note 1.)

01 Get working storage error (File Descriptor Block)

15 Same as 01

ARIYL19 *nn*
(See note 1.)

02 Get working storage error

04 Get working storage error

ARIYL21 *nn*
(See note 1.)

01 Not enough virtual storage for restore buffer

15 Not enough virtual storage for file descriptor block

ARIYL23 *nn*
(See note 1.)

01 Not enough virtual storage for archive buffer

15 Not enough virtual storage for archive file descriptor block

ARIYS08 *nn*
(See note 1.)

02 Sort REDO/UNDO - not enough virtual storage for log record

ARIYT00 *nn*

08 Get working storage error (Counter Internal Structure)

09 Get working storage error (Counter Pool Structure)

ARIYT29 *nn*

01 Not enough agents specified (NCUSERS parameter) (See message ARI0231E)

ARIYX02 *nn*

21 DASD storage assigned to the storage pool is full (see message ARI0201E) or internal DBSPACE is full

ARIYX04 *nn*

06 Create index - not enough virtual storage for sort base

ARIYX09 *nn*

04 Get working storage error (See note 1.)

05 The index was at the maximum of eight levels, and the insert would have caused the index to expand to a ninth level.

ARIYX29 *nn*

15 The index was at the maximum of eight levels, and the INSERT would have caused the index to expand to a ninth level.

ARIYX36 *nn*

01 An attempt was made to mark an entity invalid but there are already 16 invalid entities.

04 More entries in the YINVALID table than the maximum number allowed.

ARIYZ00 *nn*
(See note 1.)

02 Insufficient storage

03 Insufficient storage

05 Insufficient storage

09 Insufficient storage (TABSTAT)

Note 1: These limit errors are caused by insufficient virtual storage. See message ARI0059E for corrective action.

Limit Errors

Chapter 26. System Errors

When a system error is detected, message ARI0040E is issued. The text of the message is:

ARI0040E A database manager system error occurred - ARIxxxx nn

Explanation: Communication Link error conditions may have caused the message to be issued. Communication Link errors are VM IUCV errors. The reason code (message ARI0042I) contains the system return code. **When any of these situations occurs, contact the designated support group for your installation**.

Some of the error messages are shown below. ARIxxxx nn in the message is equal to:

ARICCLA nn

01 Communication Link error - DISCONNECT

ARICINT nn

01 No available pseudo agents

02 Communication Link error - SEVER

03 Communication Link error - ACCEPT

04 Same as 02

05 Communication Link error - Unknown interrupt type. The reason code (message ARI0042I) will contain the invalid interrupt type code.

06 Same as 05

07 Same as 02

08 Same as 02

09 Same as 05

ARICINTC ABEND CODE 251

11 Get storage error when acquiring the prereceive buffer

12 Severe communication error when prereceiving a message

13 Free storage error when returning the prereceive buffer

ARICIP1 nn

12 Communication Link error - IUCV HNDIUCV SET error

13 Communication Link error - IUCV SETCMASK error

ARICIP2 nn

08 Communication Link error - IUCV SETCMASK error

ARICJAC

The following SYSTERR code may be set:

03 Cannot find the accounting queue entry for the on-line user

ARICMUD nn

03 Communication Link error - RECEIVE

05 Communication Link error - REJECT

07 Error freeing prereceive buffer

08 Error freeing input mailbox buffer

ARICOMB nn

01 Communication Link error - REPLY

02 Same as 01

03 Communication Link error - Nonzero message length

04 Communication Link error - RECEIVE

ARISIIO nn

05 The database directory has not been properly restored.

ARIYC25 nn

09 The DB2 Server for VM system was rolling back the drop index operation. Message ARI00421 contains the DBSS Return Code. Refer to “Chapter 18. DBSS Return Codes” on page 271 for these codes.

ARIYD77 nn

08 During recovery, an entry was found in the log for a delete, shorten, or update of a tuple, and the tuple does not exist in the database. You will need to use filtered log recovery to BYPASS or ROLLBACK this LUW.

ARIYI07 nn

05 ARIYI07 is using the CP IUCV DASD block I/O function. An IUCV send request has been made, and the Return Code is in message ARI0925E. See message ARI0925E for diagnosing the Return Code.

ARIYL00 nn

62 During warmstart, ARIYL00 tries to read the checkpoint record from the log and

© Copyright IBM Corp. 1987, 2000

does not find a checkpoint at the address pointed to by the directory.

87 Performing a restore, the log disk is checked to find the last checkpoint that occurred before the restore (to determine whether the current log needs to be archived). This error occurs when the record pointed to (by the LOGCID)is not a checkpoint.

ARIYL26 nn

09 During a startup of the application server, ARIYL26 scans the logs for the HSDS (history area) of the logs and there are no HSDS records in the logs. You may have done a format and reserve of the log minidisks. Perform a coldlog and take a new database archive immediately after performing the coldlog. The coldlog will write the HSDS record to the log.

ARIYX29 nn

04 Not enough index pages in the dbspace to accomplish the INSERT

16 Not enough index pages in the dbspace to accomplish the INSERT

NOTE 1

Other situations can occur where message ARI0040E is issued. **When any of these situations occurs, make a record of what went wrong and contact the designated support group for your installation**.

Chapter 27. Hardware Errors

ARI0041E **System hardware error occurred -** *ARIxxxx nn***.**

Explanation: In this message, **ARIxxxx** identifies the DB2 Server for VM module that detected the error, and **nn** is the detection point in that module.

Make a record of what went wrong and contact the designated support group for your installation.

© Copyright IBM Corp. 1987, 2000

Chapter 28. DPROPR MESSAGES

ASN0000E–ASN0203I DPROPR Capture Messages

The following is a list of messages issued by the IBM DPROPR Capture component of the IBM Relational Data Replication Tools for the VM/ESA platform. A brief explanation of the status is provided.

These messages are intended to aid the user, as well as IBM Development and Support Center personnel.

Unless otherwise stated, error messages are issued with a return code of 8.

ASN0000E **INTERNAL ERROR OCCURRED FOR MESSAGE NUMBER =** *message_number*. **ITS SUBSTITUTION FIELDS ARE**

sub1 sub2 sub3 sub4 sub5 sub6 sub7
ACCESS RC= *return_code*.

Explanation: Capture was unable to read the file containing messages. Note that message ASN0000E is issued in English.

User Response: See the messages and codes publication for your platform for information about return codes to determine why Capture could not read the messages file.

- See the error codes for the FSREAD macro in the *VM/ESA: CMS Application Development Reference for Assembler* manual. Capture attempts to read a file called '*filename* MSG', where *filename* is specified on the LANGUAGE parameter of the CAPTURE ASNPARMS file. If the parameter is not specified, Capture attempts to read 'ASNLS001 MSG' by default.

Examine the message description for *message_number* to determine the underlying error.

ASN0001E **The Capture program encountered an SQL error. The routine name is** *routine*; **the SQL request is** *request*; **the table name is** *table_name*; **the SQLCODE is** *sqlcode*; **the SQLERRML is** *sqlerrml*; **the SQLERRMC is** *sqlerrmc*

Explanation: A nonzero SQLCODE was returned when Capture issued an EXEC SQL statement.

User Response: Take action based on the SQLCODE and SQLERRMC descriptive text. For additional information on the specific problem, refer to the appropriate SQLCODE in the section,or use the SQLCODE numbers as the topic for the ISQL HELP command.

ASN0002E **The Capture program could not connect to DB2. The routine name is** *routine*; **the SQLCODE is** *sqlcode*.

Explanation: An error occurred when Capture issued a CONNECT function to the data base.

User Response: Take action based on the SQLCODE. For additional information on the specific problem, refer to the appropriate SQLCODE in the section, or use the SQLCODE numbers as the topic for the ISQL HELP command.

ASN0005E **The Capture program encountered an error while reading the DB2 log. The routine name is** *routine*; **the log sequence number is** *log_sequence_number*; **the return code is** *return_code*; **the reason code is** *reason_code*.

Explanation: An error occurred when Capture read the DB2 log. There might be an SQL error.

A dump has been generated for this message. The return code is for CP Diagnose X'A4'.

User Response: See "Chapter 21. DB2 Server for VM Reason Codes" on page 399 for the appropriate reason code. For more information on CP Diagnosis, see the *VM/ESA: CP Programming Services* manual.

ASN0006E **The Capture program encountered an unexpected log error of unknown log variation. The routine name is** *routine*.

Explanation: An unexpected log error occurred while Capture was processing the DB2 log records. Capture could not determine the type of SQL update associated with the log record.

© Copyright IBM Corp. 1987, 2000

User Response: Contact your IBM Service Representative.

ASN0007E **The Capture program encountered an unexpected log error of an unimplemented data type. The routine name is** *routine***.**

Explanation: The SQL type in the log record data was invalid. A table might have been registered without the Control Center and contains unsupported SQL types (e.g. LONG or large object types).

User Response: Cancel the registration and use the Control Center to register tables, to ensure only valid types are registered. Or, when manually registering a table, ensure that the table has supported SQL types. See the *DB2 Replication Guide and Reference* for information about unsupported data types.

ASN0008I **The Capture program was stopped.**

Explanation: This informational message is issued when the STOP command is entered.

User Response: None required.

ASN0009E **The table was created without the DATA CAPTURE CHANGES attribute. The routine name is** *routine***; the table name is** *table_name***.**

Explanation: The source table was defined without the DCC attribute and the Capture program tried to capture changes for the replication source.

User Response:

1. Stop the Capture program.
2. Delete the replication source.
3. Define the replication source again, if you do not have the **Data capture is full-refresh only** check box selected, the Control Center will alter the source table with the DCC attribute.
4. Start the Capture program.

ASN0010E **The Capture program can not obtain enough storage. The routine name is** *routine***; the storage required is** *amount***.**

Explanation: The Capture program cannot continue processing because not enough free storage is available.

User Response: A request to obtain virtual storage could not be satisfied. You may need to increase the size of the virtual machine in which the Capture program runs.

ASN0013E **The Capture program required a column that was not defined in the change data table. The routine name is** *routine***; the table name is** *table_name***.**

Explanation: The user did not define an IBMSNAP required column in the change data table.

User Response: Ensure that the change data table definition is correct. Refer to the *DB2 Replication Guide and Reference* for more information.

ASN0014E **The processing of the Capture program has fallen below a minimum level. The log record lags current time by** *number* **seconds. The routine name is** *routine***.**

Explanation: Capture terminated because a high DB2 transaction rate caused the Capture program to run slower than the defined minimum level.

User Response: Refer to the *DB2 Replication Guide and Reference* manual for more information on the lag limit. Perform a cold start.

ASN0015E **The Capture program encountered a storage allocation error. The routine name is** *routine***; the storage required is** *amount***.**

Explanation: A storage allocation error was detected; sufficient storage is not available.

User Response: A request to obtain virtual storage could not be satisfied. You may need to increase the size of the virtual machine in which Capture runs.

ASN0016E **The Capture program could not begin capturing changes because there was not an eligible replication source.**

Explanation: The replication source information has not been defined.

Capture started but could not find source tables that were:

- Enabled with the DATA CAPTURE CHANGES option of the CREATE or ALTER TABLE statement.
- Defined as replication sources with the **Data capture is full-refresh only** check box cleared on the Define as Source window.

User Response: Ensure that the register table is defined properly. For more information about the register table, see the *DB2 Replication Guide and Reference* Verify that replication sources have been defined.

ASN0017E The Capture program encountered a severe internal error and could not issue the correct error message. The routine name is *routine*; the return code is *return_code*.

Explanation: An unexpected error occurred while Capture was executing.

User Response: Contact your IBM Service Representative.

ASN0018W The Capture program did not process updates made to the register table rows. The routine name is *routine*; the table name is *table_name*.

Explanation: The user changed a replication source while the Capture program was running and then issued a REINIT command. The register table, which contains a row for each registration, might not match the other replication source control tables.

User Response:
1. Stop Capture.
2. Delete the replication source.
3. Redefine the replication source.
4. Start Capture.

ASN0022E DB2 release *release* is not supported. The routine name is *routine*.

Explanation: The Capture program does not support this release of DB2.

User Response: Run the Capture program with the appropriate release of DB2.

ASN0023I The Capture program successfully reinitialized the register table. The table name is *table_name*.

Explanation: A REINIT command was issued and the updates were successfully made to the Capture internal control information.

User Response: None required. This is an informational message.

ASN0024I The Capture program did not need to reinitialize the register table. The table *table_name* did not change.

Explanation: The REINIT command was issued. No updates were made to the register table since initialization or the last REINIT.

User Response: None required. This is an informational message.

ASN0025I The Capture program reinitialized the register table. The table *table_name* has *number* potentially bad row(s).

Explanation: This message accompanies ASN0018W. Reinitialization was performed as requested despite potential problems reported in ASN0018W.

User Response: See ASN0018W. This is an informational message.

ASN0026W The Capture program could not allocate the trace buffer. The routine name is *routine*; the storage required is *amount*.

Explanation: A storage allocation error was detected; not enough storage is allocated for the trace buffer. The trace buffer is an information-only feature of Capture and the allocated storage is not required for Capture to run.

User Response: A request to obtain virtual storage could not be satisfied. You may need to increase the size of the virtual machine in which Capture runs.

ASN0027W The Capture program is already active.

Explanation: You tried to start more than one Capture program per database.

User Response:

Capture has determined that the resource ID used as a lock is already in use. The resource ID is specified on the ENQ_NAME parameter of the CAPTURE ASNPARMS file. (If the parameter is not specified, Capture attempts to use a resource ID of 'CAPTURE' by default.) You can change the ENQ_NAME parameter in the CAPTURE ASNPARMS to ensure unique values for each CAPTURE program if you want to run Capture for more than one database on a system. ENQ_NAME must match the IUCV *IDENT directory statement for the CAPTURE machine. See the *VM/ESA: CP Command and Utility Reference* manual for more information regarding updating the directory.

ASN0028I The Capture program is suspended by operator command.

Explanation: This informational message is issued when the SUSPEND command is entered. The Capture program has entered a wait state.

User Response: None required.

ASN0029I The Capture program is resumed by operator command.

Explanation: This informational message is issued when the RESUME command is entered. The Capture program has resumed and continued running.

User Response: None required.

ASN0030I The Capture program command or parameter entered by the operator was unrecognized.

Explanation: This message is issued when a command is not recognized by the Capture program. The only valid commands are:

STOP
SUSPEND
RESUME
REINIT
PRUNE
GETLSEQ

No parameters are allowed for the commands.

User Response: Use only valid Capture program commands.

ASN0031E The Capture program tuning parameter table can have only one row. The routine name is *routine*; the table name is *table_name*.

Explanation: The tuning parameter table was not defined correctly or has been updated with invalid rows.

User Response: Refer to the *DB2 Replication Guide and Reference* to determine the correct format of this table. Remove any invalid rows.

ASN0033E The Capture program could not reinitialize the register table. The table name is *table_name*.

Explanation: The Capture program tried to reinitialize, but there was an error in the register table. A user might have tried to update a replication source while the Capture program was running or suspended, and the register table might not match the other control tables.

User Response: This is a secondary message. See any preceding messages for more information about the error. See the *DB2 Replication Guide and Reference* for more information about reinitializing Capture and about the register table.

ASN0034E An incorrect value was supplied for column *column* of the Capture program tuning parameter table. The routine name is *routine*; the table name is *table_name*.

Explanation: The tuning parameters table does not have the correct values. Values might be out of range.

User Response: Refer to the *DB2 Replication Guide and Reference* for more information. Check the lag limit, retention period and commit frequency.

ASN0035W Some rows were found in the register table with an unsupported architectural level. The routine name is *routine*; the table name is *table_name*.

Explanation: The register table version does not match the current version of the Capture program. The current version of the Control Center is not compatible with the version of Capture that you are running.

User Response: Refer to the *DB2 Replication Guide and Reference* to check the required value for the ARCH_LEVEL column in the register table. Verify that the value in the register table at the source server is correct. If not, use compatible versions of the Control Center and the Capture program.

ASN0036E DB2 was terminated abnormally. The routine name is *routine*.

Explanation: DB2 was terminated while the Capture program was still active and the user did not specify the NOTERM start up parameter.

User Response: Start DB2 and start the Capture program.

ASN0100I The Capture program initialization is successful.

Explanation: The Capture program initialization was successfully performed. If Capture was started with the NOTERM option and and DB2 terminates, this message is also issued by Capture after DB2 comes back up.

User Response: None required. This is an informational message.

ASN0101W The Capture program warm start failed because existing data is too old; a cold start will be attempted.

Explanation: The data in the change data tables is older than the value *current_timestamp-lag_limit*. A cold start will be performed.

User Response: None required. This is an informational message. See the *DB2 Replication Guide and Reference* for more information about warm and cold starts to determine why the Capture program could not warm start.

ASN0102W The Capture program will switch to cold start as warm start information is insufficient.

Explanation: A problem occurred during the retrieval of the warm start information. The warm start table data was invalid or not found. A cold start will be performed.

User Response: None required. This is an informational message. See the *DB2 Replication Guide*

and Reference for more information about warm and cold starts to determine why the Capture program could not warm start.

ASN0103I **The Capture program started. The Enq_name is** *enq_name***; the start_type is** *start_type***; the term_type is** *term_type***; the prune_type is** *prune_type***.**

Explanation: This is an informational message that displays the Capture start up options.

User Response: None required.

ASN0104I **Change capture started for owner** *owner***; the table name** *is table_name* **at log sequence number** *log_sequence_number***.**

Explanation: Capture was started for the table owner and table name at the specified log sequence number (LSN). This message is issued for each base table for which Capture captures changes.

User Response: None required. This is an informational message.

ASN0105I **Data that has been copied was pruned from the change data table** *table_name* **and the unit-of-work table.**

Explanation: This is an informational message to indicate the pruning operation was performed.

User Response: None required.

ASN0106I **The Capture program is waiting for DB2 to come up.**

Explanation:

If the NOTERM option is specified on the ASNCCP command, Capture will wait for DB2 to come back up when DB2 fails. The operator can stop Capture with the STOP command while it is waiting.

User Response: None required. This is an informational message.

ASN0116I **The Capture program did not reinitialize the tuning parameters table. The routine name is** *routine***; the table name is** *table_name***.**

Explanation: The REINIT command was issued, but tuning parameter information from the IBMSNAP_CCPPARMS table was not available. The previous tuning parameter values were retained.

User Response: None required. This is an informational message.

ASN0117I **Warm start control information was not saved. The routine name is** *routine***; the table name is** *table_name***.**

Explanation: An error occurred that prevented warm start information from being saved in the IBMSNAP_WARM_START table. Warm start can be attempted and can take longer because backup sources will be used.

User Response: None required. This is an informational message.

ASN0120I **DataPropagator Relational Log Read Feature Trace Buffer contents:**

Explanation: This header appears prior to the trace buffer, which is printed in case of severe error.

User Response: None required. This is an informational message.

ASN0121E **The Capture program warm start failed because existing data is too old; the Capture program will terminate.**

Explanation: The time of the warm start information exceeded LAG_LIMIT.

User Response: No response required; the Capture program will terminate because WARMNS was specified.

ASN0122E **An error occurred while reading the warm start information or DB2 log. The Capture program will terminate.**

Explanation: A problem occurred during retrieval of the warm start information. The warm start table data may be invalid.

User Response: No response required; the Capture program is terminating because WARMNS was specified.

ASN0123I **The highest log sequence number of a successfully captured log record is** *log_sequence_number*

Explanation: The Capture program saved the highest log sequence number (LSN) in the warm start table. This is the point at which the Capture program finished successfully processing the log data.

User Response: No response required; this message accompanies termination.

ASN0124I **The prune command was accepted; the pruning action is queued.**

Explanation: This informational message is issued when the PRUNE command is entered. The Capture program has queued the request. Capture will prune

the change data (CD) table and the unit-of-work (UOW) table.

User Response: No response required.

ASN0125I **The current log sequence number of successfully captured log records is** *log_sequence_number***. The log timestamp is** *timestamp***.**

Explanation: The Capture program is processing the DB2 log at the log sequence number provided.

User Response: None required. This is an informational message only.

ASN0200E **An incorrect parameter** *parameter* **was passed to the Capture program.**

Explanation:

For VM,

- An incorrect parameter was specified on the ASNCCP start up command, or
- The CAPTURE ASNPARMS file contained an invalid parameter, or
- An incorrect parmameter was used on the :RESID tag in RESID NAMES file for the :DBNAME. For example, the RESID could be too long.

User Response: Verify the values supplied.

ASN0201E **The Capture program encountered a** *platform* **error. The routine name is** *routine***; the function name is** *function***; the return code is** *return_code***.**

Explanation: For *platform* VM,

- For *functionname* LINK, see the *VM/ESA: CP Command and Utility Reference* for an explanation of *return_code*. Capture received an error while attempting to LINK the minidisks identified in the *database SQLFDEF* file.
- For *functionname* FSREAD, FSPOINT, or FSSTATE errors see the *VM/ESA: CMS Application Development Guide for Assembler* Capture received an error while trying to read the *CAPTURE ASNPARMS* or *database SQLFDEF* file. *database* is the database identified on the SQLINIT or SQLGLOB commands, or it defaults to SQLDBA.
- For *functionname* XCIDRM, Capture was unable to obtain the resource ID it uses as a lock to ensure that only one Capture program is active for a DB2 database. See the *VM/ESA: CPI Communications User's Guide* for an explanation for *return_code*. Some reasons for this error include:
  - The virtual machine in which the application is running does not have authority to connect to *IDENT.
  - The virtual machine in which the application is running does not have authority to declare the resource.
- For other *functionname* values, refer to the platform product documentation. In particular:
  VM/ESA: CMS Application Development Reference
  VM/ESA: CMS Application Development Reference for Assembler
  VM/ESA: CMS Command Reference
  VM/ESA: Running Guest Operating Systems

User Response: Correct the error as described in the VM documentation.

ASN0203I **Linking to** *diskname* **minidisk** *diskowner* **as** *vdev*

Explanation: The Capture program is about to issue an internal CP LINK command to the identified database minidisk.

User Response: You may be prompted to enter the minidisk password, if necessary.

Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10594-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Station P300
522 South Road
Poughkeepsie, NY 12601-5400
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the

© Copyright IBM Corp. 1987, 2000

same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements, or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information may contain sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing, or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

C/370
CICS/370
CICS/VSE
DATABASE 2
DataPropagator
DB2
DFSMS
Distributed Relational Database Architecture
DRDA
IBM
MVS
OS/2
QMF
RACF
SAA
SQL/DS
System/370
VM/ESA
VSE/ESA
VTAM

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Bibliography

This bibliography lists publications that are referenced in this manual or that may be helpful.

DB2 Server for VM Publications

- *DB2 Server for VSE & VM Application Programming*, SC09-2889
- *DB2 Server for VSE & VM Database Administration*, SC09-2888
- *DB2 Server for VSE & VM Database Services Utility*, SC09-2983
- *DB2 Server for VSE & VM Diagnosis Guide and Reference*, LC09-2907
- *DB2 Server for VSE & VM Overivew*, GC09-2995
- *DB2 Server for VSE & VM Interactive SQL Guide and Reference*, SC09-2990
- *DB2 Server for VSE & VM Master Index and Glossary*, SC09-2890
- *DB2 Server for VM Messages and Codes*, GC09-2984
- *DB2 Server for VSE & VM Operation*, SC09-2986
- *DB2 Server for VSE & VM Quick Reference*, SC09-2988
- *DB2 Server for VM System Administration*, SC09-2980
- *DB2 Server for VSE & VM Performance Tuning Handbook*, GC09-2987
- *DB2 Server for VSE & VM SQL Reference*, SC09-2989

Related Publications

- *DB2 Server for VSE & VM Data Restore*, SC09-2991
- *DRDA: Every Manager's Guide*, GC26-3195
- *IBM SQL Reference, Version 2, Volume 1*, SC26-8416
- *IBM SQL Reference*, SC26-8415

VM/ESA Publications

- *VM/ESA: General Information*, GC24-5745
- *VM/ESA: VMSES/E Introduction and Reference*, GC24-5837
- *VM/ESA: Installation Guide*, GC24-5836
- *VM/ESA: Service Guide*, GC24-5838
- *VM/ESA: Planning and Administration*, SC24-5750
- *VM/ESA: CMS File Pool Planning, Administration, and Operation*, SC24-5751
- *VM/ESA: REXX/EXEC Migration Tool for VM/ESA*, GC24-5752
- *VM/ESA: Conversion Guide and Notebook*, GC24-5839
- *VM/ESA: Running Guest Operating Systems*, SC24-5755
- *VM/ESA: Connectivity Planning, Administration, and Operation*, SC24-5756
- *VM/ESA: Group Control System*, SC24-5757
- *VM/ESA: System Operation*, SC24-5758
- *VM/ESA: Virtual Machine Operation*, SC24-5759
- *VM/ESA: CP Programming Services*, SC24-5760
- *VM/ESA: CMS Application Development Guide*, SC24-5761
- *VM/ESA: CMS Application Development Reference*, SC24-5762
- *VM/ESA: CMS Application Development Guide for Assembler*, SC24-5763
- *VM/ESA: CMS Application Development Reference for Assembler*, SC24-5764
- *VM/ESA: CMS Application Multitasking*, SC24-5766
- *VM/ESA: CP Command and Utility Reference*, SC24-5773
- *VM/ESA: CMS Primer*, SC24-5458
- *VM/ESA: CMS User's Guide*, SC24-5775
- *VM/ESA: CMS Command Reference*, SC24-5776
- *VM/ESA: CMS Pipelines User's Guide*, SC24-5777
- *VM/ESA: CMS Pipelines Reference*, SC24-5778
- *VM/ESA: XEDIT User's Guide*, SC24-5779
- *VM/ESA: XEDIT Command and Macro Reference*, SC24-5780
- *VM/ESA: Quick Reference*, SX24-5290
- *VM/ESA: Performance*, SC24-5782
- *VM/ESA: Dump Viewing Facility*, GC24-5853
- *VM/ESA: System Messages and Codes*, GC24-5841
- *VM/ESA: Diagnosis Guide*, GC24-5854
- *VM/ESA: CP Diagnosis Reference*, SC24-5855
- *VM/ESA: CP Diagnosis Reference Summary*, SX24-5292
- *VM/ESA: CMS Diagnosis Reference*, SC24-5857

© Copyright IBM Corp. 1987, 2000

- CP and CMS control block information is not provided in book form. This information is available on the IBM VM/ESA operating system home page (http://www.ibm.com/s390/vm).
- *IBM VM/ESA: CP Exit Customization*, SC24-5672
- *VM/ESA REXX/VM User's Guide*, SC24-5465
- *VM/ESA REXX/VM Reference*, SC24-5770

C for VM/ESA Publications

- *IBM C for VM/ESA Diagnosis Guide*, SC09-2149
- *IBM C for VM/ESA Language Reference*, SC09-2153
- *IBM C for VM/ESA Compiler and Run-Time Migration Guide*, SC09-2147
- *IBM C for VM/ESA Programming Guide*, SC09-2151
- *IBM C for VM/ESA User's Guide*, SC09-2152

Other Distributed Data Publications

- *IBM Distributed Data Management (DDM) Architecture, Architecture Reference, Level 4*, SC21-9526
- *IBM Distributed Data Management (DDM) Architecture, Implementation Programmer's Guide*, SC21-9529
- *VM/Directory Maintenance Licensed Program Specification*, GC20-1836
- *IBM Distributed Relational Database Architecture Reference*, SC26-4651
- *IBM Systems Network Architecture, Format and Protocol Reference*, SC30-3112
- *SNA LU 6.2 Reference: Peer Protocols*, SC31-6808
- *Reference Manual: Architecture Logic for LU Type 6.2*, SC30-3269
- *IBM Systems Network Architecture, Logical Unit 6.2 Reference: Peer Protocols*, SC31-6808
- *Distributed Data Management (DDM) General Information*, GC21-9527

CCSID Publications

- *Character Data Representation Architecture, Executive Overview*, GC09-2207
- *Character Data Representation Architecture Reference and Registry*, SC09-2190

DB2 Server RXSQL Publications

- *DB2 REXX SQL for VM/ESA Installation and Reference*, SC09-2891

C/370 Publications

- *IBM C/370 Installation and Customization Guide*, GC09-1387
- *IBM C/370 Programming Guide*, SC09-1384

Communication Server for OS/2 Publications

- *Up and Running!*, GC31-8189
- *Network Administration and Subsystem Management Guide*, SC31-8181
- *Command Reference*, SC31-8183
- *Message Reference*, SC31-8185
- *Problem Determination Guide*, SC31-8186

Distributed Database Connection Services (DDCS) Publications

- *DDCS User's Guide for Common Servers*, S20H-4793
- *DDCS for OS/2 Installation and Configuration Guide*, S20H-4795

VTAM Publications

- *VTAM Messages and Codes*, SC31-6493
- *VTAM Network Implementation Guide*, SC31-6494
- *VTAM Operation*, SC31-6495
- *VTAM Programming*, SC31-6496
- *VTAM Programming for LU 6.2*, SC31-6497
- *VTAM Resource Definition Reference*, SC31-6498
- *VTAM Resource Definition Samples*, SC31-6499

CSP/AD and CSP/AE Publications

- *Developing Applications*, SH20-6435
- *CSP/AD and CSP/AE Installation Planning Guide*, GH20-6764
- *Administering CSP/AD and CSP/AE on VM*, SH20-6766
- *Administering CSP/AD and CSP/AE on VSE*, SH20-6767
- *CSP/AD and CSP/AE Planning*, SH20-6770
- *Cross System Product General Information*, GH23-0500

Query Management Facility (QMF) Publications

- *Introducing QMF*, GC27-0714
- *Installing and Managing QMF for VSE*, GC27-0721
- *QMF Reference*, SC27-0715
- *Installing and Managing QMF for VM*, GC27-0720
- *Developing QMF Applications*, SC27-0718
- *QMF Messages and Codes*, GC27-0717

- *Using QMF*, SC27-0716

Query Management Facility (QMF) for Windows Publications

- *Getting Started with QMF for Windows*, SC27-0723
- *Installing and Managing QMF for Windows*, GC27-0722

DL/I DOS/VS Publications

- *DL/I DOS/VS Application Programming*, SH24-5009

COBOL Publications

- *VS COBOL II Migration Guide for VSE*, GC26-3150
- *VS COBOL II Migration Guide for MVS and CMS*, GC26-3151
- *VS COBOL II General Information*, GC26-4042
- *VS COBOL II Language Reference*, GC26-4047
- *VS COBOL II Application Programming Guide*, SC26-4045
- *VS COBOL II Application Programming Debugging*, SC26-4049
- *VS COBOL II Installation and Customization for CMS*, SC26-4213
- *VS COBOL II Installation and Customization for VSE*, SC26-4696
- *VS COBOL II Application Programming Guide for VSE*, SC26-4697

Data Facility Storage Management Subsystem/VM (DFSMS/VM) Publications

- *DFSMS/VM RMS User's Guide and Reference*, SC35-0141

Systems Network Architecture (SNA) Publications

- *SNA Transaction Programmer's Reference Manual for LU Type 6.2*, GC30-3084
- *SNA Format and Protocol Reference: Architecture Logic for LU Type 6.2*, SC30-3269
- *SNA LU 6.2 Reference: Peer Protocols*, SC31-6808
- *SNA Synch Point Services Architecture Reference*, SC31-8134

Miscellaneous Publications

- *IBM 3990 Storage Control Planning, Installation, and Storage Administration Guide*, GA32-0100
- *Dictionary of Computing*, ZC20-1699
- *APL2 Programming: Using Structured Query Language*, SH21-1056
- *ESA/390 Principles of Operation*, SA22-7201

Related Feature Publications

- *DB2 for VM Control Center Operations Guide*, GC09-2993
- *DB2 Replication Guide and Reference*, SC26-9920

Index

A

abnormal termination codes 399
ARI0001E 3
ARI0002E 3
ARI0003E 3
ARI0004I 3
ARI0005E 3
ARI0006E 4
ARI0007E 4
ARI0008E 4
ARI0009E 5
ARI0010E 5
ARI0011E 6
ARI0012I 6
ARI0013E 7
ARI0014E 7
ARI0015I 7
ARI0016I 8
ARI0017E 8
ARI0018I 8
ARI0019E 9
ARI0020I 9
ARI0021E 9
ARI0022E 9
ARI0023I 9
ARI0024E 10
ARI0025I 10
ARI0026E 10
ARI0027E 10
ARI0028I 10
ARI0029I 11
ARI0030I 11
ARI0031E 11
ARI0032I 11
ARI0034E 11
ARI0035I 11
ARI0036E 11
ARI0037E 12
ARI0038E 12
ARI0039E 12, 415
ARI0040E 12, 421
ARI0041E 12, 423
ARI0042I 13
ARI0043I 13
ARI0044D 13
ARI0045I 13
ARI0046I 13
ARI0047E 15
ARI0048E 15
ARI0049E 16
ARI0050E 16
ARI0052E 16
ARI0053E 16
ARI0054E 16
ARI0055A 16
ARI0056I 17
ARI0057E 17
ARI0058E 17
ARI0059E 18
ARI0060I 18
ARI0061E 18
ARI0063I 19
ARI0064E 19
ARI0065I 19
ARI0066E 19
ARI0067E 19
ARI0068E 20
ARI0069E 20
ARI0081E 20
ARI0082E 20
ARI0083I 20
ARI0084D 21
ARI0085E 21
ARI0086E 21
ARI0087D 22
ARI0088D 22
ARI0089E 23
ARI0090D 23
ARI0091D 24
ARI0093I 24
ARI0094E 24
ARI0095I 24
ARI0096I 24
ARI0097I 25
ARI0098E 25
ARI0099I 25
ARI0100E 26
ARI0101E 26
ARI0102E 26
ARI0103E 26
ARI0104I 26
ARI0105I 27
ARI0106E 27
ARI0107E 28
ARI0110E 28
ARI0113I 28
ARI0114E 28
ARI0115E 29
ARI0116E 29
ARI0117E 29
ARI0118E 29
ARI0119E 30
ARI0120I 30
ARI0121I 30
ARI0122E 30
ARI0123E 30
ARI0124E 30
ARI0125E 30
ARI0126E 30
ARI0127E 31
ARI0128E 31
ARI0129E 32
ARI0130I 32
ARI0131E 32
ARI0132E 32
ARI0133I 32
ARI0134I 32
ARI0135E 32
ARI0136E 33
ARI0137I 33
ARI0140D 33
ARI0141D 34
ARI0142D 34
ARI0143I 34
ARI0145E 34
ARI0146I 34
ARI0150E 34
ARI0152I 35
ARI0153I 35
ARI0154I 35
ARI0155E 35
ARI0156I 35
ARI0157I 35
ARI0158E 36
ARI0159D 36
ARI0160D 36
ARI0161E 36
ARI0162D 37
ARI0164I 37
ARI0165I 37
ARI0172E 37
ARI0173I 37
ARI0174I 37
ARI0175I 38
ARI0176I 38
ARI0177E 38
ARI0178E 39
ARI0179E 39
ARI0180E 40
ARI0181I 40
ARI0182I 41
ARI0183E 41
ARI0184A 41
ARI0197E 41
ARI0198E 42
ARI0200E 43
ARI0201E 43
ARI0202I 43
ARI0203E 44
ARI0204D 44
ARI0205I 44
ARI0206D 44
ARI0207D 44
ARI0208D 45
ARI0210I 45
ARI0211I 45
ARI0212I 45
ARI0213D 46
ARI0214I 46
ARI0215E 46
ARI0216I 46
ARI0217E 47
ARI0218E 47
ARI0219E 47
ARI0220E 48
ARI0222E 48
ARI0223I 48
ARI0225E 48
ARI0226I 48
ARI0229E 48
ARI0230I 48
ARI0231E 49
ARI0232E 49

© Copyright IBM Corp. 1987, 2000

ARI0233E 49
ARI0234E 50
ARI0235I 50
ARI0236I 50
ARI0237I 50
ARI0238I 51
ARI0239I 51
ARI0240I 51
ARI0241E 51
ARI0242E 52
ARI0243I 52
ARI0244E 52
ARI0245E 52
ARI0246D 52
ARI0247D 53
ARI0248E 53
ARI0249E 53
ARI0250D 53
ARI0251D 54
ARI0252I 54
ARI0253D 55
ARI0254I 55
ARI0255I 55
ARI0256E 55
ARI0257I 56
ARI0258E 56
ARI0259I 56
ARI0260I 56
ARI0261I 56
ARI0262E 57
ARI0263D 57
ARI0264D 57
ARI0265I 57
ARI0266A 58
ARI0267D 58
ARI0268D 58
ARI0269D 59
ARI0270A 59
ARI0271A 59
ARI0272E 59
ARI0274I 60
ARI0275D 60
ARI0276E 60
ARI0277E 60
ARI0278E 61
ARI0279E 61
ARI0280A 62
ARI0281I 62
ARI0282I 62
ARI0283I 62
ARI0284I 62
ARI0285I 62
ARI0286I 62
ARI0287I 63
ARI0288I 63
ARI0289I 63
ARI0290I 63
ARI0291I 63
ARI0292I 63
ARI0293I 63
ARI0294A 63
ARI0295A 64
ARI0296A 65
ARI0297A 65
ARI0298E 65
ARI0299A 66
ARI0300A 81
ARI0301A 81
ARI0302E 81
ARI0303E 81
ARI0304I 81
ARI0305E 81
ARI0306E 82
ARI0307I 82
ARI0308E 82
ARI0309E 82
ARI0310E 82
ARI0311I 83
ARI0312E 83
ARI0313E 83
ARI0314E 83
ARI0315E 83
ARI0316E 83
ARI0317E 84
ARI0318D 84
ARI0319D 84
ARI0320I 84
ARI0321E 84
ARI0322E 84
ARI0323I 84
ARI0324E 84
ARI0325E 85
ARI0326I 85
ARI0327I 85
ARI0328I 85
ARI0329I 85
ARI0330E 85
ARI0331E 85
ARI0332E 85
ARI0333I 86
ARI0334I 86
ARI0337D 86
ARI0338I 86
ARI0340D 86
ARI0341I 86
ARI0342A 86
ARI0343A 87
ARI0344D 87
ARI0345I 87
ARI0348D 87
ARI0349E 87
ARI0350E 87
ARI0351E 87
ARI0352I 87
ARI0353D 87
ARI0354I 88
ARI0355I 88
ARI0357D 88
ARI0359I 88
ARI0360W 88
ARI0364E 88
ARI0366I 88
ARI0370D 88
ARI0371D 88
ARI0372D 89
ARI0373A 89
ARI0374A 89
ARI0375A 89
ARI0376I 89
ARI0377E 89
ARI0378E 89
ARI0379I 89
ARI0380I 89
ARI0381E 90
ARI0382I 90
ARI0383E 90
ARI0384I 90
ARI0385I 90
ARI0386I 90
ARI0387I 90
ARI0388A 90
ARI0389I 90
ARI0390I 90
ARI0392E 91
ARI0393E 91
ARI0394E 91
ARI0395E 91
ARI0396D 91
ARI0397D 91
ARI0398E 91
ARI0399I 91
ARI0428E 93
ARI0429I 93
ARI0439E 93
ARI0440E 93
ARI0442E 93
ARI0451I 93
ARI0452I 94
ARI0470E 94
ARI0471I 94
ARI0472E 94
ARI0473I 94
ARI0474I 94
ARI0499I 94
ARI0500I 105
ARI0501I 105
ARI0502I 105
ARI0503E 106
ARI0504I 106
ARI0505I 107
ARI0506I 107
ARI0511E 107
ARI0512E 107
ARI0513I 108
ARI0520E 109
ARI0521E 109
ARI0522E 109
ARI0523E 109
ARI0524E 110
ARI0525E 110
ARI0526A 110
ARI0527E 110
ARI0528I 110
ARI0529I 111
ARI0530E 111
ARI0531E 111
ARI0532I 111
ARI0533E 111
ARI0534E 111
ARI0535E 112
ARI0536I 113
ARI0537E 113
ARI0538E 113
ARI0539I 114
ARI0540E 114
ARI0541E 114
ARI0542E 114
ARI0543I 115
ARI0544E 115
ARI0545E 115
ARI0546I 115

ARI0547I 115
ARI0548E 115
ARI0549E 115
ARI0550E 115
ARI0551I 116
ARI0552I 116
ARI0553E 116
ARI0554E 116
ARI0555E 116
ARI0556E 116
ARI0558I 117
ARI0559E 117
ARI0560I 117
ARI0561I 117
ARI0562I 117
ARI0563I 118
ARI0564E 118
ARI0565I 118
ARI0566E 118
ARI0567E 118
ARI0568E 118
ARI0569E 118
ARI0570I 119
ARI0571I 119
ARI0572E 119
ARI0573E 119
ARI0574E 119
ARI0575E 119
ARI0576E 120
ARI0577E 120
ARI0578I 120
ARI0579I 120
ARI0580E 120
ARI0581E 120
ARI0582E 121
ARI0583E 121
ARI0584E 121
ARI0585E 122
ARI0586I 122
ARI0587I 122
ARI0588E 122
ARI0589E 122
ARI0590E 122
ARI0591E 122
ARI0592E 123
ARI0593E 123
ARI0594E 123
ARI0595E 123
ARI0596E 123
ARI0597E 123
ARI0598I 124
ARI0599E 124
ARI0600A 141
ARI0601E 141
ARI0602D 141
ARI0603I 141
ARI0604A 141
ARI0605E 141
ARI0606E 141
ARI0607E 142
ARI0608E 142
ARI0609E 142
ARI0610D 142
ARI0611E 142
ARI0612E 143
ARI0613E 143
ARI0614A 143
ARI0615D 143
ARI0616E 143
ARI0617E 143
ARI0618I 144
ARI0619E 144
ARI0620I 144
ARI0621E 144
ARI0622I 144
ARI0623D 144
ARI0624A 144
ARI0625D 144
ARI0626E 145
ARI0627E 145
ARI0628E 145
ARI0629E 145
ARI0630E 145
ARI0631E 145
ARI0632D 145
ARI0633A 145
ARI0634A 146
ARI0635D 146
ARI0636A 146
ARI0637D 146
ARI0638D 146
ARI0639E 146
ARI0640I 146
ARI0641I 146
ARI0642I 147
ARI0643I 147
ARI0644E 147
ARI0645E 147
ARI0646D 147
ARI0647D 147
ARI0648A 147
ARI0649A 148
ARI0650E 148
ARI0651E 148
ARI0652E 148
ARI0653E 148
ARI0654E 149
ARI0655I 149
ARI0656I 149
ARI0657I 149
ARI0658E 149
ARI0659I 149
ARI0660I 149
ARI0661E 149
ARI0662I 149
ARI0663I 150
ARI0664E 150
ARI0665E 150
ARI0666I 150
ARI0673I 150
ARI0674E 150
ARI0675E 150
ARI0676E 150
ARI0677I 150
ARI0678E 151
ARI0680I 151
ARI0681E 151
ARI0682E 151
ARI0683E 151
ARI0684I 151
ARI0685I 151
ARI0686E 151
ARI0687E 152
ARI0688D 152
ARI0689E 152
ARI0690I 152
ARI0691E 152
ARI0692E 152
ARI0693E 153
ARI0694I 153
ARI0695I 153
ARI0696E 153
ARI0697E 153
ARI0698E 153
ARI0699E 153
ARI0700I 175
ARI0701E 175
ARI0702I 175
ARI0703I 175
ARI0704I 175
ARI0705E 175
ARI0706E 175
ARI0707E 175
ARI0708I 176
ARI0709I 176
ARI0710E 176
ARI0711I 176
ARI0712I 176
ARI0713I 176
ARI0714I 176
ARI0715E 176
ARI0716E 176
ARI0717I 176
ARI0718I 176
ARI0719I 176
ARI0720I 176
ARI0721I 177
ARI0722E 177
ARI0723E 177
ARI0724I 177
ARI0725I 177
ARI0726I 177
ARI0727I 177
ARI0728I 177
ARI0729I 177
ARI0730I 177
ARI0731D 177
ARI0732A 177
ARI0733I 178
ARI0734I 178
ARI0735I 178
ARI0736I 178
ARI0737I 178
ARI0738I 178
ARI0739I 178
ARI0740I 178
ARI0741E 178
ARI0742I 178
ARI0743I 178
ARI0744E 178
ARI0745E 178
ARI0746I 178
ARI0747I 178
ARI0748E 178
ARI0749I 179
ARI0750I 179
ARI0752I 179
ARI0753I 179
ARI0754I 179
ARI0755I 179
ARI0756I 179

ARI0757I 179
ARI0758D 179
ARI0759I 179
ARI0760A 179
ARI0761I 179
ARI0762I 179
ARI0763A 179
ARI0764I 179
ARI0765I 179
ARI0766I 180
ARI0768I 180
ARI0769E 180
ARI0770E 180
ARI0771E 180
ARI0772I 180
ARI0773I 180
ARI0779A 180
ARI0780I 180
ARI0781I 180
ARI0782I 180
ARI0783I 180
ARI0784I 180
ARI0785I 180
ARI0786A 180
ARI0787I 181
ARI0788I 181
ARI0789I 181
ARI0790I 181
ARI0791I 181
ARI0792I 181
ARI0793I 181
ARI0794I 181
ARI0795I 181
ARI0796I 181
ARI0797I 181
ARI0798I 181
ARI0799I 181
ARI0800I 223
ARI0801I 223
ARI0802I 223
ARI0803E 224
ARI0804E 224
ARI0805E 226
ARI0806E 226
ARI0807E 226
ARI0808I 226
ARI0809I 226
ARI0810E 226
ARI0811I 227
ARI0812I 227
ARI0813I 227
ARI0814E 227
ARI0815I 228
ARI0816E 228
ARI0817I 228
ARI0819I 228
ARI0821E 228
ARI0822E 229
ARI0823E 229
ARI0824E 229
ARI0825E 229
ARI0827I 229
ARI0828I 230
ARI0829E 230
ARI0830E 230
ARI0831I 230
ARI0832I 230
ARI0833E 230
ARI0834E 231
ARI0835I 231
ARI0836I 231
ARI0837I 231
ARI0838E 231
ARI0839E 231
ARI0840E 231
ARI0841E 232
ARI0842E 232
ARI0843I 232
ARI0844E 232
ARI0845I 232
ARI0846I 233
ARI0847I 233
ARI0848E 233
ARI0849E 233
ARI0850I 233
ARI0851E 233
ARI0852I 233
ARI0853I 234
ARI0854E 234
ARI0855I 234
ARI0856E 234
ARI0857E 234
ARI0858E 234
ARI0859E 234
ARI0860E 235
ARI0861E 235
ARI0862E 235
ARI0863E 235
ARI0864E 236
ARI0865E 236
ARI0866E 236
ARI0867E 236
ARI0868I 236
ARI0869I 237
ARI0870I 237
ARI0871E 237
ARI0872E 237
ARI0873E 237
ARI0874I 238
ARI0875I 238
ARI0876E 238
ARI0877E 238
ARI0878E 238
ARI0879I 239
ARI0880I 239
ARI0881E 239
ARI0882E 239
ARI0883E 239
ARI0884I 239
ARI0885E 239
ARI0886E 240
ARI0887E 240
ARI0888E 240
ARI0889E 240
ARI0890E 240
ARI0891E 240
ARI0892E 240
ARI0893E 240
ARI0894I 241
ARI0895E 241
ARI0896E 241
ARI0897E 241
ARI0898I 241
ARI0899I 241
ARI0900E 251
ARI0901E 251
ARI0903E 251
ARI0904E 251
ARI0905E 251
ARI0906E 251
ARI0907E 251
ARI0908E 252
ARI0909E 252
ARI0910E 252
ARI0911E 252
ARI0912E 252
ARI0913E 253
ARI0914E 253
ARI0915I 253
ARI0916E 253
ARI0917E 253
ARI0918I 253
ARI0919D 253
ARI0920D 254
ARI0921I 254
ARI0922I 254
ARI0923E 254
ARI0924I 254
ARI0925E 254
ARI0926E 255
ARI0927E 256
ARI0928E 258
ARI0929E 259
ARI0930E 259
ARI0931E 259
ARI0932E 260
ARI0933E 260
ARI0934E 260
ARI0935E 261
ARI0936I 261
ARI0937I 261
ARI0938E 261
ARI0939E 261
ARI0940I 261
ARI0944D 261
ARI0945I 262
ARI0946I 262
ARI0947I 262
ARI0948I 262
ARI0949I 262
ARI0950E 263
ARI0951E 263
ARI0952E 263
ARI0953E 263
ARI0954E 263
ARI0955E 263
ARI0956E 263
ARI0957E 263
ARI0958E 263
ARI0959E 264
ARI0960E 264
ARI0961E 264
ARI0962E 264
ARI0963E 264
ARI0964E 264
ARI0965E 264
ARI0966E 265
ARI0967E 265
ARI0968E 265
ARI0969I 265
ARI0972E 265

ARI0973E 265
ARI0974E 265
ARI0975E 266
ARI0976E 266
ARI0980A 266
ARI0981D 266
ARI0982I 266
ARI0983A 266
ARI0984E 267
ARI0985E 267
ARI0986E 267
ARI0987E 267
ARI2000E 66
ARI2001I 67
ARI2002I 67
ARI2003I 67
ARI2004I 67
ARI2005I 68
ARI2006I 68
ARI2007D 68
ARI2008I 69
ARI2009E 69
ARI2010I 69
ARI2012I 69
ARI2014E 69
ARI2015I 69
ARI2017E 70
ARI2018E 70
ARI2019E 70
ARI2020I 70
ARI2021E 70
ARI2022I 70
ARI2023I 70
ARI2024I 70
ARI2025E 71
ARI2026I 71
ARI2027I 71
ARI2028E 71
ARI2029E 71
ARI2030E 72
ARI2033E 72
ARI2034I 72
ARI2035I 73
ARI2037D 73
ARI2038E 73
ARI2039E 73
ARI2040E 73
ARI2041I 73
ARI2042E 73
ARI2043E 74
ARI2044D 74
ARI2045E 74
ARI2046E 74
ARI2047E 74
ARI2048D 74
ARI2049I 75
ARI2050I 75
ARI2051I 75
ARI2052I 75
ARI2053E 76
ARI2054I 76
ARI2900E 77
ARI2901I 77
ARI2902E 77
ARI2903E 77
ARI2905E 77
ARI2906I 78
ARI2909I 78
ARI2910I 78
ARI2911E 78
ARI2912E 78
ARI2914E 78
ARI2915E 79
ARI2916E 79
ARI4022E 97
ARI4023I 97
ARI4100I 98
ARI4101I 98
ARI4102I 98
ARI4103I 98
ARI4106I 98
ARI4107I 99
ARI4108E 99
ARI4109E 99
ARI4110E 99
ARI4112I 99
ARI4113I 99
ARI4114I 99
ARI4115I 100
ARI4151I 100
ARI4152E 100
ARI4153I 100
ARI4154I 100
ARI4155I 101
ARI4156I 101
ARI4157E 101
ARI4159I 101
ARI4160E 101
ARI4161E 102
ARI4162I 102
ARI4164I 102
ARI4165I 102
ARI4166I 102
ARI4167E 103
ARI4168I 103
ARI4169E 103
ARI4170E 103
ARI4171I 104
ARI4172I 104
ARI4173I 104
ARI4174E 104
ARI4175I 104
ARI5361E 124
ARI5362E 124
ARI5363E 124
ARI5365E 125
ARI5366E 125
ARI5367E 125
ARI5368E 125
ARI5370E 125
ARI5375E 125
ARI5376E 125
ARI5377E 125
ARI5382E 126
ARI5383I 126
ARI5384E 126
ARI5385E 126
ARI5386E 126
ARI5387E 126
ARI5390E 126
ARI5391E 127
ARI5392E 127
ARI5394E 127
ARI5395E 127
ARI5396E 127
ARI5397E 127
ARI5398E 127
ARI5399I 128
ARI5400I 128
ARI5401E 128
ARI5402I 128
ARI5403E 128
ARI5404E 128
ARI5406I 129
ARI5500I 132
ARI5501I 132
ARI5502I 132
ARI5503I 132
ARI5504I 132
ARI5505I 132
ARI5506I 132
ARI5507I 132
ARI5508I 132
ARI5509I 132
ARI5510I 133
ARI5511I 133
ARI5512I 133
ARI5513I 133
ARI5514I 133
ARI5515I 133
ARI5516I 133
ARI5517I 133
ARI5518I 134
ARI5519I 134
ARI5520I 134
ARI5521I 134
ARI5522I 134
ARI5523I 134
ARI5524I 134
ARI5525I 134
ARI5526I 134
ARI5527I 135
ARI5528I 135
ARI5529I 135
ARI5530I 135
ARI5531I 135
ARI5532I 135
ARI5533I 135
ARI5534I 135
ARI5535I 136
ARI5536I 136
ARI5537I 136
ARI5538I 136
ARI5539I 136
ARI5540I 136
ARI5541I 136
ARI5542I 136
ARI5543I 137
ARI5544I 137
ARI5545I 137
ARI5546I 137
ARI5547I 137
ARI5548I 137
ARI5549I 137
ARI5570I 137
ARI5571I 138
ARI5572I 138
ARI5573I 138
ARI5574I 138
ARI5575I 138
ARI5576I 138

ARI5577I 138
ARI5578I 138
ARI5579I 138
ARI5580I 138
ARI5581I 139
ARI5582I 139
ARI5583I 139
ARI5584I 139
ARI5585I 139
ARI5586I 139
ARI5587I 139
ARI5588I 139
ARI5589I 139
ARI5590I 140
ARI5591I 140
ARI5592I 140
ARI5593I 140
ARI5806E 112
ARI6000I 153
ARI6001E 154
ARI6002E 154
ARI6003E 154
ARI6004E 154
ARI6005I 154
ARI6006E 154
ARI6007D 154
ARI6008D 154
ARI6009D 154
ARI6010D 154
ARI6011I 155
ARI6012A 155
ARI6013D 155
ARI6020E 155
ARI6021E 155
ARI6022D 155
ARI6023E 155
ARI6024I 156
ARI6025E 156
ARI6026E 156
ARI6027D 156
ARI6028I 156
ARI6029I 156
ARI6030D 156
ARI6031E 156
ARI6032E 156
ARI6033E 157
ARI6034D 157
ARI6035D 157
ARI6036I 157
ARI6037E 157
ARI6038I 157
ARI6039E 157
ARI6040E 157
ARI6041E 157
ARI6042I 158
ARI6043I 158
ARI6044D 158
ARI6045E 158
ARI6046D 158
ARI6047E 158
ARI6048E 158
ARI6049E 158
ARI6050E 159
ARI6051I 159
ARI6052I 159
ARI6053I 159
ARI6054E 159
ARI6055I 159
ARI6056I 159
ARI6057I 160
ARI6058I 160
ARI6059I 160
ARI6060I 160
ARI6061I 160
ARI6062E 160
ARI6063I 160
ARI6064I 160
ARI6065E 160
ARI6067I 160
ARI6068I 161
ARI6069A 161
ARI6070I 161
ARI6071I 161
ARI6072E 161
ARI6073I 161
ARI6074I 161
ARI6075E 161
ARI6076I 161
ARI6080I 161
ARI6081E 162
ARI6082I 162
ARI6084I 162
ARI6086I 162
ARI6087I 162
ARI6088D 162
ARI6089I 162
ARI6090E 162
ARI6091I 162
ARI6092I 163
ARI6093E 163
ARI6094I 163
ARI6095I 163
ARI6096E 163
ARI6097E 163
ARI6098I 163
ARI6099I 163
ARI6100D 163
ARI6101A 164
ARI6102A 164
ARI6103A 164
ARI6104E 164
ARI6105D 164
ARI6106E 165
ARI6107E 165
ARI6108I 165
ARI6109I 165
ARI6110D 165
ARI6111A 165
ARI6112A 165
ARI6113A 166
ARI6114A 166
ARI6115E 166
ARI6117E 166
ARI6118I 166
ARI6119E 166
ARI6120I 166
ARI6121I 166
ARI6122E 166
ARI6126D 167
ARI6127I 167
ARI6128I 167
ARI6129D 167
ARI6131I 167
ARI6132I 167
ARI6133E 167
ARI6134A 167
ARI6135D 167
ARI6138E 168
ARI6139E 168
ARI6140I 168
ARI6141E 168
ARI6142E 168
ARI6143I 168
ARI6145D 168
ARI6146D 168
ARI6147D 169
ARI6148I 169
ARI6149E 169
ARI6151E 169
ARI6154E 169
ARI6157E 169
ARI6160E 169
ARI6161I 169
ARI6162E 169
ARI6164A 169
ARI6165A 170
ARI6166E 170
ARI6167I 170
ARI6168I 170
ARI6169E 170
ARI6170I 170
ARI6171E 170
ARI6172I 170
ARI6173I 170
ARI6174I 170
ARI6175E 170
ARI6177E 170
ARI6178I 171
ARI6180I 171
ARI6181I 171
ARI6182A 171
ARI6183D 171
ARI6184D 171
ARI6185E 171
ARI6186D 172
ARI6187E 172
ARI6188A 172
ARI6189I 172
ARI6190D 172
ARI6191D 173
ARI6192D 173
ARI6193D 173
ARI6197E 173
ARI6198D 174
ARI6199E 174
ARI6200D 174
ARI6201I 174
ARI6202E 174
ARI6700I 174
ARI6701I 175
ARI6702I 175
ARI7001E 183
ARI7020E 183
ARI7021E 183
ARI7022E 183
ARI7039D 183
ARI7040I 183
ARI7041D 184
ARI7042E 184
ARI7043I 184
ARI7045E 184

ARI7047I 184
ARI7048I 184
ARI7049E 185
ARI7050I 185
ARI7051E 185
ARI7060I 185
ARI7061I 185
ARI7062E 185
ARI7063I 185
ARI7064I 186
ARI7065I 186
ARI7066I 186
ARI7068I 186
ARI7069I 186
ARI7070I 186
ARI7071I 187
ARI7079I 187
ARI7080A 187
ARI7085E 187
ARI7101E 187
ARI7102E 187
ARI7119E 187
ARI7120E 187
ARI7121E 188
ARI7123E 188
ARI7124E 188
ARI7125E 188
ARI7126E 188
ARI7127I 188
ARI7128E 188
ARI7180E 188
ARI7181I 189
ARI7182E 189
ARI7183E 189
ARI7184I 189
ARI7185I 189
ARI7186I 189
ARI7300E 189
ARI7301E 189
ARI7302E 189
ARI7303E 190
ARI7304E 190
ARI7305E 190
ARI7306E 190
ARI7307A 190
ARI7308A 190
ARI7309E 191
ARI7310E 191
ARI7311E 191
ARI7312I 191
ARI7313I 191
ARI7314I 191
ARI7315E 191
ARI7316E 191
ARI7317I 192
ARI7318E 192
ARI7319I 192
ARI7320I 192
ARI7321E 192
ARI7322I 192
ARI7323I 192
ARI7324E 192
ARI7325I 193
ARI7326A 193
ARI7327E 193
ARI7328E 193
ARI7329E 193
ARI7330E 193
ARI7350E 193
ARI7351E 194
ARI7354E 194
ARI7355E 194
ARI7357E 194
ARI7358E 194
ARI7359E 194
ARI7360E 194
ARI7361I 195
ARI7362E 195
ARI7363E 195
ARI7364E 195
ARI7398I 195
ARI7399I 195
ARI7500E 195
ARI7501E 195
ARI7502I 196
ARI7503I 196
ARI7504E 196
ARI7505E 196
ARI7506I 196
ARI7507E 196
ARI7508E 196
ARI7509I 197
ARI7510I 197
ARI7511I 197
ARI7512I 197
ARI7513E 197
ARI7514E 197
ARI7515E 197
ARI7516I 198
ARI7517E 198
ARI7518E 198
ARI7519E 198
ARI7520E 198
ARI7521I 198
ARI7522I 198
ARI7523E 199
ARI7524I 199
ARI7525I 199
ARI7526E 199
ARI7527I 199
ARI7528I 199
ARI7529I 199
ARI7530E 199
ARI7531I 200
ARI7532I 200
ARI7533I 200
ARI7534E 200
ARI7535E 200
ARI7537E 200
ARI7539I 201
ARI7543I 201
ARI7544I 201
ARI7545E 201
ARI7546E 201
ARI7547E 201
ARI7550D 202
ARI7551I 202
ARI7552E 202
ARI7553E 202
ARI7554I 202
ARI7555I 202
ARI7556E 202
ARI7557I 202
ARI7558E 202
ARI7559I 203
ARI7560E 203
ARI7561E 203
ARI7563E 203
ARI7564E 203
ARI7568E 203
ARI7569E 203
ARI7570I 203
ARI7572E 204
ARI7574E 204
ARI7575E 204
ARI7576D 204
ARI7577D 204
ARI7578I 204
ARI7579I 204
ARI7580I 204
ARI7581I 205
ARI7582I 205
ARI7583E 205
ARI7585I 205
ARI7590I 205
ARI7591I 205
ARI7592I 205
ARI7593I 205
ARI7601I 205
ARI7602D 205
ARI7603I 206
ARI7604E 206
ARI7607I 206
ARI7611E 206
ARI7612E 206
ARI7613I 206
ARI7614E 206
ARI7615E 207
ARI7616I 207
ARI7617E 207
ARI7618E 207
ARI7619E 207
ARI7620I 207
ARI7621I 208
ARI7622E 208
ARI7623E 208
ARI7624E 208
ARI7625E 208
ARI7626E 208
ARI7627E 208
ARI7628I 208
ARI7629E 209
ARI7630E 209
ARI7631E 209
ARI7632E 209
ARI7634E 209
ARI7635I 209
ARI7636D 209
ARI7637E 210
ARI7638I 210
ARI7639D 210
ARI7640E 210
ARI7641D 210
ARI7642I 210
ARI7643E 210
ARI7644E 211
ARI7645E 211
ARI7646E 211
ARI7650E 211
ARI7651I 211
ARI7652I 211

ARI7700I 211
ARI7701E 211
ARI7702E 212
ARI7709I 212
ARI7710I 212
ARI7711I 212
ARI7712I 212
ARI7713I 212
ARI7714I 212
ARI7715I 212
ARI7716I 212
ARI7717I 213
ARI7725E 213
ARI7726E 213
ARI7727I 213
ARI7729I 213
ARI7731I 213
ARI7732I 213
ARI7733E 214
ARI7734E 214
ARI7735I 214
ARI7736E 214
ARI7737I 214
ARI7739E 214
ARI7740E 214
ARI7742E 214
ARI7743I 215
ARI7744I 215
ARI7745I 215
ARI7746I 215
ARI7755I 215
ARI7757I 215
ARI7758I 215
ARI7761I 215
ARI7885E 216
ARI7886E 216
ARI7887E 216
ARI7888E 216
ARI7889E 216
ARI7890E 216
ARI7891E 216
ARI7892E 216
ARI7894I 216
ARI7895E 217
ARI7896E 217
ARI7897E 217
ARI7898E 217
ARI7901E 217
ARI7902I 217
ARI7903E 217
ARI7905I 217
ARI7908I 218
ARI7920E 218
ARI7921E 218
ARI7922I 218
ARI7923D 218
ARI7924I 218
ARI7932I 218
ARI7933I 218
ARI7934E 218
ARI7937E 219
ARI7939E 219
ARI7940E 219
ARI7942E 219
ARI7943E 219
ARI7944E 219
ARI7945I 219
ARI7946E 220
ARI7947E 220
ARI7949I 220
ARI7951I 220
ARI7952E 220
ARI7953E 220
ARI7954E 220
ARI7955I 220
ARI7957E 221
ARI7958I 221
ARI7959I 221
ARI7960I 221
ARI7962E 221
ARI7963E 221
ARI7964I 222
ARI7966I 222
ARI7967E 222
ARI7968E 222
ARI7969I 222
ARI7970I 222
ARI8000I 241
ARI8001E 242
ARI8002I 242
ARI8003I 242
ARI8004I 242
ARI8005I 242
ARI8006E 242
ARI8007I 242
ARI8009E 243
ARI8010E 243
ARI8011I 243
ARI8012E 243
ARI8013I 243
ARI8014I 243
ARI8015E 243
ARI8016I 244
ARI8017E 244
ARI8018E 244
ARI8020E 244
ARI8021E 244
ARI8022E 244
ARI8023E 244
ARI8024E 244
ARI8025E 244
ARI8026E 245
ARI8027E 245
ARI8028I 245
ARI8029E 245
ARI8030E 245
ARI8031E 245
ARI8032I 245
ARI8033E 246
ARI8034I 246
ARI8035I 246
ARI8700I 246
ARI8701I 246
ARI8702I 246
ARI8703I 246
ARI8980I 247
ARI8981I 247
ARI8982I 247
ARI8983I 247
ARI8990I 247
ARI8991I 247
ARI8992I 248
ARI8993I 248
ARI8994I 248
ARI8995I 248
ARI8996I 249
ARI8997I 249
ARI8998I 249
ARI8999E 249
ARICCLA 421
ARICCRA 415
ARICDSP 415
ARICEXP 415
ARICINT 421
ARICINTC 421
ARICIP1 421
ARICIP2 415, 421
ARICIP3 415
ARICIPB 415
ARICJAC 416, 421
ARICMUD 421
ARICOMB 421
ARICSHO 416
ARICSTK 416
ARISDBR 416
ARISDSK 416
ARISFDB 416
ARISFM1 416
ARISFMG 416
ARISIIO 416, 421
ARISRMG 416
ARISUXP 416
ARIXERO 416
ARIYC11 416
ARIYC25 421
ARIYD48 416
ARIYD69 416
ARIYD77 421
ARIYD81 417
ARIYI07 421
ARIYI14 417
ARIYI19 417
ARIYI22 417
ARIYI29 417
ARIYI36 417
ARIYI63 417
ARIYI66 417
ARIYI68 417
ARIYI70 417
ARIYI71 417
ARIYI72 417
ARIYI73 417
ARIYI74 417
ARIYI84 417
ARIYK12 417
ARIYK30 417
ARIYK41 418
ARIYK53 418
ARIYK54 418
ARIYK55 418
ARIYK56 418
ARIYK57 418
ARIYL00 418, 421
ARIYL01 418
ARIYL08 418
ARIYL09 418
ARIYL14 418
ARIYL17 418
ARIYL18 418
ARIYL19 418
ARIYL21 418

ARIYL23 418
ARIYL26 422
ARIYS08 418
ARIYT00 418
ARIYT29 418
ARIYX02 418
ARIYX04 418
ARIYX09 419
ARIYX29 419, 422
ARIYX36 419
ARIYZ00 419
ASN0000E 425
ASN0001E 425
ASN0002E 425
ASN0005E 425
ASN0006E 425
ASN0007E 426
ASN0008I 426
ASN0009E 426
ASN0010E 426
ASN0013E 426
ASN0014E 426
ASN0015E 426
ASN0016E 426
ASN0017E 427
ASN0018W 427
ASN0022E 427
ASN0023I 427
ASN0024I 427
ASN0025I 427
ASN0026W 427
ASN0027W 427
ASN0028I 427
ASN0029I 427
ASN0030I 428
ASN0031E 428
ASN0033E 428
ASN0034E 428
ASN0035W 428
ASN0036E 428
ASN0100I 428
ASN0101W 428
ASN0102W 428
ASN0103I 429
ASN0104I 429
ASN0105I 429
ASN0106I 429
ASN0116I 429
ASN0117I 429
ASN0120I 429
ASN0121E 429
ASN0122E 429
ASN0123I 429
ASN0124I 429
ASN0125I 430
ASN0200E 430
ASN0201E 430
ASN0203I 430

C

catalog maintenance messages 109
clause
  FOR UPDATE 288
  PRIMARY KEY 352
  UNIQUE CONSTRAINT 352
  UNIQUE INDEX 352
  WITH CHECK OPTION 296
CMS EXEC messages 81, 141
codes
  abnormal termination 399
  communication
    IUCV 400
    VM APPC/VM 400
  DB2 Server for VM 399
  DBSS 271
  introduction v
  SQLCODEs 277
  SQLSTATEs 377
  VM transaction abend 269
common error action 223
common messages 77
communication codes
  IUCV 400
  VM APPC/VM 400
communication errors 403
components of DB2 Server for VM
  system, definitions vii

D

database maintenance messages 251
DB2 Server for VM control messages 97
DB2 Server for VM reason codes 399
DB2 Server for VM system control 3
DBS Utility
  messages 223
  rerunning 223
DBSS messages 43
DBSS return codes 271
  -1 271
  -10 272
  -11 272
  -12 272
  -120 274
  -123 274
  -13 272
  -130 274
  -131 275
  -132 275
  -133 275
  -134 275
  -135 275
  -136 275
  -137 275
  -138 275
  -139 275
  -14 272
  -140 275
  -141 275
  -142 275
  -143 275
  -15 272
  -16 272
  -17 272
  -18 272
  -19 272
  -2 271
  -200 275
  -22 272
  -23 272
  -29 272
  -3 271
  -30 272
  -31 272
  -32 272
DBSS return codes 271 *(continued)*
  -38 272
  -39 272
  -4 271
  -40 272
  -400 275
  -401 275
  -402 275
  -403 275
  -404 275
  -405 275
  -406 275
  -407 275
  -409 275
  -41 272
  -410 275
  -42 272
  -43 272
  -5 271
  -58 273
  -59 273
  -6 271
  -64 273
  -7 271
  -70 273
  -71 273
  -79 274
  -8 271
  -80 274
  -81 274
  -82 274
  -83 274
  -85 274
  -86 274
  -87 274
  -89 274
  -9 271
  -90 274
  -91 274
  -92 274
  -97 274
  -98 274
  1 271
  128 271
  16 271
  2 271
  4 271
  4096 271
  64 271
  8 271
debugging communication errors 403
DPROPR Capture messages 425
Dynamic null input host variable 311

E

errors
  action 223
  common 223
  communication 403
  hardware 423
  include file access 411
  limit 415
  sequential file 407
  system 421
Extended Dynamic null indicator
  variable 311

F

flagger messages 131
format, message 1
function
  AVG 291
  COUNT 291
  MAX 291
  MIN 291
  SUM 291

H

hardware errors 423

I

I/O messages 251
include file access errors 411
input host variable
  null 311
introduction 1
ISQL messages 183
IUCV communication codes 400

K

keyword
  ALL 291
  ANY 291
  DISTINCT 291
  SOME 291

L

limit errors 415

M

manual, using 1
message format 1
messages
  catalog maintenance 109
  CMS EXEC 81, 141
  common 77
  database maintenance 251
  DB2 Server for VM control 97
  DBS Utility 223
  DBSS 43
  DPROPR Capture 425
  flagger 131
  I/O 251
  introduction 1
  ISQL 183
  preprocessor 113
  resource adapter 93
  SQL 105
  system control 3
  trace formatter 263

P

preprocessor messages 113

R

reason codes 399
rerunning DBS Utility 223
reserved words
  SQL vii
resource adapter messages 93
return codes
  DBSS 271
  VM system services 399

S

sequential file errors 407
SQL messages 105
SQLCODE
  -084 288
  +100 277
  -101 288
  -102 289
  -103 289
  -104 289
  -105 290
  -106 290
  -107 290
  -108 291
  -109 291
  -110 291
  -111 291
  -112 291
  -114 291
  +114 277
  -115 291
  -116 292
  -117 292
  +117 278
  -118 292
  -119 292
  +12 277
  -120 293
  -121 293
  -124 293
  -125 293
  -126 293
  -127 294
  -128 294
  -129 294
  -130 295
  -131 295
  -132 295
  -133 295
  -134 295
  +134 278
  -135 296
  +135 278
  -136 296
  -137 296
  -140 296
  -141 296
  +142 279
  -144 296
  -149 297
  +149 279
  +150 279
  -15000 370
  -151 297
  +151 279
  -153 297
  -154 297
  +154 279
  -157 297
SQLCODE *(continued)*
  -158 297
  -159 297
  -160 298
  -161 300
  -162 300
  -163 301
  -164 301
  -165 301
  -166 301
  -167 302
  -168 302
  -169 302
  -170 302
  -171 303
  -172 304
  -173 304
  +179 280
  -180 304
  +180 280
  -181 304
  -182 304
  -183 305
  -184 305
  -185 305
  -187 305
  -189 305
  -191 305
  -192 306
  -198 306
  +200 280
  -201 306
  -202 306
  +202 280
  -203 306
  -204 307
  +204 281
  -205 307
  +205 281
  -206 307
  +206 281
  -207 307
  -208 307
  -209 307
  -210 308
  -211 308
  -30000 370
  -30020 371
  -30021 372
  -30040 372
  -30041 372
  -30050 372
  -30051 373
  -30052 373
  -30053 373
  -30060 373
  -30061 373
  -30070 373
  -30071 374
  -30072 374
  -30073 374
  -30074 374
  -30080 374
  -30090 374
  -301 308
  -302 309
  -303 310

SQLCODE *(continued)*
-304 310
+304 282
-305 310
-306 310
-307 310
-309 311
-311 311
-312 311
-313 311
-314 311
-315 315
-316 316
-317 316
-318 316
-319 316
-320 316
-321 316
-322 316
-323 317
-324 317
-325 317
-326 317
-330 317
-331 317
+331 282
-332 318
-334 318
-401 318
+401 282
-402 318
-404 319
+404 282
-405 319
+405 283
-407 319
+407 283
-408 319
+408 283
-409 319
-412 320
-413 320
-414 320
+414 284
-415 320
+415 284
-416 320
+416 284
-417 320
-418 321
-419 321
+419 285
-422 321
-423 321
-424 321
-425 321
-440 322
-444 322
-449 322
-450 323
+464 285
+466 285
-469 323
-470 323
-471 324
-478 324
-480 324

SQLCODE *(continued)*
-482 324
+494 285
-499 324
-501 325
-502 325
-503 325
-504 325
-505 326
-506 326
-507 326
-508 326
-509 326
-510 326
-511 327
-512 327
-513 327
-515 327
-516 327
-517 327
-518 327
-519 328
-520 328
-521 328
-522 329
-523 329
-524 329
-525 329
-530 330
-532 330
-534 330
-535 330
-536 330
-537 330
-538 330
-539 331
+541 285
-542 331
-550 331
-551 331
+551 286
-552 331
+552 286
-554 332
-555 332
-556 332
-557 332
-558 332
+558 286
-560 332
-561 333
-562 333
-564 333
-565 333
-566 333
-568 333
+600 287
-601 334
-602 334
-603 334
-604 334
-605 335
-606 335
-607 335
-608 335
-609 335
-610 335

SQLCODE *(continued)*
-611 336
-612 336
-614 336
-615 336
-616 336
-617 336
-618 337
-619 337
-620 337
-621 337
-622 337
-623 338
-624 338
-625 338
-629 338
-630 339
-632 339
-640 339
-641 339
-644 339
-648 339
-649 340
-656 340
-657 340
-666 340
-667 340
-668 341
+668 287
-669 341
-670 341
-674 341
-681 342
-682 342
-683 343
-684 343
-685 343
-686 343
-687 343
-688 343
-698 344
-699 344
-700 344
-701 344
-702 345
-703 345
-704 345
-705 346
-706 346
-707 346
-708 346
-709 346
-710 346
-711 346
-712 347
-713 347
-714 347
-720 347
-721 348
-751 348
-752 349
-753 349
-754 349
-756 349
-757 349
-758 349
-759 349

SQLCODE *(continued)*
-760 350
-761 350
-762 350
-8001 370
-801 350
-802 351
+802 287
-803 352
-804 352
-805 352
-806 352
-807 353
-809 353
-810 353
-811 353
-813 354
-814 354
-815 354
-816 354
-818 354
-821 354
-823 354
-824 355
-825 355
+825 288
-826 355
-827 356
-828 356
-829 356
-830 356
-831 356
-832 357
-833 357
-834 357
-835 358
-836 358
-841 358
-844 359
-860 359
-861 359
+863 288
-900 359
-901 360
-902 360
-903 360
-906 360
-911 361
-912 361
-913 361
-914 361
-915 362
-916 362
-917 362
-921 362
-923 362
-928 362
-930 363
-931 363
-932 363
-934 364
-935 365
-939 365
-940 365
-941 366
-942 366
-943 366

SQLCODE *(continued)*
-944 367
-945 367
-946 368
-947 368
-948 369
-949 369
-963 369
-965 369
-993 369
-994 370
0 277
introduction 277

SQLSTATE
00000 379
01003 379
01004 379
01501 379
01502 379
01503 379
01504 379
01506 379
01507 379
01508 379
01509 379
01510 379
01511 380
01512 380
01513 380
01515 380
01516 380
01517 380
01519 380
01520 380
01525 380
01532 380
01533 380
01536 380
01539 380
01541 380
01542 380
01543 380
01545 380
01548 380
01555 380
01570 380
01571 380
01572 380
01573 380
01574 380
01575 381
01576 381
01577 381
01578 381
01579 381
01580 381
01581 381
01582 381
01583 381
01584 381
01585 381
01609 381
01610 381
01614 381
02000 381
07001 381
07002 381

SQLSTATE *(continued)*
07003 381
07004 381
07005 382
07006 382
07007 382
08001 382
08003 382
08004 382
0A001 379
0F001 379
21000 382
21501 382
21502 382
21504 382
22001 382
22002 382
22003 382
22007 382
22008 382
22011 383
22012 383
22019 383
22021 383
22024 383
22025 383
22501 383
22502 383
22504 383
22507 382, 383
22509 383, 387
22510 383
22512 383
22515 383, 384
22517 383
22518 383
22519 383
22520 383, 384
22521 383
22522 383
22524 383
23501 383, 391
23502 383
23503 383
23504 383
23505 383
23507 383
23515 384
23520 384
24501 384
24502 384
24503 384
24504 384
24505 384
24506 384
24516 384
25000 384
26501 384
26503 382, 384
26504 381, 384
26505 384
26506 382, 384
26507 384
26508 384
28000 384
2D528 382
2D529 382

SQLSTATE *(continued)*
34000 384
37501 384, 385
37502 384, 386
37503 384, 386
37504 384, 386
37505 385, 386
37506 385, 386
37507 385, 386
37509 385, 386
37510 385, 386
37511 385, 386
37512 385, 386
37514 385, 386
37515 385, 386
37517 385, 386
37518 385, 386
38003 385
39002 385
39501 385
3C000 384
40000 385
40001 385
40003 385
40503 385
42501 385
42502 385
42504 385
42505 385
42506 385
42507 385
42601 385
42602 386
42603 386
42604 386
42605 386
42606 386
42607 386
42609 386
42610 386
42611 386
42612 386
42614 386
42615 386
42617 386
42618 386
42622 386
42701 387
42702 387
42703 387
42704 387
42705 387
42706 387
42707 387
42709 387
42710 387
42711 387
42712 387
42719 387
42724 387
42802 387
42803 387
42805 387
42806 387
42807 388
42808 388
42809 388

SQLSTATE *(continued)*
42810 388
42811 388
42813 388
42815 388
42816 388
42818 388
42819 388
42820 388
42821 388
42823 388
42824 388
42825 388
42826 388
42827 388
42828 389
42829 389
42830 389
42831 389
42832 389
42834 389
42842 389
42852 389
42853 389
42859 389
42862 389
42867 389
42878 389
42884 389
42886 389
42888 389
42889 (formerly 55013) 389
42893 389
42895 390
42901 390
42902 390
42903 390
42905 390
42906 390
42907 390
42908 390
42909 390
42911 390
42912 390
42913 390
42914 390
42915 390
42917 390
42932 390
42937 390
42939 390
42943 391
42961 391
42969 391
44000 391
51001 381, 391
51002 391
51003 391
51004 391
51010 391
51011 379, 391
51012 391
51013 391
51015 391
51016 391
51018 382, 391
51021 391

SQLSTATE *(continued)*
51030 391
52001 387, 391
52002 387, 391
52003 387, 391
52004 387, 391
52005 387, 391
52006 387, 391
52007 387, 391
52008 384, 391
52009 387, 391
52010 387, 391
52011 387, 392
52012 387, 392
52013 384, 392
52017 382, 392
52019 387, 392
53002 387, 392
53003 387, 392
53005 387, 392
53007 388, 392
53008 388, 392
53009 388, 392
53010 388, 392
53011 388, 392
53013 388, 392
53015 388, 392
53016 388, 392
53017 392
53018 388, 392
53019 388, 392
53020 388, 392
53021 388, 392
53023 388, 392
53024 388, 392
53025 388, 392
53026 388, 392
53027 388, 392
53028 389, 392
53029 389, 392
53030 389, 392
53031 389, 392
53032 389, 392
53034 389, 392
53042 389, 392
53043 392
53044 392
53052 389, 392
53053 389, 392
53056 384, 392
53059 389, 392
53060 392
53062 389, 392
53067 389, 392
54001 392
54002 393
54003 386, 393
54004 393
54005 393
54006 393
54008 393
54011 393
54012 393
54013 393
54014 393
54015 393
54016 393

SQLSTATE *(continued)*
54017 393
54018 393
54019 393
54020 393
55002 393
55008 389, 393
55013 (now 42889) 393
55021 393
55023 393
55028 393
55044 393
56001 390, 393
56002 390, 393
56003 390, 393
56004 393
56005 390, 393
56006 390, 393
56007 390, 394
56008 390, 394
56009 390, 394
56011 390, 394
56012 390, 394
56013 390, 394
56014 390, 394
56015 390, 394
56017 390, 394
56026 384, 394
56031 394
56032 390, 394
56033 394
56034 394
56035 394
56037 390, 394
56038 394
56039 390, 394
56041 394
56042 394
56043 391, 394
56044 394
56046 394
56047 394
56048 394
56049 394
56058 394
56061 391, 394
56069 391, 394
56076 395
56079 395
57008 395
57010 395
57011 395
57012 395
57013 395
57014 395
57016 395
57017 395
57022 395
57024 395
57025 395
57026 395
57027 395
57028 395
57029 395
57030 395
57031 395
57033 396
SQLSTATE *(continued)*
57037 396
57038 396
57040 396
57044 396
57045 396
58002 396
58003 396
58004 396
58005 396
58009 396
58010 396
58011 396
58012 396
58014 396
58015 396
58016 396
58017 396
58018 396
58019 382, 396
58021 396
58022 385, 396
58023 396
58024 396
58025 396
58027 397
58029 397
58030 397
introduction 377
statement
INSERT 296, 352
SELECT 296
UPDATE 296, 352
system control messages 3
system errors 421

T

terminology vii
trace formatter messages 263
transaction abend, VM 269

U

using manual 1

V

VM abend code
11 269
13 269
15 269
17 269
18 269
19 269
21 269
3 269
4 269
5 269
7 269
9 269
free storage 269
WRTERM/LINEWRT 269
VM APPC/VM communication
codes 400
VM system services return codes 399

Readers' Comments — We'd Like to Hear from You

DB2 Server for VM
Messages and Codes
Version 7 Release 1

Publication No. GC09-2984-00

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

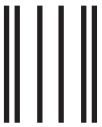
Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
GC09-2984-00

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM CANADA LTD.
DB2 Server for VSE & VM
2S/240/1150/TOR
1150 Eglinton Avenue East
North York, Ontario, Canada M3C 1H7

Fold and Tape | **Please do not staple** | Fold and Tape

GC09-2984-00

Cut or Fold
Along Line

File Number: S370/4300-50
Program Number: 5697-F42

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GC09-2984-00

Spine information:

DB2 Server for VM Messages and Codes Version 7 Release 1