DB2 DataJoiner

IBM

Messages and Problem Determination Guide

Version 2 Release 1

DB2 DataJoiner

Messages and Problem Determination Guide

Version 2 Release 1

IBM

First Edition (October 1997)

This edition applies to Version 2 of IBM DB2 DataJoiner, 5765-C36 and Version 1.2 of DataJoiner Classic Connect for MVS, 5697-DJM, and to any subsequent releases until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address below.

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

IBM Corporation, W92/H3
P.O. Box 49023
San Jose, CA 95161-9023
U.S.A.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. 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 of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter 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 the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594, U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling: (1) the exchange of information between independently created programs and other programs (including this one) and (2) the mutual use of the information which has been exchanged, should contact IBM Corporation, Department J01, 555 Bailey Avenue, San Jose, CA 95161-9023. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

Advanced Peer-to-Peer Networking (APPN)	Extended Services for OS/2
Advanced Program-to-Program	IBM
Communication (APPC)	IIN
AIX	IMS
AIXwindow	MVS
AIX/6000	IMS/ESA
APPN	MVS/ESA
AS/400	MVS/XA
Customer Information Control System (CICS)	Operating System/2
CICS/6000	Operating System/400
DATABASE 2	OS/2
DataGuide	OS/390
DataJoiner	OS/400
DataPropagator	RACF
DB2	RS/6000
DB2/2	SQL/DS
DB2/400	SQL/400
DB2/6000	System/390
DFSMS	System Modification Program Extended (SMP/E)
Distributed Relational Database Architecture	VisualAge
DRDA	VisualGen
	Virtual Telecommunications
	Access Method (VTAM)

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Microsoft, Windows, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

Intel and Pentium are trademarks of the Intel Corporation.

HP-UX is a registered trademark of Hewlett-Packard.

Solaris is a registered trademark of Sun Microsystems, Inc.

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

Cross-Platform Terminology Conventions

Some product names in the documentation refer to more than one product; other names refer to specific product levels. This list contains the most frequently used product definitions.

DataJoiner Refers to DB2 DataJoiner Version 2. References specific to or including DataJoiner Version 1 will include the version.

DB2 Family Refers to all DataJoiner- supported versions of DATABASE 2 (DB2) database server products on all platforms (DB2 for OS/390, DB2 for VM, DB2 for common servers, DataJoiner, and so on). Supported versions are

listed in the *DataJoiner Planning, Installation, and Configuration Guide* for your platform.

DB2   By itself, refers to any one or all of the DB2 for common server Version 2 database server products on all platforms, which includes DataJoiner.

If a DB2 reference is qualified with a specific operating system or version, the reference applies only to that particular version.

DB2 for CS Refers to any DB2 for common servers Version 2 database server product. This term is often used when describing DataJoiner and DB2 for common servers functional differences

About This Library

To understand the organization of the DataJoiner library, it is important to understand the relationship between DataJoiner and DB2 for CS. DataJoiner provides a "superset" of DB2 for CS. The two products share common functions and syntax; therefore, information that is common to DataJoiner and DB2 for CS is documented in the DB2 for CS books. The DataJoiner books listed in Table 1 document the function and syntax that DataJoiner has *in addition to* the function and syntax it shares with DB2 for CS.

DataJoiner, DB2 for CS, and Replication Library Publications

Table 1 lists the DataJoiner, DB2 for CS, and Replication manuals applicable to installing, configuring, administrating, using, and running applications against DataJoiner. The *DataJoiner for AIX Planning, Installation, and Configuration Guide* and the *DataJoiner for Windows NT Systems Planning, Installation, and Configuration Guide* books are provided with DataJoiner. The remaining books are provided in softcopy formats on the product CD-ROM. All listed books are provided in PostScript; most are provided in HTML (the two exceptions are the DB2 for CS Software Developer Kit publications). Additionally, most of the DB2 for CS books are provided in INF format (see Table 1).

Table 1 does not list all of the DB2 for CS books. View or print a DB2 for CS book to see the publications list for all DB2 for CS books.

If you order Classic Connect, you will receive additional books (the *DataJoiner Classic Connect Planning, Installation, and Configuration Guide*, the *DataJoiner Classic Connect data mapper Sample for Windows Installing and Using Guide*, and the *DataJoiner Messages and Problem Determination Guide*) and a program directory.

Table 1 (Page 1 of 4). DataJoiner, DB2 for CS, and Replication publications applicable to DataJoiner

Book Name	Form Number	File Prefix	INF
DataJoiner Version 2.1 Books			
DataJoiner for Windows NT Systems Planning, Installation, and Configuration Guide	SC26-9150	DJXN1	no
This book covers capacity planning, resource management, installation, and configuration tasks for IBM DataJoiner on Microsoft Windows NT operating systems.			
DataJoiner for AIX Systems Planning, Installation, and Configuration Guide	SC26–9145	DJXG5	no
This book covers capacity planning, resource management, installation, and configuration tasks for IBM DataJoiner on AIX operating systems.			
DataJoiner Administration Guide	SC26–9146	DJXD4	no

Table 1 (Page 2 of 4). DataJoiner, DB2 for CS, and Replication publications applicable to DataJoiner

Book Name	Form Number	File Prefix	INF
This book provides information that assists DBAs and other system administrators of DataJoiner, IBM's heterogeneous data access product, to perform administrative tasks. It includes a product overview section, security considerations, data source identification steps, database utility notes, performance considerations, database system monitor reference data, large object information and explain tool examples.			
DataJoiner Application Programming and SQL Reference Supplement	SC26–9148	DJXK4	no
This book provides SQL statements, descriptions of system catalog data, guidelines, and other information for application programmers. With this information, application programmers can use DataJoiner to perform multiple tasks in a distributed database environment—tasks such as creating nicknames by which to reference tables and views, invoking functions and stored procedures, passing SQL directly to databases for processing, and using server options to optimize query performance.			
DataJoiner Generic Access API Reference	SC26–9147	DJXM4	no
This book explains how to create a generic access module that allows you to use existing drivers or to create new drivers to gain access to an unlimited set of data sources.			
DataJoiner Classic Connect Planning, Installation, and Configuration Guide	GC26–8869	DJXC4	no
This book provides information on the DataJoiner Classic Connect for MVS product. The audience for this information includes application programmers, database administrators, network administrators, system administrators, and system programmers. The book documents key tasks required to set up Classic Connect in the MVS operating environment: planning your setup, installing components via SMP/E, configuring the kernel, DMSIs, and network communications, managing instances, and creating relational data maps for IMS and VSAM data.			
DataJoiner Classic Connect data mapper Sample for Windows Installing and Using Guide	GC26–8873	DJXZ2	no
This book provides information on the DataJoiner Classic Connect data mapper sample for Windows. The audience for this information includes system programmers, DBAs, or anyone that needs to produce relational maps (USE grammar) for IMS and VSAM data. The book documents key tasks required to set up and use the data mapper in the Windows environment: installing product files, starting the product, and generating USE grammar statements for input to DataJoiner Classic Connect projection utilities.			
DataJoiner Messages and Problem Determination Guide	SC26–9149	DJXP4	no
This book describes the messages and codes issued by DataJoiner and Classic Connect instances. For messages that report errors, the book explains the cause of the errors and recommends corrective actions. The book also provides guidelines on using diagnostic tools to isolate and understand problems.			
DB2 for CS and Replication Books			
DB2 Information and Concepts Guide	SH20–4664	SQLG0	no

Table 1 (Page 3 of 4). DataJoiner, DB2 for CS, and Replication publications applicable to DataJoiner

Book Name	Form Number	File Prefix	INF
Provides product and conceptual information to anyone who needs a comprehensive overview of the DB2 products. It is useful when deciding which DB2 products suit your environment. It also includes a glossary of terms used in the book			
DB2 Administration Guide	S20H-4580	SQLD0	yes
Contains information required to design, implement, and maintain a database to be accessed either locally or in a client/server environment.			
DB2 Database System Monitor Guide and Reference	S20H–4871	SQLF0	yes
Includes a description of how to use the Database System Monitor and a description of all the data elements for which information can be collected.			
DB2 Command Reference	S20H–4645	SQLN0	yes
Provides the reference information needed to use system commands and the DB2 command line processor to execute database administrative functions. Describes the commands that can be entered at an operating system command prompt or in a shell script to access the database manager. Explains how to invoke and use the command line processor, and describes the command line processor options. Provides a description of all the database manager commands.			
DB2 API Reference	S20H–4984	SQLB0	yes
Provides information about the use of application programming interfaces (APIs) to execute database administrative functions. Presents a description of APIs and the data structures used when calling APIs, as well as detailed information on the use of database manager API calls in applications written in the supported programming languages.			
DB2 SQL Reference	S20H–4665	SQLS0	yes
Is intended to serve as a reference for syntax and rules governing the use of SQL statements. Syntax diagrams, semantic descriptions, rules and examples are provided for the SQL statements. Catalog views, product maximums, release-to-release incompatibilities, and a glossary are also included in this book.			
DB2 Application Programming Guide	S20H–4643	SQLA0	yes
Discusses the application development process and how to code, compile, and execute application programs that use embedded SQL to access the database. It includes discussions on programming techniques and performance considerations for the application programmer.			
DB2 Call Level Interface Guide and Reference	S20H–4644	SQLL0	yes
Is a guide and reference manual for programmers using the Call Level Interface. DB2 Call Level Interface is a callable SQL interface based on the X/Open** CLI specification and is compatible with Microsoft** Corporation's ODBC.			
DB2 Messages Reference	S20H–4808	SQLM0	yes
Lists messages and explanations. Each explanation includes the action to be taken when a message or code is issued.			

Table 1 (Page 4 of 4). DataJoiner, DB2 for CS, and Replication publications applicable to DataJoiner

Book Name	Form Number	File Prefix	INF
DB2 Problem Determination Guide	S20H–4779	SQLP0	yes

Provides information that helps in determining the source of errors, recovering from problems, and describing and reporting defects.

DDCS User's Guide

Internet Newsgroups DataJoiner questions, answers, and discussions can be found in:

- bit.listserv.db2-l
- comp.databases
- comp.databases.ibm-db2

About This Book

This book will help you interpret messages and codes issued by DataJoiner (including Classic Connect) and then identify corrective actions. This preface:

- Explains who should use this book

- Identifies conventions

- Identifies DataJoiner publications

Who Should Read This Book

This book helps you understand the error messages issued by DataJoiner. It contains information for Application Programmers, Database Administrators, System Administrators, and System Programmers. Any person responsible for DataJoiner technical support will use this book.

Format Conventions

The format for date and time, and the characters used as separators for date and time, are intended for a system configured to use the United States national language format. Your date and time display might be different, depending on the national language format used by your system.

What's New in DataJoiner Version 2?

DataJoiner Version 2 offers many new features and enhancements. This chapter describes some of the major changes for this version, and points you to sources of more information in the DataJoiner and DB2 libraries. Major enhancements include:

DB2 Version 2 functionality

DataJoiner is built on the DB2 Version 2 code base, which means that DataJoiner provides all the major functional enhancements provided by DB2, including:

- Extended SQL capabilities
- An enhanced SQL optimizer
- Improved database performance
- Systems management support
- Robust integrity and data protection
- Object relational capabilities
- National language support (NLS)

See the *DB2 Administration Guide* for detailed information about these features.

DataJoiner for Windows NT

DataJoiner has extended its reach to provide an industrial strength heterogeneous database management system to the Windows NT platform. DataJoiner for Windows NT supports the same SQL and features as DataJoiner for UNIX-based platforms.

Expanded DataJoiner SQL support

This version of DataJoiner contains many new and modified SQL statements. New DDL statements provide greater flexibility and safety in defining your DataJoiner environment—users can create, alter, and drop mappings for data sources, users, user-defined and built-in functions, and data types. Additionally, new SQL DML statements provide enhanced functions for local and distributed queries; an example is the CASE expression, which is useful for selecting an expression based on the evaluation of one or more conditions.

Distributed heterogeneous update support

DataJoiner now allows you to update multiple heterogeneous data sources within a distributed unit of work while maintaining transaction atomicity. This task is accomplished through adherence to the two-phase commit model. Supported data sources include most versions of the DB2 Family and, with the appropriate XA libraries, various other data sources as well.

New graphical installation, configuration, and administration tools

A variety of new tools is available to help you accomplish most administrative chores. TaskGuides walk you through common tasks, such as configuring communications and data source access. The Administrator's Toolkit provides a collection of tools designed to assist you with the day-to-day operation of DataJoiner. It consists of the following components:

The Command Line Processor A system command prompt used to access and manipulate databases.

The Database Director Allows you to perform configuration, backup and recovery, directory management, and media management tasks.

Visual Explain A tool for graphically viewing and navigating complex SQL access plans.

The DB2 Performance Monitor Monitors the performance of your DB2 system for tuning purposes.

Stored procedures

DataJoiner now supports stored procedures at remote data sources as well as the local DataJoiner database. Use stored procedures to speed application performance. For example, applications that process huge amounts of data at a server but return smaller result sets should run faster as stored procedures. Another benefit is that stored procedures usually reduce network traffic between clients and databases.

DataJoiner stored procedures can augment standard data security. For example, in a 3-tier environment, data can be retrieved from a remote server and then processed at the DataJoiner server; only a subset of data needs to available to the client.

Heterogeneous data replication

DataJoiner now provides replication administration as an integrated component. You can define, automate, and manage replication data from a single control point across your enterprise. A GUI provides administrative support for the replication environment, with objects and actions that define and manage source and target table definitions. DataJoiner's Apply component performs the actual replication, tailoring and enhancing data as you specify, and serving as the interface point to and from your various data sources.

System catalog information available in views

DataJoiner provides views from which you can access system catalog information about each DataJoiner database. Some of these views contain data—for example, data about tables, indexes, and servers—that was accessible only from tables in previous versions of DataJoiner. Other views contain data—for example, data about stored procedures, server options, and server functions—that is now available in Version 2.

RDB Support

With Version 2, DataJoiner continues to increase the number of natively supported data sources. Oracle RDB is the most recent addition.

Performance Enhancements

In addition to general engine performance improvements, this latest version offers new query rewrite capabilities, improved pushdown performance, and remote query caching.

Section 1. DataJoiner Instance Diagnostic Tools and Messages: Introduction

Attention: For information about problems not documented in this book, see the *DB2 Problem Determination Guide for common servers* (S20H–4779).

Chapter 1. Using Diagnostic Tools

Use the diagnostic tools described in this chapter to gather and process data that can help identify the cause of a problem.

Once collected, the data can be examined by you or by someone who is familiar with the problem, or provided to IBM for analysis.

When to Use a Diagnostic Tool and Why

These tools are designed to be used as part of the overall problem determination process and should be used only as specified by one of the following:

- The messages in this book
- An authorized IBM service representative

Only procedural information is provided in this chapter; you should not attempt to determine when it is appropriate to use one or more diagnostic tools based solely upon this information.

Brief descriptions of each diagnostic tool follow. Directions for using each are provided later in this chapter.

Notes:

1. This guide focuses on the DataJoiner problem determination perspective. For data source or communication software problems, refer to the documentation provided with that product.

2. Information should be sent to IBM for analysis only as recommended by an IBM service representative.

3. Do not send IBM information that is considered to be confidential or proprietary by you or your end user.

Types of Diagnostic Tools

Identifying the source of most software problems begins with examining messages displayed on the user's screen or saved by one of the message-logging utilities described on the following pages. If this information does not help you resolve the problem, using a trace or dump facility may be appropriate.

However, there are exceptions. For this reason, you should rely on the information provided in the descriptions of messages in this book, or recommendations by an IBM service representative, on how to proceed for a given problem.

First Failure Data Capture

First Failure Data Capture (FFDC) is a facility that obtains and conveys information produced at the time that a failure occurs. For more information, see "Using First Failure Data Capture" on page 3.

The Independent Trace Facility

The Independent Trace Facility is a command-line interface used to record or trace the sequence of events related to a DataJoiner API. Procedures for using this facility are provided in "Using the Independent Trace Facility" on page 8.

The Logging Utilities

- System Log (syslog)

  DataJoiner uses the system logger (syslog) to log error and warning conditions.

- Error Log

  The error logging facility is used to provide a record of error information for problem determination for DataJoiner. It also provides a reporting function to display or print the contents of the log file.

Process Status Utility

The **ps** command is a system command which returns process status information about active processes to standard output.

The Core File

A core file is created in the current directory when severe errors occur. It contains a memory image of the terminated process, and can be used to determine what function caused the error.

Using First Failure Data Capture

This section discusses the configuration parameters that specify the location and level of First Failure Data Capture (FFDC) information. The section then describes the log, alerts, and dump files that contain the information.

Configuration Parameters

The Database Manager configuration file contains two parameters for FFDC information:

- DIAGPATH specifies the fully qualified path for diagnostic information for Version 2 of DB2. In the AIX environment:
  - The default for DIAGPATH is " " (null string).
  - If a diagnostic directory is not specified by DIAGPATH, the FFDC information will be located in INSTHOME\sqllib\db2dump. (INSTHOME is the home directory of the instance owner.)

- DIAGLEVEL specifies the level of diagnostic information that you will receive. The possible values for DIAGLEVEL are:
  - DIAGLEVEL (0) — NO diagnostic data captured
  - DIAGLEVEL (1) — SEVERE errors only
  - DIAGLEVEL (2) — SEVERE and NON-SEVERE errors
  - DIAGLEVEL (3) — SEVERE errors, NON-SEVERE errors, and WARNINGS (the default)
  - DIAGLEVEL (4) —SEVERE errors, NON-SEVERE errors, WARNINGS, and INFORMATION messages

  Note: In this text, severe errors, non-severe errors, warnings, and information messages are all referred to as "errors."

  The supplied default for DIAGLEVEL is 3; you can change it to 0, 1, 2, or 4.

The following list shows command line processor examples for changing the amount of diagnostic information and its location:

- To obtain the most diagnostic information, type:

  ```
  DB2 "UPDATE DATABASE MANAGER CONFIGURATION USING DIAGLEVEL 4"
  ```

- To log no diagnostic information, type:

  ```
  DB2 "UPDATE DATABASE MANAGER CONFIGURATION USING DIAGLEVEL 0"
  ```

- To reset the diagnostic path to the default, type:

  ```
  DB2 "UPDATE DATABASE MANAGER CONFIGURATION USING DIAGPATH NULL"
  ```

- To set the diagnostic path to a new directory, type:

  ```
  DB2 "UPDATE DATABASE MANAGER CONFIGURATION USING /tmp/errors"
  ```

For more information on database manager administration commands, see the *DB2 Command Reference* and the *DB2 API Reference*.

Error Logging

The First Failure Service Log is an ASCII file that contains (1) errors that are logged by DB2 for CS and (2) diagnostic information. The file is located in the diagnostic directory (specified by the DIAGPATH parameter) and is named DB2DIAG.LOG.

The DB2DIAG.LOG file contains the following information:

- The location of the error that is being reported

- A diagnostic message (DIA?????) explaining the reason for the error

- Any available supporting data, such as SQLCA data structures and pointers to the location of any extra dump information

Figure 1 on page 5 gives you an idea of the types of information that you will find in DB2DIAG.LOG.

Mon Mar 13 13:52:01 1995
ghutchis pid(134898) process (db2sysc) relation_data_serv   sqlrrdi Probe:0 Database (DATA1)
DIA3715C An incorrect database signature was found.

INSTANCE NAME / TIME OF THE ERROR / FUNCTION NAME / DATABASE ALIAS / Explanation of the error

Mon Mar 13 13:52:01 1995
ghutchis pid(134898) process (db2sysc) relation_data_serv   sqlrrdi Probe:0 Database (DATA1)
DIA9999E A internal return code occurred. Report the following : "0xFFFFE10E".

Data Title :SQLCA pid(134898)

SQLCODE

sqlcaid : SQLCA   sqlcabc: 136   sqlcode: -980   sqlerrml: 0
sqlerrmc:
sqlerrp : sqlrita
sqlerrd : (1) 0xFFFFE10E   (2) 0x00000000   (3) 0x00000000
          (4) 0x00000000   (5) 0x00000000   (6) 0x00000000
sqlwarn : (1)   (2)   (3)   (4)   (5)   (6)
          (7)   (8)   (9)   (10)   (11)
sqlstate:

Extra dump data provided

Dump File : /u/ghutchis/sqllib/db2dump/134898.dmp Data : SQLCA
Dump File : /u/ghutchis/sqllib/db2dump/134898.dmp Data : REQUEST TYPE

Figure 1. Sample DB2DIAG.LOG File

Note that in the DB2DIAG.LOG file in Figure 1 on page 5:

- A timestamp indicates when the error occurred.
- The instance name, function name, and database alias are all listed to assist in pinpointing the error.
- A diagnostic message (DIA?????) explains the error that occurred.

- The appropriate SQLCODE is listed (to get help for an SQLCODE, type `DB2 "?"` `SQLnnnn`, where `nnnn` is the SQLCODE).

- Additional dump information is listed (if it is available) to provide more information about the error.

New information is appended to the bottom of the DB2DIAG.LOG file as it is generated, so you should check the end of the file for the most recent error data.

Note: DB2DIAG.LOG *does not get pruned* automatically. If this file becomes too large, erase it and a new file will be generated by the system.

Alerts

An *alert* is a notification that is issued when a severe error that the database administrator needs to know about occurs. (An alert is likely to be seen first by the network administrator and then brought to the attention of the database administrator.) Alerts are logged in:

- The First Failure Service Log (DB2DIAG.LOG). If the DB2 SNMP subagent is activated, any alert reported in this log will result in a corresponding SNMP trap. For more information about DB2DIAG.LOG, see "Error Logging" on page 4. For a list of alert numbers and IDs in DB2DIAG.LOG, see Table 2.

- The Alert Log (DB2ALERT.LOG). It contains the following information:
    - The location of an alertable error condition
    - A diagnostic message (DIA?????) explaining the reason for the error

When an alert is issued in the AIX environment, an error record is written to the system error log (syslog). For information about syslog, see "The System Error Log" on page 15.

Table 2 shows the alert numbers that can appear in DB2DIAG.LOG, and their corresponding alert IDs.

Table 2 (Page 1 of 2). Representation of Alert IDs in DB2DIAG.LOG	
Alert Number	**Alert ID**
1	171099F3
2	E339D296
3	FB8BEFD7
4	EF64B29A
5	1188505E
6	7286DE23
7	7286DE23
8	7286DE23
9	7286DE23
10	7286DE23

Table 2 (Page 2 of 2). Representation of Alert IDs in DB2DIAG.LOG	
Alert Number	**Alert ID**
11	7286DE23
12	7286DE23
13	7286DE23
14	7286DE23
15	7286DE23
16	7286DE23
17	A70F6F93
18	A70F6F93
19	A70F6F93
20	A70F6F93
21	A70F6F93
22	A70F6F93
23	A70F6F93
24	2257C33F
25	C299284E
26	2E0AA333
27	D67E885A
28	DA23E856
29	9A22708B
30	91EC5326
31	3AED0327
32	36B0632B
33	46E34E31

Dump Files

Dump files are created when an error occurs for which there is additional diagnostic information, such as information about internal control blocks. Every data item written to the dump files has a timestamp associated with it to aid in problem determination, and when a dump file is created or appended to, a reference is made in the DB2DIAG.LOG file.

Note: Dump files are in binary and can be examined only by an IBM service representative.

Using the Independent Trace Facility

DataJoiner uses the Independent Trace Facility to trace events, dump trace data to a file, and format trace data.

Preliminary Considerations

Read the information that follows before attempting to use the Independent Trace Facility:

- The Independent Trace Facility should be used only as directed in this book or by an IBM service representative.

- System performance can be affected when the Independent Trace Facility is turned on, therefore, only use this facility when error conditions are occurring.

- Only procedural information is presented in this chapter. Descriptions of particular problems and suggestions as to what event codes to trace are provided by an authorized IBM service representative.

- Minimum system activity is recommended when using this utility.

- Information should be sent to IBM for analysis only upon direction from an IBM service representative. Do not send IBM information that is considered to be confidential or proprietary by you or your end user.

Using Independent Trace Facility Commands

Independent Trace Facility tracing is activated by the **db2trc on** command. The syntax for the **db2trc** command is shown in the following diagram. Considerations for using Independent Trace Facility commands follow.

Syntax Restrictions:

1. In the -m option, *num* is greater than or equal to 0 and less than or equal to 255.

2. In the -m option, the comma (,) may be repeated any number of times for items in a series; however, there can be at most three periods (.) that separate the four byte-mask fields.

Parameters

on   Use this parameter to start the Independent Trace Facility. See "Starting an Independent Trace Facility Trace" for information about options for this parameter.

dump   To reproduce the error and then dump the trace out to a file, in the current directory, type:

```
db2trc dump trace.bin
```

where *trace.bin* is the name of a binary file. You must specify a file name with this parameter. This file is saved in the current directory unless you explicitly include the path.

off   After you dump the trace to a file, turn trace off by typing:

```
db2trc off
```

Starting an Independent Trace Facility Trace

To use the Independent Trace Facility, you must first turn trace on by typing:

```
db2trc on
```

db2trc on has the following options:

-m *mask*   Specifies trace record types. The *mask* variable consists of four byte-masks separated by periods. The byte-masks correspond to

- *mask* = * (trace everything)
- *max_sys_errors* = 1
- *max_record_size* = 16K
- Trace destination = -s (shared memory)
- Records to retain = -l (last)
- *buffer_size* = 64KB if the trace destination is shared memory, or 2MB if the trace destination is a file.

A detailed description of these options follows.

Specifying Trace Record Type

-m *mask*

This option is used to limit the types of trace records that will be recorded in the trace buffer.

DataJoiner sends trace records to the Independent Trace Facility for many different events. Each trace record contains an ID made up of 4 bytes. These bytes represent the product number, the event type, the component number, and the function number respectively.

Because each of these 4 fields is represented by a byte, the Independent Trace Facilitycan trace up to 256 products, each with up to 256 event types. Each event type can have up to 256 components, and each component can have up to 256 functions. An ID is represented by 4 numbers, each in the range 0 to 255, separated by periods. For example, 4.7.47.53 is the ID of a trace record generated by product 4, event type 7, component 47, and function 53.

Like an ID, a mask is also made up of four fields, but the fields now represent sets instead of values. For example, 1.0-20.2-5,15.* is a mask that represents product 1, event types 0 through 20, components 2 through 5 and 15, and all functions. The asterisk (*) indicates 0 through 255.

The mask acts as a filter to accept or reject each trace record on the basis of its ID. A trace record is captured only if the mask accepts it. A mask accepts an ID only if each of the four fields of the ID belong to the set represented by the corresponding fields in the mask.

A mask is composed of up to four byte-masks separated by periods.

- The *first byte-mask* describes the set of products to trace.

- The *second byte-mask* describes the set of event types that you want to trace. The following event types are defined:

0	system_error
1	system_error
2	system_error
3	non-fatal_error
4	non-fatal_error

5	api_errcode
6	cei_errcode
7	fnc_errcode
8	trap error
10	api_entry
11	api_exit
13	api_retcode
15	api_data
20	cei_entry
21	cei_exit
23	cei_retcode
25	cei_data
30	fnc_entry
31	fnc_exit
33	fnc_retcode
35	fnc_data

A brief description of these event types follows:

system_error

Indicates that a fatal error occurred. A *fatal error* is a software or hardware condition that should not occur during normal operation. In addition, the program being traced does not have code to recover from the error in such a way as to guarantee the correctness of the results.

system_error indicates one of the following conditions:

– Hardware failure
– Insufficient resources
– Code defect
– Database that is damaged and unusable

non-fatal_error

Indicates that a situation occurred that may eventually cause a system_error. The code that detected the non-fatal error is not sure that this condition will materialize into a system error, so it declares it as a non-fatal error.

api_ *xxxxxxx*, **cei_***xxxxxxx*, **fnc_***xxxxxxx*

Several event types have one of these prefixes. They refer to APIs (application programming interfaces) CEIs (component external interfaces) and FNCs (functions). APIs are functions that DataJoiner applications can call, CEIs are functions that other DataJoiner components can call, and FNCs are functions that can only be called from the DataJoiner components to which they belong.

*xxx*_**entry,** *xxx*_**exit,** *xxx*_**retcode,** *xxx*_**errcode**

Several event types have one of these suffixes. When you turn trace on, you use event types with these suffixes to turn trace on for entries to functions, exits to functions, entries to functions with arguments, exits to functions with return codes, or exits to functions with negative return codes, respectively.

You can limit the amount of information in the trace buffer by tracing only *xxx*_entry and *xxx*_exit. If you do this, you will get all entries and exits, but the argument and return_code information will not be traced. Likewise, if you trace only *xxx*_errcodes, you will only get trace records for function exits of functions that trace their return codes and that return a negative code.

The event type numbers have been carefully chosen so that, when you specify a mask, you can use the following ranges for the second byte-mask:

0-2	Trace only system errors.
0-4	Trace only system and non-fatal errors.
0-9	Trace all kinds of errors.
0-11	Trace only errors and APIs with no arguments or return codes.
0-19	Trace only errors and APIs with arguments and return codes.
0-29	Trace only errors, APIs, and CEIs.
0-39	Trace only errors, APIs, CEIs, and FNCs.
0-65	Trace only errors, APIs, CEIs, FNCs, and data structures.

It is recommended to use '*' as the component (or components) to trace on.

- The *third byte-mask* describes the set of components to trace. The following components are currently defined for DataJoiner.

1	(unused)
2	buffer_pool_services
3	dist_data_services
4	data_management
5	base_sys_utilities
6	config/install
7	(unused)
8	(unused)
9	index_manager
10	gateway_drda_ar
11	(unused)
12	(unused)
13	database monitor
14	sql_compiler
15	oper_system_services
16	data_protection
17	users/groups st_proc
18	relation_data_serv
19	sort/list_services
20	(unused)
21	database_utilities
22	data_services
23	(unused)
24	(unused)
25	(unused)
26	global_services
27	cobol_precompiler

28	c_precompiler
29	fortran_precompiler
30	aps_general_service
31	aps_precompiler_svc
32	aps_binder_service
33	aps_runtime_services
34	(unused)
35	lock_manager
36	dist_data_services
37	gateway_drda_ar
38	sql_compiler
39	relation_data_serv
40	(unused)
41	command line processor
42	command line interpreter
43	common communication interface
44	base_sys_utilities
45	X/Open API support
46	DBA utility
47	DBA utility
48	Segment Manager
49–71	
	Reserved for compiler
72	LOB Manager
73	rds_interpreter
74	operating_system_services
75	DRDA AS
76	base_support_utilities
77	database_utilities
78	buffer_pool_services
79	directory_services
87	mid
88	mid2
89	db2ra
90	Informix
91	MS_SQL
92	Generic

- The *fourth byte-mask* determines the set of functions to trace. Trace all functions unless directed to do otherwise by an IBM service representative.

Each byte-mask can be either an asterisk (*), which represents 0 through 255, or a series of byte ranges separated by commas. A byte range can be either a number (from 0 to 255) or two numbers separated by a dash.

Several mask examples follow:

```
0-3,10
```

Because only the first byte-mask was specified, the last three byte-masks default to asterisk (*). Consequently this mask accepts all trace records from products 0, 1, 2, 3, and 10.

```
1.*.10-20,30-40.80-100
```

This mask accepts trace records that come from product 1 (which represents DataJoiner), any event type, components 10 through 20 and 30 through 40, and functions 80 through 100.

```
1,2.0-9,40-41.5,15,18
```

This mask accepts trace records for products 1 and 2; event types 0 through 9 and 40 through 41 (all error and miscellaneous event types); components 5, 15, and 18. Since the fourth byte-mask was omitted, it defaults to asterisk (*). Consequently, trace records are collected for all functions within those three components.

```
*
```

Specifying a mask that contains only an asterisk (*) causes trace records to be accepted for all products (0 to 255), all event types, all components, and all functions. **db2trc on** uses the asterisk (*) as the default mask if the -m option is not specified.

Limiting the Number of Non-System Errors the Trace Will Hold

-e *max_sys_errors*

This option specifies the maximum number of system errors that can occur before the Independent Trace Facility begins rejecting trace records with non-system-error event types. Once this maximum is reached, only system-error-related events will be captured regardless of what you have specified in the second byte-mask.

Limiting the Size of Trace Records

-r *max_record_size*

This option is used to limit the size of trace records. This option will allow you to hold more trace records in the trace buffer. The *max_record_size* parameter must be a number between 0 and 16384. Trace records that would have exceeded 4+*max_record_size* bytes are truncated to 4+*max_record_size* bytes. The 4 bytes that are always kept are called the ID and represent the product number, event type, component, and function. The default setting includes the entire record.

Specifying Trace Buffer Location

-s|-f *filename*

Only one of these two options can be used. Use -s if you want the trace buffer to reside in shared memory and -f if you want it to reside in a file.

If you use -f, the *filename* parameter should not exceed 255 characters. The file created has exactly the same format as the ones created by the **db2trc dump** command. This option generates overhead because it causes a file system access

every time a trace record is generated. This option is meant to be used mainly to ensure that you get at least a binary trace file in cases where the error causes the system to hang or fail.

Specifying Trace Buffer Size and Wrapping

-l|-i [*buffer_size*]

Only one of these two options can be used. The -l option is the default. It causes the trace buffer to wrap when full, storing new data at the start of the buffer; old data is overwritten. '-i' specifies the retention of the initial trace records (no more records are written to the buffer once it is full).

The *buffer_size* parameter, which determines the size in bytes of the trace buffer, can be used with either option; however, it is not mandatory. The buffer size is specified in bytes and is rounded down to a multiple of 64KB. It may be as large as the swap space or as the file system permits. You can specify the *buffer_size* value in hexadecimal format (preceded by 0x), in octal format (preceded by 0), or in decimal format (first digit must be 1 to 9). The default for the *buffer_size* parameter is 64KB.

The System Error Log

DataJoiner uses the system logger (syslog) to log error and warning conditions. Entries are added to the syslog based on (1) priority and (2) what facility caused the error or warning condition. For example, security, the kernel, and system daemons can cause entries to be logged in the syslog. DataJoiner is represented by the facility called 'user'. Priority refers to the urgency of the message. The levels of priority, from highest to lowest, are:

- Emergency
- Alert
- Critical
- Error
- Warning
- Notice
- Information
- Debug

DataJoiner uses only one priority:

Priority   *Description*

Alert   Very severe situation; for example, agents killed or possible corrupt databases. These situations should be investigated.

Setting Up the syslog

Follow these steps to set up the syslog.

Step 1

Each machine can be configured to log various combinations of facilities and priorities. To begin logging DataJoiner error and warning conditions in the syslog, you must start logging for the facility of type 'user'. This is accomplished by editing the /etc/syslog.conf file. For example, add the following line to this file:

```
user.alert     fully_qualified_file_name
```

Note: You must have root authority to do this. Also, if this file grows quickly, you may have to reduce its size periodically.

The 'user' portion of 'user.alert' indicates that the facility to log is 'user'. This includes DataJoiner and any other applications logging messages as a 'user' facility.

The 'alert' portion indicates that messages with a priority of alert will be logged.

Step 2

The 'fully_qualified_file_name' is the file with a fully qualified path where the messages will be logged, and the SQLCA will be dumped. This file will not be created by the system. Consequently, you must create this file (in the path of the 'fully_qualified_file_name').

Note: It is okay if this file is empty.

Step 3

The syslog daemon must be sent a signal of '1' to extract the latest information from the /etc/syslog.conf file. This can be accomplished by entering the following command:

```
kill -1 pid_of_syslogd
```

You must have root authority to do this.

Note: 'pid_of_syslogd' is the process id of the 'syslogd' process. This can be obtained by issuing the **ps -ef | grep syslogd** command.

Using the syslog

Any syslog messages generated by DataJoinerappear in the file specified in /etc/syslog.conf.

The first line in the system log for DataJoiner entries will consist of the following information:

- Date
- Time
- Hostname
- 'DB2' identifier
- Process ID of reporting process

- Instance name

- Reporting component

- Reporting function

- probeid

- error_num

- alert_num

Note: The **ps** command is useful to gather more information about the process id of the reporting process. For example, you can execute the following command from an operating system command prompt to get information on the reporting process:

```
ps -ef | grep pid
```

The first line of a DataJoiner entry in the system log will have the following format:

```
date time hostname DB2[pid]: DB2 (instance_name) component_name
    function_name reports: probeid with error_num and alert_num
```

Subsequent lines may contain the following information or a subset of this information:

Information   Description

Symptom string

    Contains further information about where and why the problem occurred.

Hexdump of data

    Return code, small structure, or character string of information.

Information about additional dump files

    Larger structures and other binary data may be dumped to files under the directory /u/$DB2INSTANCE/sqllib/db2dump. The name of the file will be included in the syslog along with an eyecatcher to identify each piece of dump data.

An example of an entry in the system log is shown below:

```
Jun 18 15:02:53 bluejay DB2[46827]: DB2 (db2inst1) oper_system_services sqlobeep reports .
: probe id 1 with error  .
2055 and alert num 0
Jun 18 15:02:53 bluejay DB2[46827]:   extra symptom string provided: RIDS/sqlesysc_ MS/ABNOR_TERRM PRCS/8
Jun 18 15:02:53 bluejay DB2[46827]:    data:   54686973  20697320  616e2065  78616d70         This is an examp
Jun 18 15:02:53 bluejay DB2[46827]:    data:   6c65206f  66206c6f  67676564  20646174         le of logged dat
Jun 18 15:02:53 bluejay DB2[46827]:    data:   61                                              a
Jun 18 15:02:53 bluejay DB2[46827]:   2 piece(s) of dump data provided and dumped to file  .
/u/db2inst1/sqllib/db2dump/db263806.dmp
Jun 18 15:02:53 bluejay DB2[46827]:        1. 'DUMP EXAMPLE #1' has been dumped
Jun 18 15:02:53 bluejay DB2[46827]:        2. 'DUMP EXAMPLE #2' has been dumped
```

Note: System log entries from the database manager are in English only.

Using the Core File

A core file is created in the current directory when a program terminates abnormally. Errors such as memory address violations, illegal instructions, bus errors, and user-generated quit signals commonly cause a core dump. The core file that is created contains a memory image of the terminated process. The name of the core file in the current directory is 'core'.

If a core file is created, you can use a debugging program, such as the **dbx** system command on AIX and the **xdb** system command on HP-UX, to determine what function caused the problem. This is a simple check that will help you identify whether DataJoiner is in error, or whether an operating system or application error is responsible for the problem.

This section shows an example that uses the **dbx** command. If you are operating in an HP-UX environment, substitute **xdb** for each use of **dbx**.

Note: The **dbx** command has much more function than is described in this section. To access this additional information, enter **man dbx** from an AIX command prompt.

To determine the function which caused the core file dump to occur, enter the following command from a command prompt:

```
dbx program_name core_filename
```

 where *program_name* is the name of the program that terminated abnormally, and *core_filename* is the name of the file that contains the core file dump. The *core_filename* parameter is optional. If you do not specify it, the default name of 'core' is used.

Note: To obtain symbolic information, compile the application using the

```
-g
```

option.

The following example shows how to use the **dbx** command to read in the core file for a program called 'main'. At a command prompt, enter:

```
dbx main
```

 Output similar to the following will appear on your display:

```
dbx version 3.1 for AIX.
Type 'help' for help.
reading symbolic information ...
[using memory image in core]

segmentation.violation in freeSegments at line 136
136          (void) shmdt((void *) pcAddress[i]);
(dbx) where
freeSegments(numSegs = 2, iSetId = 0x2ff7f730, pcAddress = 0x2ff7f758, line
136
in "main.c"
main (0x1, 2ff7f7d4), line 96 in "main.c"
```

The **dbx** command displays the name of the function that caused the core dump. In the above example, the function name is freeSegments. If the function name begins with 'db2', 'sql', or 'ddcs', this indicates that the error likely occurred in DataJoiner; otherwise, the operating system or another application caused the core file to be created.

The example also shows the use of the **where** subcommand. Enter **where**, at the dbx prompt, to display the program path to the point of failure. In the example, the error occurred at line 136 of freeSegments which was called from line 96 in "main.c".

Note: In an HP-UX environment, the subcommand is **t**. If you are in an HP-UX environment, simply substitute **t** for each use of **where**.

Finally, to end the **dbx** command, type **quit** at the dbx prompt.

The ps Command

The **ps** command is an operating system command which returns process status information about active processes to standard output. Without flags, the **ps** command displays information about the current workstation. The flags control the types of information displayed for each active process, and the flags may be applied simultaneously yielding a cumulative effect.

The following command is commonly used:

```
ps -aef | grep db2 | grep -v grep | more
```

It lists all the active database engine (instance) processes on the system.

Note: For a complete description of the **ps** command, enter the **man ps** command from a system command prompt.

Chapter 2. Introduction to Messages

This book assumes that you are familiar with the functions of DataJoiner. You can use the information in the following chapters to identify and resolve an error or problem. You can use the information also to understand where messages are generated and logged.

Information Available Online

The messages accessible from the operating system command prompt with DataJoiner are DB2/6000 messages. The messages accessible online are identical to the messages in this book. The following message types are accessible from the operating system command line:

Prefix	Description
CLI	Call Level Interface messages
DBA	Error messages generated by the database director and the database administration utility
DBI	Error messages generated by installation and configuration processes
DB2	Error messages generated by the command line processor
SQL	Error messages generated by the database manager when an error condition has been detected
SPM	Error messages generated by the LU 6.2 Sync Point Manager

Message identifiers consist of a three character message prefix (CLI, DBA, DBI, DB2, SQL, or SPM), followed by a four or five digit message number, and terminated by a single letter which describes the severity of the error message.

To access help on these error messages, enter the following command at the operating system command prompt:

```
db2 ? XXXnnnnn
```

where XXX represents the message prefix and where nnnnn represents the message number.

Note: The message identifier accepted as a parameter of the **db2** command is not case sensitive, and the terminating letter is not required.

Therefore, the following commands will produce the same result:

* db2 ? SQL0000N
* db2 ? sql0000
* db2 ? SQL0000n

If the message text is too long for your screen, AIX users can pipe message results to pg:

```
db2 ? XXXnnnnn │ pg
```

Help can also be invoked in the interactive input mode. To enter the interactive input mode, enter the following at the operating system command prompt:

```
db2
```

Once in the interactive input mode, you can enter commands at the following command prompt:

```
db2 =>
```

To get DataJoiner message help in this mode, type the following command after the prompt:

```
? XXXnnnnn
```

Note: If the message text exceeds the length of the screen, users with non-graphical workstations should invoke the command from a command prompt where you can pipe the output to the 'pg' program.

Other Message Sources

When running other programs on the system, you can receive messages with prefixes other than those mentioned in this book.

For information on these messages, see the publications that you received for that program product.

Reason Codes

Reason Codes are internal database manager return codes. They can appear when you receive SQLCODE values 902 or 980. Record the codes if you will request IBM Technical Support for a problem that involves SQLCODE values 902 or 980.

Section 2. DataJoiner Instance Messages

Chapter 3. CLI Messages

CLI0001W

Disconnect error.

Cause: An error occurred during the disconnect. However, the disconnect succeeded.

Action: Validate that the communication between the client and the server are still active.

CLI0002W

Data truncated.

Cause: An output buffer specified is not large enough to contain the data.

Action: Increase the size of the output buffer.

CLI0003W

Privilege not revoked.

Cause: The statement was a REVOKE statement and the user did not have the specified privilege.

Action: No action needed.

CLI0004W

Invalid connection string attribute.

Cause: An invalid or unsupported connection string attribute was specified in the connection string but the driver was able to connect to the data source anyway.

Action: No action needed.

CLI0005W

Option value changed.

Cause: The driver did not support the specified option value and substituted a similar value.

Action: No action needed.

CLI0006W

SQLCancel treated like a close.

Cause: The SQLCancel call was treated like a SQLFreeStmt call with the SQL_CLOSE option.

Action: No action needed.

CLI0100E

Wrong number of parameters.

Cause: The number of parameters specified in SQLSetParam or SQLBindParameter was less than number of parameters in the SQL statement.

Action: Respecify the SQL statement or provide more input parameters through SQLSetParam or SQLBindParameter.

CLI0101E

The statement did not return a result set.

Cause: The previous statement does not result in a result set.

Action: Respecify the SQL statement.

CLI0102E

Invalid conversion.

Cause: The conversion between the application data type and SQL data type is not supported by the driver.

Action: Respecify a data conversion which is supported by the driver.

CLI0103E

Too many columns.

Cause: The number of columns specified in SQLBindCol is greater than the number of columns in the current result set.

Action: Respecify the SQL statement or reset the bound column information in SQLBindCol.

CLI0104E

Unable to connect to data source.

Cause: The driver was unable to establish a connection with the data source.

Action: Ensure the server is started and that the communications between the client and server are correct.

© Copyright IBM Corp. 1995, 1997

CLI0105E

Connection in use.

Cause: The specified connection handle is already being used and the connection is still opened.

Action: Allocate a new connection using SQLAllocConnect and retry the connection or terminate the existing connection.

CLI0106E

Connection is closed.

Cause: The connection specified by the connection handle is no longer active.

Action: Establish a new connection.

CLI0107E

Connection failure during transaction.

Cause: The connection failed during the execution of the function and it cannot be determined whether the COMMIT or ROLLBACK occurred before the failure.

Action: Establish a new connection.

CLI0108E

Communication link failure.

Cause: The connection between the driver and the data source failed during execution of this function.

Action: Establish a new connection.

CLI0109E

String data right truncation.

Cause: The data specified through SQLSetParam or SQLBindParameter is larger than the maximum size allowed for the corresponding use of a parameter marker.

Action: Respecify the parameter using SQLSetParam or SQLBindParameter.

CLI0110E

Invalid output or indicator buffer specified.

Cause: The returned data was NULL but the output or indicator buffer specified was a NULL buffer.

Action: Respecify the output or indicator buffer supplying a non-NULL buffer and retry the operation.

CLI0111E

Numeric value out or range.

Cause: Returning the numeric data would have caused the whole part of the number to be truncated.

SQLPutData was called more than once for a parameter and the input data was not of type character or binary.

Action: Respecify the output bindings either through SQLBindCol or SQLGetData to avoid creating a numeric data truncation.

Do not call SQLPutData for a parameter if the application data type specified for that parameter through SQLSetParam or SQLBindParameter is not SQL_C_CHAR or SQL_C_BINARY.

CLI0112E

Error in assignment.

Cause: The data sent for a parameter or column was incompatible with the data type of the associated table column.

Action: Respecify the output binding through SQLBindCol or SQLGetData or the input binding through SQLSetParam or SQLBindParameter.

CLI0113E

Invalid date format.

Cause: The data sent for a date field was invalid.

Action: Respecify the date data.

CLI0114E

Datetime field overflow.

Cause: The data sent for a date, time, or timestamp parameter, or column, was invalid.

Action: Respecify the date, time, or timestamp data.

CLI0115E CLI0126E

CLI0115E

Invalid cursor state.

Cause: The statement is not positioned on a row.

Action: Position the statement on a row by calling SQLFetch or SQLExtendedFetch and retry the operation.

CLI0116E

Invalid transaction state.

Cause: There was a transaction in progress when SQLDisconnect was called.

Action: Call SQLTransact before calling SQLDisconnect.

CLI0117E

Invalid cursor name.

Cause: An invalid or duplicate cursor name was specified in SQLSetCursorName.

Action: Respecify a valid cursor name through SQLSetCursorName.

CLI0118E

Invalid SQL syntax.

Cause: An invalid or incorrect SQL statement was specified.

Action: Respecify a valid SQL statement.

CLI0119E

Unexpected system failure.

Cause: An unexpected system failure occurred while processing the function.

Action: Restart the application and try again.

CLI0120E

Memory allocation failure.

Cause: The driver was unable to allocate memory required to support execution or completion of the function.

Action: Verify that your system has enough memory to complete the desired operation.

CLI0121E

Invalid column number.

Cause: The value specified in iCol is less than zero, greater than the number of columns in the result set, or greater than the maximum number of columns allowed in a result set.

Action: Respecify a valid value for iCol.

CLI0122E

Program type out of range.

Cause: The value specified for fCType is not valid.

Action: Respecify a valid value for fCType.

CLI0123E

SQL data type out of range.

Cause: The value specified for fSQLType is not valid.

Action: Respecify a valid value for fSQLType.

CLI0124E

Invalid argument value.

Cause: The value specified for an argument was invalid. Possible reasons may be a null pointer, invalid length, invalid option, etc.

Action: Re-examine the argument passed in to the function and determine which argument is invalid.

CLI0125E

Function sequence error.

Cause: This function was called in an invalid sequence.

Action: Correct the sequence in your application and retry the operation.

CLI0126E

Operation invalid at this time.

Cause: The operation the system is trying to perform is invalid at this time.

Action: Correct the sequence of operation and retry the operation.

CLI0127E

Invalid transaction code.

Cause: The transaction option specified in SQLTransact was not SQL_COMMIT or SQL_ROLLBACK.

Action: Specify either SQL_COMMIT or SQL_ROLLBACK and retry the operation.

CLI0128E

Unexpected memory handling error.

Cause: Memory handling error.

Action: An unexpected error occurred in the driver while handling some internal memory buffer. Restart your application.

CLI0129E

No more handles.

Cause: A call was made to SQLAllocEnv, SQLAllocConnect, or SQLAllocStmt but no more handles are available in the driver.

Action: Call SQLFreeEnv, SQLFreeConnect, or SQLFreeStmt to free up some handles which are no longer being used in the application.

CLI0130E

No cursor name available.

Cause: There was no open cursor on the statement and no cursor had been set with SQLSetCursorName.

Action: Specify a cursor name through SQLSetCursorName.

CLI0131E

Invalid string or buffer length.

Cause: The length of the buffer specified is invalid.

Action: Specify a valid buffer length.

CLI0132E

Descriptor type out of range.

Cause: The description type specified is not valid.

Action: Specify a valid descriptor type.

CLI0133E

Option type out of range.

Cause: The option type specified is not valid.

Action: Specify a valid option type.

CLI0134E

Invalid parameter number.

Cause: The number specified for the parameter number is less than zero or greater than the maximum parameter supported by the data source.

Action: Specify a valid parameter number.

CLI0135E

Invalid scale value.

Cause: The scale value specified is invalid.

Action: Specify a valid scale value.

CLI0136E

Function type out of range.

Cause: The function type is invalid.

Action: Specify a valid function type value.

CLI0137E

Information type out of range.

Cause: The information type is invalid.

Action: Specify a valid information type value.

CLI0138E

Column type out of range.

Cause: The column type is invalid.

Action: Specify a valid column type value.

CLI0139E

Scope type out of range.

Cause: The scope type is invalid.

Action: Specify a valid scope type value.

CLI0140E   CLI0153E

CLI0140E

Uniqueness option type out of range.

Cause: The uniqueness option type is invalid.

Action: Specify a valid uniqueness option type value.

CLI0141E

Accuracy option type out of range.

Cause: The accuracy option type is invalid.

Action: Specify a valid accuracy option type value.

CLI0142E

Direction option out of range.

Cause: The direction option is invalid.

Action: Specify a valid direction option value.

CLI0143E

Invalid precision value.

Cause: The precision value is invalid.

Action: Specify a valid precision value.

CLI0144E

Invalid parameter type.

Cause: The parameter type is invalid.

Action: Specify a valid parameter type value.

CLI0145E

Fetch type out of range.

Cause: The fetch type is invalid.

Action: Specify a valid fetch type value.

CLI0146E

Row value out of range.

Cause: The row value is invalid.

Action: Specify a valid row value.

CLI0147E

Concurrency option out of range.

Cause: The concurrency option is invalid.

Action: Specify a valid concurrency option value.

CLI0148E

Invalid cursor position.

Cause: The cursor position is invalid.

Action: Specify a valid cursor position value.

CLI0149E

Invalid driver completion.

Cause: The driver completion is invalid.

Action: Specify a valid driver completion value.

CLI0150E

Driver not capable.

Cause: The operation is valid but not supported by either the driver or the data source.

Action: Specify a valid operation.

CLI0151E

No data pending.

Cause: SQLParamData or SQLPutData was called but there is no data at execute pending on this statement.

Action: Respecify the parameters through SQLSetParam or SQLBindParameter.

CLI0152E

Not a string value.

Cause: The function expected a string argument to be supplied.

Action: Respecify the argument to the function.

CLI0153E

Invalid file name length.

Cause: The length specified for the file name is invalid.

Action: Specify a valid file name length.

CLI0154E

Invalid connection state.

Cause: An attempt to change the connection type was made but the connection or another connection (in the case of Distributed Unit of Work) is already opened.

Action: Do not attempt to change the connection type once the connection is opened.

CLI0155E

File name length is too long.

Cause: The length supplied for the file name is greater than supported.

Action: Specify a valid file name length.

CLI0156E

Error closing a file.

Cause: An unexpected condition occurred while closing a file.

Action: Retry the operation.

CLI0157E

Error opening a file.

Cause: An unexpected condition occurred while opening a file.

Action: Retry the operation.

CLI0158E

Error writing to a file.

Cause: An unexpected condition occurred while writing to a file.

Action: Retry the operation.

CLI0159E

Error deleting a file.

Cause: An unexpected condition occurred while deleting a file.

Action: Retry the operation.

CLI0164E

Nullable type out of range.

Cause: An invalid nullable value was specified.

Action: Respecify the nullable value.

CLI0165E

Error in row.

Cause: An error occurred while fetching one or more rows. (Function returns SQL_SUCCESS_WITH_INFO.)

Action: Retry the operation.

CLI0166E

PARMLIST syntax error.

Cause: The PARMLIST value in the stored procedures catalog table contains a syntax error.

Action: Respecify the row for this stored procedure.

CLI0167E

Operation was cancelled.

Cause: SQLCancel was called on the statement.

Action: Resubmit the operation.

CLI0171E

The database was unable to connect. Reason:

Cause: The database connection failed. The 'Reason:' will explain why the database was unable to connect.

Action: Fix the problem, and try the connect again.

CLI0172E

The database was unable to connect. Would you like to continue with the configuration? Reason:

Cause: The database connection failed. The 'Reason:' will explain why the database was unable to connect.

Action: Fix the problem, and try the connect again or continue with the configuration.

CLI0175E   CLI0603E

CLI0175E

No ODBC Driver Manager has been found. ODBCINST.INI file cannot be opened.

Cause: Neither one of the Visigenic or the Intersolv ODBC Driver Managers have been installed. An ODBC Driver Manager must be present before the DB2 ODBC Driver can be registered into the associated ODBCINST.INI file.

Action: Install an ODBC Driver Manager, and try running this command again.

CLI0176E

Unable to write to the ODBCINST.INI file.

Cause: An error has occurred while writing to an ODBCINST.INI file to be used with an ODBC Driver Manager. This may be due to invalid format of the original ODBCINST.INI file.

Action: Rename the existing ODBCINST.INI file to something else and try running this command again. If this error persists, contact your technical support.

CLI0177E

Cannot determine where the DB2 OS/2 Client is installed.

Cause: An error has occurred while trying to locate where the DB2 OS/2 client is installed.

Action: Check that the DB2 OS/2 Client has been correctly installed.

CLI0178I

The DB2 ODBC Driver has been successfully registered.

Cause: The DB2 ODBC Driver has been registered with the installed ODBC Driver Manager(s).

Action: The user can now use the appropriate ODBC Administrator tool from the ODBC Driver Manager vendor to configure the ODBC data sources.

CLI0179E

Unable to display dialog box.

Cause: The DB2 ODBC Driver cannot open a dialog box with the window handle provided by the application.

Action: The application must pass a valid window handle.

CLI0600E

Invalid connection handle or connection is closed.

Cause: Connection had been closed prior to the operation.

Action: Ensure connection close is not called prior to the operation.

CLI0601E

Invalid statement handle or statement is closed.

Cause: Statement had been closed prior to the operation.

Action: Ensure statement close and statement connection close are not called prior to the operation.

CLI0602E

Memory allocation error on server.

Cause: Cannot allocate memory on server.

Action: Ask your database administrator to check the DB2 JDBC log file for details. Rerun your program.

CLI0603E

CallableStatement get*** method was called without registerOutParameter.

Cause: Get*** method was called on a parameter that was not registered using registerOutParameter.

Action: Add registerOutParameter call for the parameter.

CLI0604E

CallableStatement get*** method was called without calling execute.

Cause: The CallableStatement get*** method was called prior to calling CallableStatement execute.

Action: Make sure that CallableStatement execute is called prior to any CallableStatement get*** method.

CLI0605E

CallableStatement get*** method did not match the type used in registerOutParameter.

Cause: The CallableStatement get*** method was not the one that corresponds to the type used in registerOutParameter for this parameter.

Action: Change to the get*** method that matches to the type used in registerOutParameter for this parameter. (Refer to JDBC specification)

CLI0606E

The returned value from a column is incompatible with the data type corresponding to the get*** method.

Cause: The value in a CHAR/VARCHAR/LONGVARCHAR column is not a valid numeric value.

Action: Use the appropriate get methods other than the ones that return numeric values.

CLI0607E

Invalid datetime format.

Cause: The value in CHAR/VARCHAR/LONGVARCHAR column is invalid date/time/ timestamp value.

Action: Use appropriate get*** method other than getDate/getTime/ getTimestamp.

CLI0608E

Invalid conversion.

Cause: The get*** method is invalid on this column type.

Action: Use the valid get*** method (see JDBC specification) to retrieve data from this column.

CLI0609E

Numeric value out of range.

Cause: The value of this column is too large/small for a short or int value, conversion will cause data loss.

Action: Use the get method that can accommodate the value.

CLI0610E

Invalid column number.

Cause: The column number is less than 1 or greater than the total number
 of columns in the ResultSet.

Action: Make sure the column number is no less than 1 and no greater than the total number of columns.

CLI0611E

Invalid column name.

Cause: The given column name cannot be found in the ResultSet.

Action: Ensure the column name is correct.

CLI0612E

Invalid parameter number.

Cause: The given parameter number is less than 1 or greater than the total number of parameters.

Action: Ensure the parameter number is no less than 1 and no greater than the total number of parameters.

CLI0613E CLI0620E

CLI0613E

Program type out of range.

Cause: The object given in the PreparedStatment/CallableStatement setObject is not a valid object type.

Action: Make sure the object to be set is one of the object types allowed in setObject (see JDBC specification).

CLI0614E

Error sending to the socket, server is not responding.

Cause: Error happened while sending data to server, server may be down.

Action: Ensure the server is up, and rerun your program.

CLI0615E

Error receiving from socket, server is not responding.

Cause: Error happened while receiving from server, server may be down.

Action: Ensure JDBC server is up, and rerun your program.

CLI0616E

Error opening socket.

Cause: Cannot open the socket to server, server may be down.

Action: Ensure JDBC server is up, and rerun your program.

CLI0617E

Error closing socket.

Cause: Cannot close the socket to server. Server may be down.

Action: Ensure JDBC server is up, and rerun your program.

CLI0618E

Userid and/or password invalid.

Cause: The given userid/password is not valid.

Action: Make sure the userid/password is correct, and rerun your program.

CLI0619E

Invalid UTF8 data format.

Cause: When getUnicodeStream is called on columns that are not DBCS columns, the decoding from UTF8 format is done, but the data cannot be decoded properly.

Action: Use getString/getBytes/getAsiiStream/getBinaryStream instead.

CLI0620E

IOException, error reading from input stream.

Cause: Error happened when reading data to be put from input stream.

Action: Make sure the file exists, and file length is correctly given.

Chapter 4. DBA Messages

Each message has a message identifier that consists of a prefix (DBA) and the message number. Messages are listed in numeric sequence based upon the message number. These messages are generated by the Database Director, the DBA Utility, Visual Explain, the Snapshot Monitor, and the Event Analyzer.

DBA0010

The application programming interface "<program>" failed to complete successfully with a reason code of "<number>". Try again or contact your local support.

Cause: An application programming interface call failed to complete.

Action: Try the operation again or contact your system administrator.

DBA0011

An unexpected error occurred with the following message from the failing component:.

Cause: An unexpected error occurred.

Action: If the error message text indicate an SQL error, refer to the appropriate documentation.

DBA0012

A background process ended abnormally.

Cause: An unexpected error occurred in the background process causing it to stop.

Action: Contact your system administrator.

DBA0013

An error occurred while attempting to disconnect from database "<name>".

Cause: A process currently connected to the database cannot be disconnected at this time.

Action: Try the operation again or contact your system administrator.

DBA0015

The installation path cannot be retrieved.

Cause: During initialization, the program queries the database manager installation path.

Action: Verify that a DB2 instance has been created. Attempt a db2start.

DBA0016

The instance path cannot be retrieved.

Cause: During initialization, the program queries the database manager instance path.

Action: Verify that a DB2 instance has been created. Attempt a db2start.

DBA0018

The application programming interface "<program>" failed to complete successfully for object "<name>". Try again or contact your local support.

Cause: An application programming interface call failed to complete.

Action: Try the operation again or contact your system administrator.

DBA0040

An error occurred while attempting to connect to database "<name>".

Cause: The requested operation requires a connection to the database.

Action: Ensure the database is accessible. Try the operation again or contact your system administrator.

DBA0041 DBA0105

DBA0041

An error occurred while attempting to attach to database manager "<name>".

Cause: The requested operation requires an attachment to the database manager.

Action: Ensure the database manager is accessible. Try the operation again or contact your system administrator.

DBA0042

An error occurred while attempting to start the database manager.

Cause: The requested operation requires the database manager to be active.

Action: Ensure the database manager is started before trying the operation again.

DBA0049

An error occurred while attempting to load help file "<name>".

Cause: The help file cannot be found or an error occurred while reading the file.

Action: Verify that the help file was installed and is not damaged.

DBA0050

An error occurred while attempting to load DLL "<name>".

Cause: The Dynamic Link Library cannot be found or an error occurred while reading the file.

Action: Verify that the DLL was installed, is not damaged, and its location is part of the LIBPATH parameter in the CONFIG.SYS file.

DBA0100

An error occurred while attempting to access the job control file "<name>".

Cause: The job control file is corrupted.

Action: Possible solutions include:

- Recover the job control file from a backup copy, and submit the request again.
- Remove the job control file. This will remove all information saved about existing jobs.

DBA0101

Are you sure you want to delete job "<number>"? This will delete all data associated with the job.

Cause: You are about to delete the record of a job.

Action: Ensure that you want to delete the job.

DBA0102

Are you sure you want to delete all of the selected jobs? This will delete all data associated with the jobs.

Cause: You are about to delete all records of the selected jobs.

Action: Ensure that you want to delete the jobs.

DBA0103

The job has completed for the request "<description>". For the status and output of the job, view job "<number>" in the recovery jobs window.

Cause: The job has completed.

Action: To view the status and output of the job, use the recovery jobs window.

DBA0104

A job was created for the request "<description>". Its job number is "<number>". To view the status and output of the job, use the recovery jobs window.

Cause: The job was successfully started.

Action: To view the status and output of the job, use the recovery jobs window.

DBA0105

A system error occurred while attempting to create a job for the request "<description>".

Cause: An unexpected error occurred while creating the job, causing it to stop.

Possible reasons include:

- The job control file is full.
- Too many processes are running.
- Insufficient memory.

Action: Delete some jobs from the job control file, or contact your system administrator.

DBA0106

A process created for the job ended abnormally.

Cause: An unexpected error occurred in the background process, causing it to stop.

Action: Ensure that your system has sufficient resources. Resubmit the request if desired.

DBA0107

There are no jobs to display.

Cause: There are no jobs, or there are no jobs that meet the include criteria.

Action: No action is required, or change the include criteria.

DBA0108

Are you sure you want to stop job "<number>"?

Cause: You are about to stop the execution of a job.

Action: Ensure that you want to stop the job.

DBA0109

Are you sure you want to stop all of the selected jobs?

Cause: You are about to stop the execution of all the selected jobs.

Action: Ensure that you want to stop all of the selected jobs.

DBA0120

Are you sure you want to restart database "<name>"?

Cause: You are about to bring the database out of the restart pending state.

Action: Ensure that you want to restart the database.

DBA0121

Are you sure you want to restart all of the selected databases?

Cause: You are about to bring the databases out of the restart pending state.

Action: Ensure that you want to restart the databases.

DBA0122

Restart failed because indoubt transactions exist against the database and the connection to the database was dropped.

Cause: The restart operation encountered indoubt transactions. This left the database in an inconsistent state. The DBA Utility dropped the connection to the database.

Action: Resolve the indoubt transactions. If they are not resolved, applications must restart the database whenever it is required.

If you are in an XA/DTP environment, and the transaction manager that was using the database is available, instruct it to resolve the indoubt transactions.

Alternatively, with great caution, use the Command Line Processor to heuristically complete the transactions.

DBA0130

Database "<name>" is not roll-forward pending, and none of its table spaces is roll-forward pending. Resubmit the request when it is necessary.

Cause: The specified database does not need to be rolled forward, or is not configured for roll-forward recovery.

Action: Resubmit the request when roll-forward recovery is necessary.

DBA0131

Database "<name>" is not roll-forward pending. Resubmit the request when it is necessary.

Cause: The specified database does not need to be rolled forward, or is not configured for roll-forward recovery.

Action: Resubmit the request when roll-forward recovery is necessary.

DBA0132 DBA0156

DBA0132

None of the table spaces for database "\<name>" is roll-forward pending. Resubmit the request when it is necessary.

Cause: The table space or table spaces do not need to be rolled forward, or the database is not configured for roll-forward recovery.

Action: Resubmit the request when roll-forward recovery is necessary.

DBA0133

Are you sure you want to stop recovering database "\<name>"? Any uncommitted transactions will be rolled back and the database will be ready for use.

Cause: Recovering a database requires restoring the database and optionally rolling forward transactions from log files. After each step, you can stop the recovery and make the database ready for use, or you can continue the recovery later if desired.

Action: Proceed if you have completed all the desired recovery for the database.

DBA0134

Are you sure you want to stop recovering all of the selected databases? Any uncommitted transactions will be rolled back and the databases will be ready for use.

Cause: Recovering a database requires restoring the database and optionally rolling forward transactions from log files. After each step, you can stop the recovery and make the database ready for use, or you can continue the recovery later if desired.

Action: Proceed if you have completed all the desired recovery for all of the selected databases.

DBA0140

The recovery has been completed and the accessible table spaces have been restored. To view the status of the storage, use the windows that display the details about the table spaces and their containers. To recover inaccessible table spaces, select "Pause to allow table space container redefinition", and resubmit the request.

Cause: The restore process only restores table spaces with accessible storage. There are one or more table spaces whose storage must be redefined.

Action: If desired, select "Pause to allow table space container redefinition", and resubmit the request.

DBA0150

An error occurred while processing the print request. The temporary print file could not be opened.

Cause: The file "/tmp/.db2ddtmp" is a temporary file that is used for printing. It could not be opened.

Action: Possible solutions include:

- Verify whether the "/tmp" directory is full.
- Verify whether the "/tmp" directory is accessible.

DBA0151

An error occurred while processing the print request. The buffer was corrupted.

Cause: The buffer used for printing was not saved properly.

Action: Resubmit your print request. If the problem persists, contact your system administrator.

DBA0152

An error occurred while processing the print request. The lpr command returned a non-zero return code.

Cause: The lpr command returned a non-zero return code.

Action: Resubmit your print request. If the problem persists, contact your system administrator.

DBA0156

The number of objects retrieved exceeds the limit of what can fit in the window. A partial list is displayed.

Cause: The window height required to display the retrieved objects exceeds a system limit.

Action: You can reduce the number of objects displayed in the window by doing the following:

- If you are in a tree view, collapse parts of the tree. You can also change the settings to hide the icons in the window. Select "Director", then "Open as",

and then "Settings" to bring up the Settings notebook. On the Tree View page, make sure there is no check mark in the "Display icons" check box.
- If the "Include" choice is available on the "View" menu, you can specify a subset of the objects to display in the window. When you select "Include", a window appears and prompts you for subsetting criteria.

DBA0200

A system error occurred while attempting to start the command.

Cause: An unexpected error occurred in the command, causing it to stop.

Action: Possible solutions include:

- Verify that your system has sufficient memory.
- Where appropriate, stop other programs using the system.
- Contact your system administrator.

DBA0201

The command is specified incorrectly.

Cause: The command has invalid syntax.

Action: Issue the command with -h option to see the correct syntax.

DBA0202

The name of the database must be between 1 and "<maximum>" characters long.

Cause: The specified database name is either empty or too long.

Action: Reissue the command with a valid database name.

DBA0203

The explainable statement text must be between 1 and "<maximum>" characters long.

Cause: The specified explainable statement text is either empty or too long.

Action: Reissue the command with valid explainable statement text.

DBA0204

The query tag cannot have more than "<maximum>" characters.

Cause: The specified query tag is too long.

Action: Reissue the command with a valid query tag.

DBA0205

The user ID cannot have more than "<maximum>" characters.

Cause: The specified userid is too long.

Action: Reissue the command with a valid userid.

DBA0206

The valid range for query number is from 0 to "<maximum>".

Cause: The specified query number is either too small or too large.

Action: Reissue the command with a valid query number.

DBA0207

The valid values for optimization class are 0, 1, 3, 5, 7 and 9.

Cause: The specified optimization class is not one of the valid values.

Action: Reissue the command with a valid optimization class.

DBA0208

The database "<name>" cannot be found.

Cause: The database name specified in the command does not exist in the system database directory.

Action: Reissue the command with an existing database name or catalog the database in the system database directory.

DBA0209 DBA2055

DBA0209

The database manager "<name>", in which the specified database resides, is not cataloged in the node directory.

Cause: The database manager is not cataloged in the node directory.

Action: Catalog the database manager in the node directory.

DBA1000

Are you sure you want to exit the Database Director? This will close all windows associated with it.

Cause: You are about to exit the Database Director. Exiting the Database Director will close all windows associated with it.

Action: Ensure that you want to exit the Database Director.

To close an individual window, select "Close" on the window. To exit the Database Director, close all Database Director windows, or select "Exit".

To stop displaying this confirmation, go to the Database Director settings window.

DBA1001

Are you sure you want to refresh all the objects displayed?

Cause: You are about to refresh all objects displayed.

Action: Ensure that you want to refresh all the objects.

In the tree view it is faster to refresh objects selectively than to refresh the entire tree of objects.

To stop displaying this confirmation, go to the Database Director settings window.

DBA2002

A system error occurred while attempting to start the command.

Cause: An unexpected error occurred in the command causing it to stop.

Action: Possible solutions include:

- Verify that your system has sufficient memory.
- Where appropriate, stop other programs using the system.

- Contact your system administrator.

DBA2003

A system error occurred while performing the operation.

Cause: An unexpected error occurred causing the operation to stop.

Action: Contact your system administrator.

DBA2004

An error occurred while attempting to access help file "<name>".

Cause: The help file could not be found, or an error occurred while reading the file. Help will not be available.

Action: Verify that the help files were installed and are not damaged.

DBA2052

There are no recovery history entries to display.

Cause: The recovery history of the database has no entries, or there are no entries that meet the include criteria.

Action: No action is required, or change the include criteria by selecting the Include choice in the View menu.

DBA2053

The database has no table spaces.

Cause: The database has no table spaces.

Action: No action is required.

DBA2054

The table space has no containers.

Cause: The table space has no containers.

Action: No action is required.

DBA2055

The system database directory has no entries.

Cause: The system database directory has no entries.

Action: No action is required.

DBA2056

The node directory has no entries.

Cause: The node directory has no entries.

Action: No action is required.

DBA2057

The database connection services directory has no entries.

Cause: The database connection services directory has no entries.

Action: No action is required.

DBA2058

There are no databases at the specified location.

Cause: There is no local database directory at the specified location.

Action: Specify another location.

DBA2100

Are you sure you want to uncatalog database "<name>"?

Cause: You are about to uncatalog the database.

Are you sure you want to uncatalog dcoecb

DBA2106

Are you sure you want to drop table space "<name>"? This will delete all data associated with this table space.

Cause: You are about to drop the table space. Dropping a table space deletes all the objects in the table space. You must have SYSADM authority to drop a table space.

Action: Ensure that you want to drop the table space.

DBA2107

Are you sure you want to drop all of the selected table spaces? This will delete all data associated with these table spaces.

Cause: You are about to drop the table spaces. Dropping table spaces deletes all the objects in the table spaces. You must have SYSADM authority to drop table spaces.

Action: Ensure that you want to drop all of the selected table spaces.

DBA2110

Are you sure you want to overwrite file "<name>"?

Cause: You are about to overwrite an existing file.

Action: Ensure that you want to overwrite the file.

DBA2150

The database alias "<name>" already exists for another database.

Cause: You requested that a backup be recovered into a new database, but the database name you specified is already used as an alias for an existing database.

When a database is created, the database is cataloged in the system database directory using the database name as the alias, and this alias must be unique.

Action: Specify a database name that is not being used as an alias, or request that the backup be recovered into the existing database.

DBA2151

The database alias "<name>" was not found in the system database directory.

Cause: You requested that a backup be recovered into an existing database, but a database no longer exists with this alias.

Action: Select an existing database, or request that the backup be recovered into a new database.

DBA2152

Change failed. The original directory entry was uncataloged, but cataloging the new entry failed with SQLCODE "<sqlcode>". Cataloging the original entry also failed with SQLCODE "<sqlcode>".

Cause: To change the directory entry, the DBA Utility uncataloged the existing entry and tried to catalog an entry with new values. This catalog operation failed and an attempt to catalog the existing entry again also failed.

Action: Try to recatalog the old entry, or to catalog the new entry. If cataloging fails again, read the error you receive for instruction on how to proceed.

DBA2154

Select the appropriate containers and the operation you want to perform on them.

Cause: To remove a file system, select one or more containers from the list and select "Remove file systems". To unmount a file system, select one or more containers and select "Unmount file systems" (only containers with a file system mounted can be unmounted).

Action: Select one or more appropriate containers from the list, and resubmit the request.

DBA2198

Your request "<description>" is not valid.

Cause: Possible reasons include:

- One or more values you entered are not valid.
- The operation cannot be performed under the current state of the database or the database manager.

Action: Make the necessary corrections and resubmit the request.

DBA2199

Provide a value for "<name>".

Cause: A value is required.

Action: Provide the value and resubmit the request.

DBA2200

Date "<date>" is not valid. Enter a date in the format displayed.

Cause: The specified date is not valid.

Action: Enter the date in the format displayed in the window.

DBA2201

Time "<time>" is not valid. Enter a time in the format displayed.

Cause: The specified time is not valid.

Action: Enter the time in the format displayed in the window.

DBA2202

The value for "<parameter>" must be between "<minimum>" and "<maximum>".

Cause: The value must be within the specified range.

Action: Correct the value and resubmit the request.

DBA2203

An error occurred while attempting to write to file "<name>".

Cause: An unexpected error occurred while write to the file.

Possible reasons include:

- Invalid file name or path.
- Insufficient authorization.

Action: Make the appropriate correction and resubmit the request.

DBA2204

An error occurred while attempting to read file "<name>".

Cause: An unexpected error occurred while reading the file.

Possible reasons include:

- The file is missing or damaged.
- The file is bigger than what the utility can handle.
- Insufficient authorization.

Action: Make the appropriate correction and resubmit the request.

DBA2205

The value for "<name>" is not valid.

Cause: The value you entered is not valid.

Action: Correct the value and resubmit the request.

DBA2206

The value for "<name>" must not exceed "<maximum>".

Cause: The value you entered is not valid.

Action: Correct the value and resubmit the request.

DBA2207

The value for "<name>" must not be smaller than "<minimum>".

Cause: The value you entered is not valid.

Action: Correct the value and resubmit the request.

DBA2208

The "<name>" entries must not exceed "<maximum>" bytes.

Cause: The value you entered exceeded the maximum number of bytes allowed.

Action: Correct the value and resubmit the request.

DBA2209   DBA2218

DBA2209

Provide a value for "<field>" in the "<name>" page.

DBA2219

An error occurred while copying the contents of the directory.

Cause: The directory contents were not copied correctly, possibly due to insufficient DASD, or a system error.

Action: Check the amount of free space on your system. If the problem persists, contact your system administrator.

DBA2220

The attempt to unmount a file system failed.

Cause: An attempt to unmount a file system failed. Possible reasons include an incorrect mount path, or insufficient authority.

Action: Check the path you have specified and your system authority.

DBA2221

The attempt to mount a file system failed.

Cause: An attempt to mount a file system failed. Possible reasons include an incorrect mount path, a mount path already in use, or insufficient authority.

Action: Check the path you have specified and your system authority.

DBA2222

An error occurred while attempting to remove a directory.

Cause: An attempt to remove a directory failed. Possible reasons include an incorrect path or a system failure.

Action: Check the path you have specified.

DBA2223

An error occurred while attempting to remove a file system.

Cause: An attempt to remove a file system failed. Possible reasons include an incorrect mount path, a file system already removed, or insufficient authority.

Action: Check that the file system exists, that the specified mount path is correct, and that you have SYSADM authority.

DBA2224

An error occurred while attempting to extend a file system.

Cause: An attempt to extend a file system failed. Possible causes include insufficient DASD, insufficient authority, or an incorrect path.

Action: Ensure that you have sufficient free DASD within the associated volume group, that your system authority is SYSADM, and that you have specified the correct path name.

DBA2225

The specified database could not be found.

Cause: The specified database could not be located. A possible cause is an invalid alias.

Action: Use a valid database alias.

DBA2226

An error occurred while attempting to create a file system.

Cause: An attempt to create a file system failed. Possible reasons include an incorrect mount path, a mount path already in use, insufficient authority, or insufficient DASD within the associated volume group.

Action: Check the path you have specified, your system authority, and ensure there is sufficient free space in the associated volume group.

DBA2227

The specified volume group is not valid.

Cause: The specified volume group does not exist.

Action: Specify a valid volume group name for the system to which you are connected.

DBA2228

The specified path is not valid.

Cause: The specified path does not exist, or is not valid for the intended operation.

Action: Specify a valid path.

DBA2230   DBA2281

DBA2230

Some of the paths associated with the selected containers are already mount points for existing file systems. Select "Continue" to remove these file systems and continue with the mount operation, or select "Cancel" to cancel the request.

Cause: You attempted to mount file systems with paths already in use by other file systems. One possible cause is that the file systems were not removed when the database or table space using these file systems was dropped.

Action: Select "Continue" to remove the other file systems and continue with the mount operation, or select "Cancel" to cancel the request.

DBA2250

The database configuration was successfully updated. All applications must disconnect from database "<name>" before the changes become effective.

Cause: The database configuration was successfully updated, but the active database configuration cannot be changed until all applications are disconnected from the database. Once all applications are disconnected, the changes will come into effect with the first connection to the database.

Action: Ensure that all applications are disconnected from the database, and then reconnect to the database.

DBA2251

The database manager configuration was successfully updated. The database manager "<name>" must be stopped before most changes become effective.

Cause: The database manager configuration was successfully updated.

Changes to the configuration parameter "dftdbpath" become effective immediately.

For the remaining configuration parameters, the changes do not come into effect until all applications have disconnected from the database, and the database manager is successfully stopped and started again.

Action: To ensure that all changes come into effect, ensure that all applications have disconnected from the database, stop the database manager, and start it again.

You can stop and start the database manager by executing the db2stop and db2start commands, respectively.

DBA2252

Directory changes may not be effective until the directory cache is refreshed.

Cause: If directory caching (database manager configuration parameter dir_cache) is enabled, system database, node, and database connection services directory files are cached in memory. Directory changes may not become effective until the directory cache is refreshed. Refer to the dir_cache configuration parameter for a description on directory caching.

Action: To refresh directory cache, exit the application and re-invoke the database director. To refresh the directory information, invoke "Refresh" in the details view of the directory.

DBA2270

The following message was received from the database manager:

Cause: A message was received from the database manager.

Action: Proceed based on to the message displayed.

DBA2281

The recovery request on database "<name>" has been suspended because some table space containers need to be redefined. Select "Redefine..." to redefine the containers. Select "Continue" to continue the recovery after you have redefined the containers. Select "Cancel" if you choose to stop the recovery.

Cause: The recovery is waiting for the containers to be redefined.

Action: Redefine the containers and select "Continue" to proceed, or select "Cancel" to stop the recovery.

---- **DBA2282** ----

The restore agent is not available to process your request on database "<name>". Resubmit your request later.

Cause: Possible reasons are:

- The restore agent is used exclusively by a restore or redefine table space container operation.
- The restore agent terminated abnormally.

Action: Resubmit your request later.

---- **DBA2283** ----

The restore agent is no longer available to process your request on database "<name>". Resubmit a recovery request if desired.

Cause: The restore agent is no longer available because the restore operation has completed or has been stopped.

Action: Resubmit a recovery request if desired. This will create another restore agent to process your request.

---- **DBA2284** ----

Select "Restore", "Roll forward" or "Make the database ready for use after recovering it" before submitting your request.

Cause: There is nothing to be done in your request. You must select at least one of the following operations:

- Restore
- Roll forward
- Make the database ready for use after recovering it

Action: Select the desired operations and resubmit the request.

---- **DBA2285** ----

Select "Restore" or "Roll forward the database to the end of logs" before submitting your request.

Cause: There is nothing to be done in your request. You must select at least one of the following operations

- Restore
- Roll forward the database to the end of logs

Action: Select the desired operations and resubmit the request.

---- **DBA2286** ----

Select "Entire backup image", "Recovery history", or "Table space table" to restore.

Cause: You must specify what to restore.

Action: Select what to restore and resubmit the request.

---- **DBA2287** ----

The table space requires at least one container in order to continue with the redefinition request.

Cause: You must specify at least one container.

Action: Select "Add" and fill in the necessary fields to add a container to the list.

---- **DBA2288** ----

No containers have been specified. Select "Add" in the "<name>" page in order to create containers.

Cause: You must specify at least one container.

Action: Select "Add" and fill in the necessary fields to add a container to the list.

---- **DBA2289** ----

No containers have been specified. Select "Add" in order to create containers.

Cause: The current list does not contain any new containers.

Action: Select "Add" and fill in the necessary fields to add a container to the list.

---- **DBA2291** ----

Select a database to fill the system database directory entry in the catalog view.

Cause: You must select a database entry to fill the view.

Action: Select a database entry from this view to fill the catalog view.

DBA2300 DBA3011

DBA2300

"<device>" is not a valid tape device.

Cause: The operating system did not accept the specified tape device. Local tape devices are of the form "\\.\TAPEn", where n represents the drive number (0 is the first).

Action: Correct the name of the tape device and resubmit the request.

DBA2301

Tape device "<device>" does not support a tape mark blocksize of "<blocksize>".

Cause: The DBA Utility uses a tape mark blocksize that is not supported by the specified tape device.

Action: Use the Command Line Processor to submit the request. Specify a supported blocksize in the DB2 INITIALIZE TAPE command.

DBA2302

An error occurred operating tape device "<device>".

Cause: An unexpected error occurred while operating the specified tape device.

Action: Resubmit the request. If the problem persists, contact your system administrator.

DBA2900

You must specify all of the required fields before your request can be processed.

Cause: Some required inputs are not filled in.

Action: Fill in the fields and resubmit the request.

DBA3007

Are you sure you want to delete the explained statement?

Cause: You are about to delete the explained statement. This will delete the rows in the corresponding explain tables.

Action: Ensure that you want to delete the explained statement.

DBA3008

Are you sure you want to delete the explained statements?

Cause: You are about to delete the explained statements. This will delete the rows in the corresponding explain tables.

Action: Ensure that you want to delete the explained statements.

DBA3009

A system error occurred. The Visual Explain tool could not continue processing.

Cause: An unexpected error occurred causing the operation to stop.

Action: Possible solutions include:

- Verify that your system has sufficient memory.
- Where appropriate, stop other programs using the system.
- Contact your system administrator.

DBA3010

The explain snapshot has been deleted. Your request cannot be completed.

Cause: The explain snapshot that this request is acting on has been deleted from the explain tables.

Action: Regenerate the snapshot.

DBA3011

The access plan graph of this explain snapshot cannot be displayed. Your request cannot be completed.

Cause: Either the explain snapshot is corrupted or the explain snapshot is constructed from a newer version of the database manager than your Visual Explain Tool. The access plan graph cannot be constructed by the Tool.

Action: Explain your statement and try again if you suspect the snapshot is corrupted. Use a version of Visual Explain Tool compatible with your database manager to graph the explain snapshot otherwise.

DBA3012

String not found.

Cause: The search string is not found in the text being displayed.

Action:

DBA3013

Text is missing. You must specify the text string to be found.

Cause: You did not specify a search string.

Action: Specify a search string.

DBA3014

No selection is made in the list. You must select at least one entry in the list.

Cause: You did not make a selection in the list.

Action: Select at least one entry in the list.

DBA3015

There are no referenced columns associated with this access plan.

Cause: There were no columns referenced in the query for the referenced table.

Action:

DBA3016

There are no page fetch pairs statistics associated with this index.

Cause: There are no page fetch pairs statistics stored in the catalog table for this index. Refer to the Administration Guide for more details.

Action: Refer to the Administration Guide for more details.

DBA3017

There are no referenced functions associated with this access plan.

Cause: This access plan did not require the use of any functions.

Action:

DBA3018

There are no column distribution statistics associated with this column.

Cause: There are no column distribution statistics stored in the catalog table for this column. Refer to the Administration Guide for more details.

Action: Refer to the Administration Guide for more details.

DBA3019

There are no indexes associated with this table.

Cause: There were no indexes defined on the referenced table at the time of explain and there are no indexes currently defined in the table.

Action:

DBA3020

An error occurred while a COMMIT against the database was being attempted and no SQLCODE can be returned.

Cause: An unexpected error occurred.

Action: Contact your system administrator.

DBA3021

An error occurred while a ROLLBACK against the database was being attempted and no SQLCODE can be returned.

Cause: An unexpected error occurred.

Action: Contact your system administrator.

DBA3022

The Get Database Manager Configuration api passed back a non-zero return code.

Cause: An unexpected error occurred causing the operation to stop.

Action: Contact your system administrator.

DBA3023

The Get Database Configuration api passed back a non-zero return code.

Cause: An unexpected error occurred causing the operation to stop.

Action: Contact your system administrator.

DBA3024

An error occurred while saving to the file "<file>".

Cause: Possible problems include:

- The file cannot be opened for write.
- The file name does not conform to the file system convention.

Action: Possible solutions include:

- Change the file attribute to read write.
- Specify the correct file name.

DBA3025

An error occurred while retrieving from the file "<file>".

Cause: Possible problems include:

- The file is not readable.
- The file does not exist.

Action: Possible solutions include:

- Specify the correct file name.

DBA3026

No printers are installed.

Cause: There are no printers installed on your machine.

Action: Install a printer on your machine.

DBA3033

There are no referenced table spaces associated with this access plan.

Cause: This access plan did not reference any database objects contained in a table space.

Action:

DBA4000

Are you sure you want to stop monitoring and exit the Snapshot Monitor.

Cause: You have requested to stop monitoring all database objects that are currently monitored and to exit the Snapshot Monitor.

Action: You may press Yes to process the request or No to ignore it.

DBA4003

You have selected too many performance variables to be shown in this view. The limit for both performance variable and information element columns is "<number>".

Cause: This occurs when:

- The number of performance variables in the current performance variable profile that have the DISPLAY(SUMMARY) keyword exceeds the limit for this view, or
- When you select too many performance variables using the "Include performance variables" action when the view is open

If the number of performance variables to be displayed in this view exceeds the limit, some performance variables are ignored and are not shown in the view.

Action: If a performance variable that you have selected for viewing is not in the view, use the "Include performance variables" action to remove a performance variable already displayed and select the desired performance variable.

DBA4004

An unexpected error occurred while trying to get a snapshot. The Snapshot Monitor is paused. The message from the failing component is: "<message>".

Cause: The Snapshot Monitor has paused the collection of snapshots due to the unexpected error returned by the database manager that is being monitored. This usually occurs when the database manager is stopped or communication with the database manager is interrupted.

Action: Determine the cause of the error from the referenced message.

If you do not wish to continue monitoring, select the "Stop all monitoring" choice from the "Snapshot monitor"

---

menu on any of the Snapshot Monitor windows for the database manager in question.

If you wish to continue monitoring, first correct the error and then select the "Resume monitoring" choice from the "Snapshot monitor" menu on any of the Snapshot Monitor windows for the database manager in question.

DBA4005

An unexpected error occurred while displaying the performance graph view. Presentation Manager returned the following error identifier: "<number>".

Cause: An unexpected error occurred while using Presentation Manager services.

Action: Write down the error identifier returned by Presentation Manager and contact support.

DBA4012

A fatal error has occurred in the database system monitor. It has returned invalid data.

Cause: The database system monitor returned data that is not recognized by the Snapshot Monitor. This can occur due to network problems or internal problems with the database manager system.

Action: Try your action again, and if the error persists please contact support.

DBA4070

The IBM-supplied performance variable profile "<name>" cannot be found.

Cause: The IBM-supplied performance variable profile is not found in the installation directory.

Action: Either reinstall the Snapshot Monitor, or specify a new performance variable profile to be used. To specify a new profile use the "Open as settings" action from the "Snapshot monitor" menu in any of the views.

DBA4071

The Snapshot Monitor tried to write to the performance variable profile "<name>", but it failed due to file access permissions.

Cause: The Snapshot Monitor encountered problems while writing to the specified performance variable profile.

Action: Examine file permissions for the file or the directory where the file resides, and make sure that they allow writing to the file.

DBA4072

The Snapshot Monitor detected "<number>" syntax errors while parsing the specified performance variable profile. "<number>" performance variable definitions were parsed correctly. For errors look in file "<name>".

Cause: Some of the performance variable definitions in the specified profile are invalid and are not recognized by the Snapshot Monitor.

Action: Examine the errors listed in the error file and fix them, pause the monitor, open the snapshot monitor settings view and press OK to pick up the changes, resume monitoring.

DBA4080

You have specified a capture interval that is not between 00:00:01 and 99:59:59. Please specify a valid value.

Cause: The value entered for the capture interval cannot be used by the Snapshot Monitor, because it does not fall into the specified range.

Action: Type a valid value.

DBA4091

The specified performance database does not exist.

Cause: The performance database has been dropped or uncataloged since the last time the Snapshot Monitor was used.

Action:

Do one of the following actions:

• Create a database with the same name as the old performance database.

DBA4094   DBA4703

- Select a new performance database for saving alerts. Use the "Open as settings" action from the "Snapshot monitor" menu in any view to display the Settings window. Type the database name in the "Write alerts in performance database" field.

DBA4094

The performance variable profile "<name>" was not found. You must specify a valid performance variable profile in order to use the Snapshot Monitor.

Cause: The performance variable profile last used with the Snapshot Monitor was either deleted or moved to a different location.

Action: Specify a new profile name. Use the "Open as settings" action from the "Snapshot monitor" menu in any view to display the Settings window, and type the profile name in the "Performance variable profile" field.

DBA4100

The X axis size must be between 1 and 999 snapshots.

Cause: You have entered a number that is outside this range.

Action: Enter a value that falls within the range.

DBA4110

There is already a performance variable profile with the same name in your default profile directory. Do you want to replace it with the IBM-supplied profile?

Cause: You have selected to copy the IBM-supplied performance variable profile. However, a profile with the identical name already exists in that directory.

Action: Select "OK" to copy the file and to overwrite the existing one, or "Cancel" to ignore the request.

DBA4115

An invalid combination of threshold and rearm values has been entered.

Cause: Upper rearm values must always be less than or equal to upper threshold values. Lower rearm values must always be greater than or equal to lower threshold values.

Action: Re-enter your values, and make sure they meet the above conditions.

DBA4701

The database manager "<name>" where the performance database "<name>" resides is not started, or is not cataloged, or cannot be accessed due to a communication problem. No alerts will be recorded.

Cause: An ATTACH command to the database manager that manages the performance database failed, or the database manager is not cataloged.

No alerts will be recorded. The Snapshot Monitor setting to write alerts has been disabled.

Action: Make sure the database manager that manages the performance database is started and/or cataloged. To re-enable the writing of alerts to the performance database, use the "Open as settings" action from the "Snapshot monitor" menu in any view to display the Settings window. Then select the "Write alerts in performance database" check box.

DBA4702

The performance database "<name>" for recording alerts does not exist in the system database directory. No alerts will be recorded.

Cause: The performance database specified in the Snapshot Monitor Settings window does not exist in the system database directory.

No alerts will be recorded. The Snapshot Monitor setting for writing alerts has been disabled.

Action: Specify a new performance database. Use the "Open as settings" action from the "Snapshot monitor" menu in any view to display the Settings window, select the "Write alerts in performance database" check box, and specify the new performance database name.

DBA4703

The table SSMON_ALERT does not have the definition expected by the Snapshot Monitor. No alerts will be recorded.

Cause: The table SSMON_ALERT in the performance database does not have the definition expected by the Snapshot Monitor.

No alerts will be recorded. The Snapshot Monitor setting for writing alerts has been disabled.

Action: Drop the table SSMON_ALERT. Use the "Open as settings" action from the "Snapshot monitor"

menu in any view to access the Settings window, select the "Write alerts in performance database" check box. The SSMON_ALERT table will be created for you by the Snapshot Monitor.

DBA4704

The table SSMON_ALERT used for recording alerts was created in the performance database by the Snapshot Monitor.

Cause: The table SSMON_ALERT was created in the performance database by the Snapshot Monitor, as specified in the Snapshot Monitor Settings window. For details on the format of the SSMON_ALERT table, see the file CALRTTBL.DDL in the samples directory.

Action:

DBA4705

An error occurred while a COMMIT for inserts made to the alert table in the performance database was being attempted, and no SQLCODE can be returned. No alerts will be recorded.

Cause: An unexpected error occurred.

No alerts will be recorded. The Snapshot Monitor setting for writing alerts has been disabled.

Action: Contact your system administrator.

DBA4706

An error occurred while updates to the alert table were being attempted. A ROLLBACK against the performance database failed, and no SQLCODE can be returned. No alerts will be recorded.

Cause: An unexpected error occurred.

No alerts will be recorded. The Snapshot Monitor setting for writing alerts has been disabled.

Action: Contact your system administrator.

DBA4707

The table SSMON_ALERT does not exist. It may have been deleted. No alerts will be recorded.

Cause: The table SSMON_ALERT does not exist in the performance database. It may have been deleted.

No alerts will be recorded. The Snapshot Monitor setting for writing alerts has been disabled.

Action: To re-enable the writing of alerts to the performance database, use the "Open as settings" action from the "Snapshot monitor" menu in any view to display the Settings window. Then select the "Write alerts in performance database" check box. The SSMON_ALERT table will be created for you by the Snapshot Monitor.

DBA4708

A performance variable exceeded its upper threshold value.

Cause: You have defined an upper threshold for the specified performance variable, and it was exceeded.

Action: Examine the definition of the performance variable. If the threshold value is not valid, change it to avoid further alerting.

If this is an IBM-supplied performance variable, use the "Performance variable" action from the "Help" menu to display the online help, and examine the description of the performance variable and suggested actions when an alert is generated.

Otherwise, you may want to contact your system administrator for further explanation and investigation.

DBA4709

The following performance variable fell below its lower threshold value "<name>".

Cause: You have defined a lower threshold for the specified performance variable, and the performance variable fell below it.

Action: Examine the definition of the performance variable. If the threshold value is not valid, change it to avoid further alerting.

If this is an IBM-supplied performance variable, use the "Performance variable" action from the "Help" menu to display the online help, and examine the description of the performance variable and suggested actions when an alert is generated.

DBA4710   DBA5300E

Otherwise, you may want to contact your system administrator for further explanation and investigation.

DBA4710

You have canceled the "<name>" command on the performance database. No alerts will be recorded.

Cause: The "Cancel" push button in the Logon Window was pressed while working with the performance database. No alerts will be recorded. The Snapshot Monitor setting for writing alerts has been disabled.

Action: To re-enable the writing of alerts to the performance database, use the "Open as settings" action from the "Snapshot monitor" menu in any view to display the Settings window. Then select the "Write alerts in performance database" check box.

DBA5006E

PIPE Event Monitors are not supported.

Cause: The Event Analyzer only handles traces produced by FILE Event Monitors.

Action: Use a FILE Event Monitor.

DBA5007E

The specified Event Monitor does not exist.

Cause: The Event Monitor names specified with the -evm option could not be found in the catalogs for the database specified with the -db option. The Event Monitor may have been dropped or you may be trying to connect to the wrong database.

Action: Make sure that the Database alias specified with the -db is properly cataloged and that the Event Monitor has not been dropped. If the latter is the case, re-create the event monitor.

DBA5250I

The number of objects retrieved exceeds the limit of what can fit in the window. A partial list is displayed.

Cause: The window height required to display the retrieved objects exceeds a system limit.

Action: You can reduce the number of objects displayed in the window by doing the following:

- If the "Include" choice is available on the "View" menu, you can specify a subset of the objects to

display in the window. When you select "Include", a window appears and prompts you for subsetting criteria.

DBA5300E

Event Analyzer invocation error.

Cause: Two modes of operation are provided for reading Event Monitor trace files:

1. User specifies the directory where the trace files reside (using -path)

   This mode allow users to move trace files from a server and analyze them locally, even if the Event Monitor has been dropped.

2. User specifies the database and event monitor names; db2eva automatically locates the trace files.

   When this mode is used, db2eva connects to the database, and issues a 'select target from sysibm.syseventmonitors' to locate the directory where the Event Monitor writes its trace files. The connection is then released, unless -conn is specified. The Event Monitor must not have been dropped when using this mode.

NOTE: Both the -conn and -db options require that the Database with which the trace was recorded still exists, and that the user has sufficient access privileges to connect and select from the system catalogs.

Action: Re-issue the command with valid arguments.

Usage:

```
db2eva -path evmon-target ffl-conn ffl-db database-alias"" |
     -db database-alias -evm evmon-name ffl-conn"
```

Where:

-path evmon-target specifies the directory containing the Event Monitor trace files.

-db database-alias specifies the database for which the Event Monitor is defined, as cataloged on the machine where the trace is analyzed. database-alias overrides the database name specified in the trace.

-evm evmon-name is the Event Monitor producing the trace files to analyze. If specified, db2eva locates the trace files for the user by reading the Event Monitor target from the database system catalog.

-conn requests db2eva to maintain a connection to the database specified by the -db option, or if -db is not supplied, to the database specified in the trace file header. Keeping a connection allows the Event Analyzer to obtain information that is

not contained in the trace files, such as the text for static SQL statements. (The Statement Text Events for static SQL only contain package creator, package number, and section number. When the -conn option is specified, db2eva connects to the database and retrieves the text from the Database system catalog, 'sysibm.sysstmt'.) The default is not to keep a connection. Window Manager Options can be intermixed with the db2eva arguments.

DBA5350E

The specified file cannot be opened.

Cause: The specified file could not be opened in read-only mode for an unknown cause.

Action: Check the spelling of the directory specified by the -path option, its access authority, and that it contains a non-empty, readable trace file called 00000000.evt.

NOTE: The first trace file for an Event Monitor is always named 00000000.evt and cannot be renamed.

DBA5351E

The specified file cannot be found.

Cause: There is no 00000000.evt file in the directory specified by -path.

Action: Check the spelling of the directory specified by the -path option, its access authority, and that it contains the trace file 00000000.evt.

NOTE: The first trace file for an Event Monitor is always named 00000000.evt and cannot be renamed.

DBA5352E

The path specified was not found.

Cause: The -path option specifies a directory that does not exist.

Action: Check the spelling of the directory specified by the -path option, its access authority, and that it contains the trace file 00000000.evt.

NOTE: The first trace file for an Event Monitor is always named 00000000.evt and cannot be renamed.

DBA5353E

Access to the specified file was denied.

Cause: The specified file could not be opened in read-only mode. The -path option may be specifying a directory to which you do not have sufficient access authority.

Action:

- Check the access authority to the directory specified by the -path option.
- Verify that no one has locked the specified file in exclusive mode.

DBA5354E

The data cannot be read from the specified file.

Cause: The Event Monitor trace file contains unexpected data, or cannot be accessed.

The trace file was likely corrupted during transmission or has been removed.

Action: Transmit your trace files from your server again. When transmitting trace files from a remote server, ensure that the transmission is done in binary mode.

DBA5355E

An Event Monitor log header could not be found in the specified file.

Cause: The first file written to by an Event Monitor is named 00000000.evt and contains a data structure that identifies the trace characteristics. This data structure could not be read. Possible causes:

- The trace file was corrupted.
- The trace file is empty. This can happen if the Event Monitor is active but, has not flushed its buffers yet.

Action:

- If the trace was transmitted from a remote server, retransmit, ensuring that the transmission is done in binary mode.
- If the trace file is empty, turn off the Event Monitor to force it to flush its buffer.

DBA5356E   DBA5357E

DBA5356E
The byte order is not valid in the specified file.

Cause: The Event Monitor log header (the first record written to a new trace) specifies whether the trace contains little-endian (e.g., OS/2) or big-endian (e.g., AIX) data. The value found in the trace file specified neither of these two supported types of data.

The trace file was likely corrupted during transmission.

Action: Transmit your trace files from your server again. When transmitting trace files from a remote server, ensure that the transmission is done in binary mode.

DBA5357E
Unexpected data was found at the specified offset in the specified file.

Cause: The Event Monitor trace file contains unexpected data.

The trace file was likely corrupted during transmission.

Action: Retransmit your trace files from the server, ensuring that the transmission is done in binary mode.

Chapter 5. DBI Messages

This chapter contains the error messages which may be generated by installation.

DBI1001E

Usage: "<command-name>" instanceName [-a authType] [-u udfName]

Cause: An incorrect argument was entered. The command accepts the required argument instanceName and the arguments authType and udfName.

Action: Enter the command again as follows:

```
command instanceName [-a authType] [-u udfName]
  command       is db2icrt or db2instance
  instanceName  is the name of the instance
  -a authType   is the authentication type
                (SERVER, CLIENT or DCS) for the
                instance. The default is SERVER.
  -u udfName    is the name of the user under
                which fenced user defined
                functions will be run.
```

DBI1002E

The instanceName "<instance-name>" is invalid.

Cause: The instance name must be the login name of an existing user and cannot:

1. start with sql, ibm or sys.

2. start with a numeral or contain characters other than a-z,$,#,@, or 0-9.

Action: Try the command again with a valid instance name.

DBI1003E

The file or directory "<file-name>" is missing.

Cause: A file or directory that is required to process this command is missing.

Action: If the file belongs to one of the database products then verify that the product is installed correctly and reinstall the product if necessary. If the file belongs to an instance then the instance may have already been removed or it may be corrupted.

DBI1004E

The file or directory "<file-name>" already exists;

Cause: A file or directory that the command needs to create already exists.

Action: Examine the specified file or directory. If the file or directory exists as a result of a previous successful completion of the command then no action is required. Otherwise, you will need to either rename or remove the specified file or directory before trying the command again.

DBI1005E

The message file "<file-name>" cannot be found.

Cause: The message file required by the instance commands is missing from the system; it may have been deleted or the database products may have been loaded incorrectly.

Action: Verify that the product option containing the message file is installed correctly. If there are verification errors; reinstall the product option.

DBI1006E

The file "<file-name>" could not be opened.

Cause: A system error occurred when attempting to open a file.

Action: Investigate and correct the cause of the failure and then try the command again.

DBI1007E

An attempt to write the file "<file-name>" failed.

Cause: A system error occurred when attempting to write to a file.

Action: Investigate and correct the cause of the failure and then try the command again.

DBI1008E   DBI1015E

DBI1008E

Usage: db2idrop instanceName [-f]

Cause: An incorrect argument was entered for the db2idrop command. The db2idrop command has one required argument instanceName and one optional argument -f.

Action: Enter the command again as follows:

```
db2idrop instanceName [-f]
  instanceName  is the name of the instance
  -f            is the force applications flag.
                If this flag is specified then all
                applications using the instance
                will be forced to terminate.
                (Applicable on AIX only.)
```

DBI1009E

The database manager configuration file could not be initialized.

Cause: An error occurred attempting to initialize the database manager configuration file. The Version 2.1 instance could not be created or migrated.

Action: Try the command again.

DBI1010E

The instance is active.

Cause: An attempt was made to drop an instance which is active.

Action: Make sure all applications that are using the instance have finished executing before stopping the instance by issuing a db2stop.

DBI1011E

Usage: db2imigrev instanceName [-f]

Cause: An incorrect argument was entered for the db2imigrev command. The db2imigrev command has one required argument instanceName and one optional argument -f.

Action: Enter the command again as follows:

```
db2imigrev instanceName [-f]
  instanceName  is the name of the instance which
                will be migrated backward from
                Version 2 to Version 1.
  -f            is the force applications flag.
                If this flag is specified then all
                applications using the instance
                will be forced to terminate.
                (Applicable on AIX only.)
```

DBI1012E

Usage: db2ilist

Cause: An incorrect argument was entered for the db2ilist command. The db2ilist command does not have any arguments.

Action: Enter the command again as follows:

```
db2ilist
```

DBI1013E

A File error occurred when accessing the node directory.

Cause: An open, read, write or close error occurred when accessing one of the files in the node directory. Possible causes of this error are: insufficient file permissions or ownerships or insufficient disk space.

Action: Resolve the file access problem and then try the command again.

DBI1014E

An error occurred removing the file or directory "<dfile-name>".

Cause: An attempt to remove a file or directory was unsuccessful.

Action: Adjust the file permissions of the specified file or directory and then issue the command again.

DBI1015E

An error occurred trying to create the file or directory "<dfile-name>".

Cause: An attempt to create a file or directory was unsuccessful.

Action: Check that there is enough space in the file system and that you have write permission for the required directory.

DBI1016E

The instance directory is locked by another command.

Cause: The command failed because the instance directory is locked by another instance command. The instance commands acquire an exclusive lock to the instance directory to avoid conflicting simultaneous commands against the same instance and contention updating the instance list for commands operating against different instances.

Action: Wait for all other occurrences of the instance commands to finish executing before trying the command again. If no other instance commands are running then delete the file instance.lock in the instance directory.

DBI1017E

Migration of the node directory failed.

Cause: A system error such as running out of disk space or memory has been encountered during the migration of the node directory. Instance migration failed.

Action: Make sure there is sufficient disk space and memory before trying the command again.

DBI1018E

The node directory is corrupted and cannot be migrated.

Cause: Either the node directory files are not the same or the node directory file is corrupted.

Action: You must either repair or remove the node directory from the sqllib directory before trying the command again.

DBI1019E

An error occurred moving the file or directory "<file-name>" to "<file-name>".

Cause: An error occurred attempting to backup a file or directory. The command could not be processed successfully.

Action: Determine why the specified file or directory could not be moved and then try the command again.

DBI1020E

One or more local databases have been detected that cannot be migrated. Check the logfile "<logfile-name>" for details.

Cause: If a catalogued local database is in any of the following states:

- backup pending
- roll-forward pending
- database inconsistent

or if the database has one or more objects that use either SYSCAT or SYSSTAT as the object qualified name then the database cannot be successfully migrated to Version 2.1.

Action: Check the file migration.log in the home directory of the instance for a list of all databases that cannot be migrated.

If a local database is in either the backup pending, the roll-forward pending or the database inconsistent states then you must take the appropriate actions to move all such databases into a consistent state.

If the database has one or more objects that use either SYSCAT or SYSSTAT as the object qualified name then all such objects must be dropped and recreated using a different object qualified name.

DBI1021E

The authType parameter "<authentication-type>" is not valid.

Cause: The authType parameter which specifies the authentication type used for the instance is not valid. Valid authentication types are: (SERVER, CLIENT and DCS). The default is SERVER.

Action: Try the command again with a valid authType parameter.

DBI1022E

The directory "<directory-name>" is not a Version "<version-number>" instance.

Cause: The command cannot proceed because the specified directory does not point to the required Version of the database products.

Action: The command cannot be run with the specified instance.

DBI1023E   DBI1029E

DBI1023E

The primary group "<group-name>" of the instanceName "<instance-name>" is invalid.

Cause: The primary group name of the instance cannot:

1. be one of guests, admins, users or local

2. start with sql or ibm.

3. start with a numeral or contain characters other than a-z,$,#,@, or 0-9.

Action: Change the primary group of the instance to a group with a valid name and then try the command again.

DBI1024E

Unexpected error.

Cause: An unexpected error occurred during the execution of one of the instance commands.

Action: Call service.

DBI1025E

The database manager could not be started.

Cause: The database manager could not be started when attempting to check if cataloged local databases can be migrated.

Action: Resolve why the database manager could not be started and then try the command again.

DBI1026E

Usage: db2iupdt instanceName [-u udfName]

Cause: An incorrect argument was entered for the db2iupdt command. The db2iupdt command has one required argument instanceName and the argument udfName.

Action: Enter the command again as follows:

```
db2iupdt instanceName [-u udfName]
  instanceName  is the name of the instance to
                be updated.
  -u udfName    is the name of the user under
                which fenced user defined
                functions will be run.
```

DBI1027E

The udfName parameter "<udf-name>" is not valid.

Cause: The udfName parameter which specifies the user name under which fenced user defined functions will be run:

• Must be the login name of an existing user.

• Cannot be set to the root or the bin user.

Action: Try the command again with a valid udfName parameter.

DBI1028E

Usage: db2imigr instanceName [-a authType] [-u udfName] [-f]

Cause: An incorrect argument was entered. The command accepts the required argument instanceName and the arguments authType, udfName and -f.

Action: Enter the command again as follows:

```
db2imigr instanceName [-a authType] [-u udfName]
[-f]
  instanceName  is the name of the instance
  -a authType   is the authentication type
                (SERVER, CLIENT or DCS) for the
                instance. The default is SERVER.
  -u udfName    is the name of the user under
                which fenced user defined
                functions will be run.
  -f            is the force applications flag.
                If this flag is specified then
                all applications using the
                instance will be forced to
                terminate. (Applicable on AIX
                only.)
```

DBI1029E

Applications are still using instance "<instance-name>".

Cause: There are applications that are still running that are using the specified instance. All applications using this instance must be terminated before the command can be completed successfully. You can get a list of the applications that are currently using the instance by issuing the command:

```
db2 list applications
```

Action: You can either wait for the applications to end by themselves, or you can explicitly force the applications to end. If you want to force all the

applications to end you can re-run the command with the "-f" option. Alternatively, you can logon as the instance owner and run the command

```
db2 force application all
```

Re-running the command with the "-f" option forces all applications to end and also stops the database manager.

```
┌── DBI1030E ─────────────────────────────────┐

  Product %s is required to run the db2help
  command.

└─────────────────────────────────────────────┘
```

Cause: The IPF/X runtime viewer must be installed on the workstation for you to run the db2help command.

Action: Install the IPF/X runtime viewer package from your installation media.

```
┌── DBI1031E ─────────────────────────────────┐

  Product %s should be installed to provide
  National Language support.

└─────────────────────────────────────────────┘
```

Cause: If you are operating in an non-English environment, you need to install the appropriate messages and resource files to provide national language support for the IPF/X runtime viewer. This includes language specific online help for using the IPF/X viewer.

Action: Install the appropriate IPF/X messages and resource support from your installation media.

```
┌── DBI1032E ─────────────────────────────────┐

  To get access to full set of online books,
  product %s should be installed.

└─────────────────────────────────────────────┘
```

Cause: Not all books are translated into each of the languages supported by DB2. If you wish to view a book that has not been translated, you should install the English set of books. This allows you to view the book in English. Once you have installed this option, re-issue the the db2help command. Follow the instructions there to set up your environment to view online books in multiple languages.

Action: Install the English set of books in additon to the set of online books specific to your language environment. Follow instructions from db2help and set

up your environment to view online books in multiple languages.

```
┌── DBI1033E ─────────────────────────────────┐

  Environment variable BOOKSHELF is not set.

└─────────────────────────────────────────────┘
```

Cause: This environment variable is used by the IPF/X viewer to locate the online books. db2help will not work properly if it has not been set.

Action: Set the BOOKSHELF environment to

```
/usr/lpp/db2_02_01/doc/locale
```

where locale is the language locales, and have the following values:

```
Languages      locale
----------     -------
Chinese        zh_CN
English        En_US
French         Fr_FR
German         De_DE
Italian        It_IT
Japanese       Ja_JP
Korean         ko_KR
Portuguese     pt_BR
Spanish        Es_ES
```

Re-issue the db2help command for further instructions in setting up your environment.

```
┌── DBI1034E ─────────────────────────────────┐

  Environment variable BOOKSHELF is not set.

└─────────────────────────────────────────────┘
```

Cause: The environment variable is used by the IPF/HP viewer to locate the online books. db2help will not work properly if it has not been set.

Action: Set the BOOKSHELF environment to

```
/opt/IBMdb2/V2.1/doc/%L
```

where %L is the language locales, and have the following values:

```
Languages      %L
----------     -------
English        en_US
French         fr_FR
German         de_DE
Italian        it_IT
Spanish        es_ES
```

Re-issue the db2help command for further instructions in setting up your environment.

DBI1035E   DBI1038E

DBI1035E

Environment variable BOOKSHELF is not set.

Cause: The environment variable is used by the
IPF/SUN viewer to locate the online books. db2help will
not work properly if it has not been set.

Action: Set the BOOKSHELF environment to

```
/opt/IBMdb2/V2.1/doc/%L
```

where %L is the language locales, and have the
following values:

```
Languages      %L
----------     -------
English        en_US
French         fr
German         de
Italian        it
Spanish        es
```

Re-issue the db2help command for further instructions in
setting up your environment.

DBI1036E

**Product %s is required to run the db2help
command.**

Cause: The IPF/X runtime viewer package must be
installed on the workstation for you to run the db2help
command.

Action: Install the IPF/X runtime viewer from your
installation media.

DBI1037E

**Product %s is required to run the db2help
command.**

Cause: The DB2 online document option for your
language environmnet must be installed on the
workstation for you to run the db2help command.

Action: Install the appropriate DB2 online document
option for your language environment from the
installation media.

DBI1038E

**National Language Support for %s is required to
run db2help.**

Cause: National Language support for the IPF/X viewer
is required to run db2help. DB2 for AIX is shipped with
ipfx.Runtime, which works on both AIX 3.2.5 and AIX
4.1.x systems. However, another version (ipfx.rte) is
shipped with AIX 4.1, and is a pre-requisite for other
softwares. Verify what is installed on the system before
installing the appropriate message support.

Action:

Issue the following command and view the output list:

```
lslpp -l | grep ipfx
```

1. If the resulting list includes "ipfx.rte", install the
   appropriate message option from your AIX 4.1
   installation media. The LPPs have names like
   ipfx.msg.locale.rte, where locale is the language
   locale you are using. For example, if you are using
   Fr_FR, install the ipfx.msg.Fr_FR.rte option. Also
   select ipfx.rte for an overwrite install.
2. If the resulting list shows only "ipfx.Runtime", install
   the appropriate message option from the DB2
   installation media. The LPPs have names like
   ipfx.nls.locale, where locale is the language locale
   you are using. For example, if you are using
   Fr_FR, install the ipfx.nls.Fr_FR option.
3. If the resulting list includes "ipfx.rte", but the
   message option LPPs for ipfx.rte are not available,
   then install the message support LPP from the DB2
   installation media. Also select ipfx.Runtime for an
   overwrite install to ensure that you have the proper
   viewer. This will, however, overwrite the ipfx.rte
   viewer and may cause problems for other
   applications that pre-requisite ipfx.rte.

For users in the ISO codepage environment, install the
corresponding IPF/X message option for the IBM-850
codepage as well. For example, if you are working in
de_DE, install also the ipfx.nls.De_DE option (if using
ipfx.Runtime) or the ipfx.msg.De_DE.rte option (if using
ipfx.rte).

DBI1039E

Product %s is required to run the db2help command.

Cause: The DB2 documentation pacakge must be installed for you to run the db2help command.

Action: Consult the installation manual and install the appropriate DB2 documentation package, then issue the db2help command again.

DBI1040E

Install online documents and set environment variable BOOKSHELF.

Cause: There is no DB2 online documentation currently installed.

Action: Install the appropriate online documentation option. The choices are:

```
Documentation Options (INF)  Locale
--------------------------   ------
db2_02_01.doc.zh_CN.ipfx     zh_CN
db2_02_01.doc.zh_TW.ipfx     zh_TW
db2_02_01.doc.ko_KR.ipfx     ko_KR
db2_02_01.doc.pt_BR.ipfx     pt_BR
db2_02_01.doc.Ja_JP.ipfx     Ja_JP
db2_02_01.doc.It_IT.ipfx     It_IT
db2_02_01.doc.Fr_FR.ipfx     Fr_FR
db2_02_01.doc.Es_ES.ipfx     Es_ES
db2_02_01.doc.En_US.ipfx     En_US
db2_02_01.doc.De_DE.ipfx     De_DE
```

Once installed, you need to set the BOOKSHELF environment variable to include the string /usr/lpp/db2_02_01/doc/Locale (where Locale is the appropriate value listed above).

DBI1041E

Environment variable BOOKSHELF has not been set properly for reading DB2 online documentation.

Cause: The Environment variable BOOKSHELF has not been set properly for the IPF viewer to locate DB2 online documentation.

Action: Find out what online documentation option has been installed and set the value of BOOKSHELF accordingly. If the option

```
db2_02_01.doc.locale.ipfx
```

is installed, the BOOKSHELF environment variable should include the string:

```
/usr/lpp/db2_02_01/doc/locale
```

Where locale is one of zh_CN, zh_TW, ko_KR, pt_BR, Ja_JP, It_IT, Fr_FR, Es_ES, En_US or De_DE.

DBI1044E

Usage: db2maint [-f]

Cause: An incorrect argument was entered for the db2maint command.

Action: Enter the command again as follows:

```
db2maint [-f]
   -f          is the force applications flag.
               If this flag is specified then all
               applications using all Version 2
               instances will be forced to terminate.
               (Applicable on AIX only.)
```

DBI1045E

Usage: db2mtoff

Cause: The db2mtoff command does not have any arguments.

Action: Enter the command again as follows:

```
db2mtoff
```

DBI1051E

Incorrect number of arguments for the db2icrt command.

Cause: The db2icrt command accepts the argument instanceName.

Action: Enter the command again as follows:

```
db2icrt instanceName
```

DBI1052E

The instanceName "<instance-name>" is invalid.

Cause: The instance name is invalid; the instance name must be an alphanumeric string that has eight characters or less.

Action: Try the command again with a valid instance name.

DBI1081E   DBI1106W

DBI1081E

The udfName parameter must be specified for DB2 Server instances.

Cause: If the DB2 Server product is installed then the udfName parameter must be specified. The udfName parameter is set to the name of the user under which fenced user defined functions and DARI functions will be run.

Action: Enter the command again adding the arguments "-u udfName" where udfName is the name of the user that you want your user defined functions or DARI functions to run under. For security reasons it is recommended that you do not use the instance name as the udfName. However, if you are not planning to use user defined functions or DARI functions then setting the udfName to the instance name will save you the step of having to create another user for the udfName.

DBI1101W

The link "<file-name>" was not created.

Cause: A link was not created because a file by the same name already exists.

Action: Rename the file and try the command again.

DBI1102W

The instance "<instance-name>" is already in the instance list.

Cause: The instance that is being created or migrated is already in the instance list.

Action: Verify that the list of instances as reported by the db2ilist command is correct.

DBI1103W

The instance list could not be updated.

Cause: The instance could not be added to or removed from the instance list.

Action: Check the file permissions on the instance list file /usr/lib/db2/V2.1/instance/db2idir.

DBI1104W

The instance "<instance-name>" was not found in the instance list.

Cause: The specified instance was not found in the instance list.

Action: Verify that the list of instances as reported by the db2ilist command is correct.

DBI1105W

The database manager configuration files could not be merged.

Cause: An error was detected when attempting to merge two database manager configuration files. The previous database manager configuration file could not be merged with the new database manager configuration file.

Action: Check the values in the new database manager configuration file and update the parameters as required.

DBI1106W

The authentication type of a cataloged local database has been changed.

Cause: One or more cataloged database entries were detected that have an authentication type that is different from the authentication type that was specified for the instance. If no action is taken then all cataloged local database entries will assume the authentication type of the instance they belong to.

Action: Check the file migration.log in the home directory of the instance for a list of the cataloged local database entries that are not the same as the authentication specified for the instance. If you want the database to maintain it's previous authentication type then you can either change the authentication type of the instance or you can move the database to another instance that has the desired authentication type. Before changing the authentication type of the instance, however, you should make sure that you want all the cataloged local database entries to have the new authentication type.

DBI1201N

The instance command was successful.

Cause: All processing completed normally.

Action: No further action is required.

Chapter 6. DB2 Messages

Each message has a message identifier that consists of a prefix (DB2) and a message number. Messages are listed in numeric sequence based on the message number. These messages are generated by the command line processor to alert you when a warning or error condition occurs. Messages are displayed directly on the screen interface. The command line processor returns DB2 and SQL messages.

For more information about SQL messages, see Chapter 7, "SQL Messages" in this guide.

DB20000I

The "<command>" command completed successfully.

Cause: The command identified in the message executed successfully.

The command line processor has successfully completed the command and is ready for the next command.

Action: Continue to the next command.

DB21001E

The option "<option-letter>" specified after the 'db2' command or in the DB2OPTIONS variable is incorrect.

Cause: The option specified is not supported. The supported options are:

```
Option  Description
------  -----------------------------------
  -a    Display SQLCA
  -c    Auto-commit
  -e    Display SQLCODE/SQLSTATE
  -f    Read from input file
  -l    Log commands in history file
  -o    Display output
  -p    Display db2 interactive prompt
  -r    Save output report to file
  -s    Stop execution on command error
  -t    Set statement termination character
```

```
  -v    Echo current command
  -w    Display FETCH/SELECT warning messages
  -z    Save all output to file
```

The command cannot be processed.

Action: Resubmit the command with a valid option.

DB21002E

The parameter for option "<option-letter>" specified after the 'db2' command or in the DB2OPTIONS variable is missing or incorrect.

Cause: The following lists the options with parameters:

```
Option        Description
------        ---------------------------------------
-ec           Display SQLCODE
-es           Display SQLSTATE
-f<filename>  Read from input file <filename>
-l<filename>  Log commands in history file <filename>
-r<filename>  Save output report to file <filename>
-td<x>        Set termination character to 'x'
-z<filename>  Save all output to file <filename>
```

Action: Resubmit the command with the valid option and parameter.

DB21003E

The value "<value>" in "<environment-variable>" is not valid.

Cause: The value for DB2BQTRY must be between 0 and 4294967295. The value for DB2BQTIME, DB2RQTIME or DB2IQTIME must be between 1 and 4294967295.

Action: Set the environment variable with the correct value and resubmit the command.

DB21004E

You cannot specify both an input file and a command when invoking the command line processor.

Cause: You cannot specify both the -f option and a command line command when invoking the command line processor.

Action: Correct the error and issue the command again.

DB21005E

An error occurred while accessing the file "<filename>".

Cause: The following could have caused the error:

- file permissions do not allow file access
- the file does not exist

Action: Correct the error and try again.

DB21006E

The input command is too long. The maximum length is "<length>".

Cause: The input command cannot exceed the length specified.

Action: Correct the error and resubmit the command.

DB21007E

End of file reached while reading the command.

Cause: The last command was not executed because end of file was reached. Terminate the last command with a ';' (or your defined termination character) if the -t option was used. Remove the '\' from the last line in the command if the +t option was used.

Action: Correct the error and resubmit the command.

DB21008E

Commands can only be entered from the command line processor interactive mode or file input mode.

Cause: The user attempted to enter a command line processor command from the DOS prompt.

Action: Use command line processor interactive mode or file input mode.

DB21010I

Help given for "<help-command-phrase>".

Cause: This message only appears in the history file specified with the -l option.

Action: No action required.

DB21015E

The command line processor back-end process request queue or input queue was not created within the timeout period.

Cause: Either the values for the DB2BQTRY and DB2BQTIME environment variables need to be increased or the command line processor back-end program "db2bp" cannot be started. The "db2bp" program must reside in the correct database manager install path and users must have execute permission on the file.

Action: Correct the error and resubmit the command.

DB21016E

The command line processor encountered a system error while sending the command to the back-end process.

Cause: One of the following could have happened:

- The backend process was abnormally terminated.
- A system error occurred when reading from or writing to the back-end process queues.
- A system error occurred when reading from the front-end process output queue.

Action: Resubmit the command. If the error occurs again, get help from your system administrator.

DB21017E

The command line processor encountered a system error with the front-end process output queue. Reason code = <reason code>.

Cause: A system error occurred while creating or reading from the front-end process output queue.

If the reason code is -2499, the command line processor output queue conflicts with an existing queue.

Action: Resubmit the command. If the error occurs again, record the message number and reason code, and get help from your system administrator.

DB21018E DB21027E

DB21018E

A system error occurred. Command Line Processor could not continue processing.

Cause: One of the following caused the system error:

- Command line processor did not successfully install its interrupt signal handler.

- Command line processor did not successfully open the back-end process queues.

- Command line processor did not successfully start up the back-end process.

- The back-end process was abnormally terminated.

- The front-end process did not successfully allocate or free memory.

- Command line processor received one of the following program termination signals:

    – SIGILL
    – SIGTRAP
    – SIGEMT
    – SIGBUS
    – SIGSEGV
    – SIGSYS

Action: Retry the command. If the problem persists, record the DB2 message number. If the trace was active, save the trace information and contact your technical support with the following information:

- Problem description
- DB2 message number
- SQLCA if possible
- Trace file if possible

DB21019E

An error occurred while accessing the directory "<directory>".

Cause: The following could have caused the error:

- directory permissions do not allow access

- the directory does not exist

Action: Correct the error and resubmit the command.

DB21024I

This command is asynchronous and may not be effective immediately.

Cause: This message is displayed after a FORCE command with the ASYNC clause.

Action: No action required.

DB21025I

Client changes will not be effective until the next time the application is started. Server changes will not be effective until the next db2start command.

Cause: Changes to the database manager configuration parameters become effective only after they are loaded into memory. This typically occurs after a **db2start** on a server and after the application restarts on a client.

Action: If necessary, ensure that all applications disconnect from the database. If this is not a client change, issue DB2STOP and DB2START.

DB21026I

All applications must disconnect from this database before the changes become effective.

Cause: The database configuration command successfully processed but the active database configuration cannot be changed until all applications have disconnected from the database. After the applications disconnect from the database, the first connection to the database will cause the changes to become effective.

Action: Ensure that all applications disconnect from the database and issue a db2 connect to command.

DB21027E

Isolation level may not be changed while connected to a database.

Cause: The user attempted to change the isolation level while connected to a database. The command is ignored.

Action: If a change in isolation level is necessary, disconnect from the current database, then set the isolation level and re-connect to the database.

DB21028E

The cursor "<cursor-name>" has not been declared.

Cause: The specified cursor needs to be declared before issuing an OPEN, FETCH, or CLOSE SQL statement.

Action: Declare the cursor and resubmit the command.

DB21029E

The cursor "<cursor-name>" has already been declared and opened.

Cause: The user attempted to declare a cursor that is open.

Action: Close the cursor and resubmit the open

DB21041W DB21051E

DB21041W

<number> indoubt transaction(s) not shown.

Cause: Command line processor could not display all the in-doubt transactions. The number of transactions not displayed is "<number>".

Action: Finish processing the current list of indoubt transactions in order to reduce the total number of indoubt transactions in the system. Next, reissue the list indoubt transactions command.

DB21042E

You must specify a transaction number.

Cause: A transaction number must be specified with the commit (c), rollback (r), or forget (f) in-doubt transaction subcommands.

Action: Reissue the command with the appropriate transaction number.

DB21043E

"<subcommand>" is not a valid request.

Cause: The in-doubt transaction subcommand specified is not valid. Valid subcommands are:

```
Subcommand    Description
----------    ----------------------------
c <number>    Heuristically commit the
              in-doubt transaction <number>.
r <number>    Heuristically rollback the
              in-doubt transaction <number>.
f <number>    Heuristically forget the
              in-doubt transaction <number>.
l <number>    List all in-doubt transactions
              or the in-doubt transaction
              <number>.
q             Exit LIST INDOUBT TRANSACTION
              prompting.
```

```
NOTE: The "f <number>" command is not available
in DB2 Parallel Edition.
```

Action: Correct the error and resubmit the command.

DB21044E

Cannot COMMIT transaction number "<number>".

Cause: The transaction must be in a prepared state (i) when you issue the commit (c) subcommand.

Action: Issue another command.

DB21045E

Cannot ROLLBACK transaction number "<number>".

Cause: The transaction must be in a prepared (i) or idle (e) state. when you issue the rollback (r) subcommand.

Action: Issue another command.

DB21046E

Cannot FORGET transaction number "<number>".

Cause: The transaction must be in a heuristically committed (c) or heuristically rolled-back (r) state when you issue the forget (f) command.

Action: Issue another command.

DB21050E

"<state>" is not a valid SQLSTATE.

Cause: The sqlstate specified is not valid or could not be found. Valid states are numerical and two or five digits in length.

Action: Resubmit the command using a different state.

DB21051E

The command is not supported for this environment.

Cause: The command requested is currently not supported by the command line processor in the environment being used.

Action: Resubmit the command on a different platform or within a different environment.

```
DB21052I
```

"<command>" has been submitted for execution.

Cause: The command line processor is attempting to execute the indicated command. It will be unable to determine whether the command has completed successfully.

One such command is "xview SQLS00AA.INF", which depends on the installation of xview, the availability of the SQLS00AA.INF on-line book, the proper setting of the BOOKSHELF environment parameter and the type of connection you have to the database.

Action: If the command appears unsuccessful, submit it from outside the command line processor.

```
DB21053W
```

Automatic escalation will occur when you connect to a database that does not support <isolation-level>.

Cause: Isolation levels are database dependent. Some, like NC, are only supported by specific databases. If you select an isolation level that is not supported by the database you are connecting to, it will automatically escalate to a supported level.

Action: Connect to a database that supports the isolation level you have selected, or select a different isolation level.

```
DB21054E
```

The command line processor encountered a system error with the <command> command. Reason code = <reason code>.

Cause: The command line processor was unable to get enough memory to process your command. A system error occurred while processing it.

```
Reason
code   Description    Syntax
------ -------------- --------------------------
1      Data file      LOAD FROM file/pipe/dev ...
2      Lob data       LOBS FROM lob-path ...
3      Directory      USING directory ...
4      Source         FROM dir/dev ...
5      Target         TO dir/dev ...
6      Tablespace     TABLESPACE tblspace-name ...
7      Tblspace-def   MANAGED BY ...
8      Container data USING (PATH ..., PATH ...)
                                      or
                      USING ({FILE|DEVICE} ...,
```

```
                      {FILE|DEVICE}...)
```

Action: Correct the error and resubmit the command.

```
DB21055W
```

Command not valid for type 2 connections.

Cause: The information returned by the GET CONNECTION STATE command does not apply to type 2 connections.

Action: Issue QUERY CLIENT to verify CONNECT = 1.

```
DB21056W
```

Directory changes may not be effective until the directory cache is refreshed.

Cause: If directory caching (DBM CFG dir_cache) is enabled, database, node, and DCS directory files are cached in memory. Directory changes may not become effective until the directory cache is refreshed. Refer to the dir_cache configuration parameter in the ADMIN guide for a description on directory caching.

Action: To refresh CLP's directory cache, issue a db2 TERMINATE. To refresh the directory information for another application, stop and restart that application. To refresh the directory information for the database, stop (db2stop) and restart (db2start) the database.

```
DB21057E
```

Invalid tape device specified.

Cause: The tape device passed to the operating system was not accepted. On Windows NT it must be of the form "\\.\TAPEx" where x represents the drive number (0 is the first).

Action: Reissue the command specifying a valid tape device.

```
DB21058E
```

Invalid tape position specified.

Cause: The tape mark position specified is not valid. On Windows NT the first tape position to which a backup is record is 1. Subsequent backup images begin at tape mark 2 and so on.

Action: Reissue the command specifying a valid tape position.

DB21059E   DB21080E

DB21059E

Invalid tape blocksize specified.

Cause: The tape mark blocksize specified is not in the range supported by the tape device. In addition, for backup/restore to work it must also be a factor of or multiple of 4096.

Action: Reissue the command specifying a valid tape size.

DB21060E

General tape failure.

Cause: An unexpected return code was returned from the tape operation.

Action: Resubmit the command. If the error occurs again, get help from your system administrator.

DB21061E

Command line environment not initialized.

Cause: You have tried to invoke the command line processor from a command window that was not started by db2cmd.exe.

Action: Issue DB2CMD to start a command window that has the command line processor environment initialized.

DB21080E

No previous RESTORE DATABASE command with REDIRECT option was issued for this database alias, or the information about that command is lost.

Cause: You have tried to run the RESTORE DATABASE command with the CONTINUE or ABORT option. However, either you didn't previously issue a RESTORE DATABASE command with the REDIRECT option, or you did issue such a command but for a different database alias than the one specified on the failing command. Another possible cause for this message could be that the information about a correctly issued previous RESTORE DATABASE ... REDIRECT command was lost. This can happen if the CLP back-end process terminates abnormally, or if you issue the TERMINATE command.

Action: Restart the whole redirected restore process by issuing the RESTORE DATABASE ... REDIRECT command and SET TABLESPACE CONTAINERS commands again. Then issue the RESTORE DATABASE ... CONTINUE command.

Chapter 7. SQL Messages

Each message has a message identifier that consists of a prefix (SQL) and a message number. Messages are listed in numeric sequence based upon the message number. There are three message types: notification, warning, and critical. Message identifiers ending with an *N* are error messages. Those ending with a *W* indicate warning or informational messages. Message identifiers ending with a *C* indicate critical system errors.

The message number is also referred to as the *SQLCODE*. The SQLCODE is passed to the application as a positive or negative number, depending on its message type (N, W, or C). N and C yield negative values whereas W yields a positive value. DB2 returns the SQLCODE to the application and the application can get the message associated with the SQLCODE. DB2 also returns an *SQLSTATE* value for conditions that could be the result of an SQL statement. SQLSTATE values are listed in Chapter 9, "SQLSTATE Messages." Some SQLCODE values have associated SQLSTATE values. SQLSTATEs associated with a given SQLCODE (if applicable) have been documented with each message in this chapter.

Variable parameters for SQL messages are shown as descriptive names.

```
┌─ SQL0000W ──────────────────────────────────┐
│  Statement processing was successful.        │
└──────────────────────────────────────────────┘
```

Cause: The SQL statement executed successfully, unless a warning condition occurred.

Action: Check SQLWARN0 to ensure that it is blank. If it is blank, the statement executed successfully. If it is not blank, a warning condition exists. Check the other warning indicators to determine the particular warning condition. For example, if SQLWARN1 is not blank, a string was truncated.

Refer to the *API Reference*.

sqlcode: 0

sqlstate: 00000, 01003, 01004, 01503, 01504, 01506, 1509, 01517

```
┌─ SQL0001N ──────────────────────────────────┐
│  Binding or precompilation did not complete  │
│  successfully.                               │
└──────────────────────────────────────────────┘
```

Cause: For reasons specified in the previous messages, the bind or precompile request was unsuccessful.

No package is created.

Action: Refer to the messages in the message file. Resubmit the command.

If installing the sample database, drop it and install the sample database again.

```
┌─ SQL0002N ──────────────────────────────────┐
│  The bind file name is not valid.            │
└──────────────────────────────────────────────┘
```

Cause: For reasons specified in the previous messages, the bind file name cannot be used as specified.

No package is created.

Action: Refer to the messages in the message file. Resubmit the command.

If installing the sample database, drop it and install the sample database again.

```
┌─ SQL0003N ──────────────────────────────────┐
│  The database name is not valid.             │
└──────────────────────────────────────────────┘
```

Cause: For reasons specified in the previous messages, the database name cannot be used as specified.

No package is created.

Action: Refer to the messages in the message file. Resubmit the command.

SQL0004N   SQL0011N

SQL0004N

The password is not valid.

Cause: The password contains characters that are not valid or the password is too long.

No package is created.

Action: Resubmit the command with a valid password.

SQL0005N

The message file name is not valid.

Cause: For reasons specified in the previous messages, the message file name cannot be used as specified.

No package is created.

Action: Refer to the messages in the message file. Check the name of the message file. Check the attributes of the message file if it exists. Resubmit the command.

SQL0006N

The datetime format parameter is not valid.

Cause: The value of the datetime format parameter is not in the valid range of 0 through 3.

No package is created.

Action: Resubmit the command with a valid format parameter.

SQL0007N

The character "<character>" following "<text>" is not valid.

Cause: The specified "<character>" is not a valid character in SQL statements. The "<text>" field indicates the 20 characters of the SQL statement that preceded the character that is not valid.

DataJoiner users: some DataJoiner data sources do not provide the appropriate values for the "<character>" and "<text>" message tokens. In these cases, "<character>" and "<text>" will have the following format: "<data source>:UNKNOWN", indicating that the actual values for the specified data source are unknown.

The statement cannot be processed.

Action: Remove or replace the character that is not valid.

sqlcode: -7

sqlstate: 42601

SQL0008N

The token "<token>" found in a host variable declaration is not valid.

Cause: A host variable declaration has invalid syntax. The precompiler cannot identify the host variable.

The statement cannot be processed. None of the host variables declared in the statement (up to the semicolon) are accepted.

Action: Examine the syntax of the host variable declaration.

SQL0009W

An attempt to override a precompiler option was ignored.

Cause: An attempt was made to override a precompiler option.

The option is ignored.

Action: Ensure that all precompiler options are specified correctly.

SQL0010N

The string constant beginning with "<string>" does not have an ending string delimiter.

Cause: The statement contains a string constant, beginning with "<string>", that is not terminated properly.

The statement cannot be processed.

Action: Examine the statement for missing apostrophes in the indicated string constant.

sqlcode: -10

sqlstate: 42603

SQL0011N

The comment is not terminated.

Cause: The comment is not terminated properly.

The statement cannot be processed.

Action: Examine the statement for a missing or extra comment delimiter in the indicated comment.

SQL0012W

Correlation without qualification has occurred for the column "<column>".

Cause: The named column occurs in a SELECT statement, is not explicitly qualified, and occurs in a table specified in the FROM clause of an outer select. Consequently, the reference to the column in the SELECT statement has been taken to be an outer reference and correlation will occur.

The statement was processed under the assumption that correlation was intended.

Action: Ensure that correlation was intended. It is a good practice to explicitly qualify intended outer references.

sqlcode: +12

sqlstate: 01545

SQL0013N

An empty delimited identifier is not valid.

Cause: During precompilation, either a cursor name, statement name, database name, or authorization ID, specified as an empty string, was found. This is not valid. The statement cannot be processed.

Action: Provide a valid cursor name, statement name, database name, or authorization ID.

SQL0014N

The source file name is not valid.

Cause: The source file name specified in the call to the precompiler contains invalid characters or the pointer to the source file name is invalid.

No package was created.

Action: Ensure that the source file name is correct.

SQL0015N

The host variable data type "<token_1>" is not valid. Use "<token_2>" instead.

Cause: If the WCHARTYPE CONVERT precompile option is in effect then the graphic host variable must be declared with the data type 'wchar_t', not 'sqldbchar'.

If the WCHARTYPE NOCONVERT precompile option is in effect (the default) and 'wchar_t' is defined as a 4-byte integer on this platform, then the graphic host variable

must be declared with the data type 'sqldbchar', not 'wchar_t'.

Action: Replace the current data type of the host variable with the data type specified in the message.

SQL0020W

Bind or precompile option(s) "<option-name(s)/option-number(s)>" are not supported by the target database and will be ignored.

Cause: One or more options that were specified at precompile/bind time are not supported by the target DBMS. The unsupported options will be ignored. The most probable cause is that an option which is only supported while connected to a DRDA application server has been specified while connected to a non-DRDA server.

Enter "db2 ? bind" or "db2 ? prep" for a list of which options are supported only by a DRDA application server.

Possible option numbers and their corresponding option names include:

1   DATETIME
2   LANGLEVEL
4   ISOLATION
5   BLOCKING
6   GRANT
8   SQLFLAG
16 CONNECT
17 SQLRULES
18 DISCONNECT
19 SYNCPOINT
20 ISOLATION
21 BINDFILE
22 SQLCA
23 PACKAGE
24 OPTLEVEL
25 SYNTAX or SQLERROR
26 NOLINEMACRO
30 LEVEL
31 COLLECTION
32 VERSION
33 OWNER
34 QUALIFIER
35 TEXT
40 VALIDATE
41 EXPLAIN
42 ACTION
44 REPLVER
45 RETAIN
46 RELEASE

SQL0021W SQL0028C

47 DEGREE
50 STRDEL
51 DECDEL
55 CHARSUB
56 CCSIDS
57 CCSIDM
58 CCSIDG
59 DEC
60 WCHARTYPE
61 DYNAMICRULES
62 INSERT
63 EXPLSNAP
64 FUNCPATH
65 SQLWARN
66 QUERYOPT

Action: Ensure the specification of bind or precompile "<option-name(s)/option-number(s)>" was intended while connected to this DBMS.

SQL0021W

Invalid precompiler option "<option>" is ignored.

Cause: The option specified in the message is not a valid precompiler option.

The option is ignored.

Action: Ensure that all precompiler options are specified correctly.

SQL0022W

Duplicate precompiler option "<option>" is ignored.

Cause: The precompiler option "<option>" is a duplicate.

The option is ignored.

Action: Ensure that all precompiler options are specified only once.

SQL0023N

The database name is not valid.

Cause: The specified database name is not a valid name.

Precompilation is terminated.

Action: Ensure that the database name is spelled correctly and follows the rules for short identifiers.

SQL0024N

A database name was not specified.

Cause: No database name was provided to proceed with precompilation.

Precompilation is terminated.

Action: Provide a database name.

SQL0025W

Binding or precompilation completed with warnings.

Cause: Binding or precompilation succeeded, but warnings were issued. A package and/or bindfile was created as requested in the command.

Action: Refer to the messages in the message file. If necessary, correct the problems and resubmit the command.

SQL0026N

The password is not valid.

Cause: The specified password is not a valid password.

Precompilation is terminated.

Action: Ensure that the password provided follows the rules for valid passwords.

SQL0028C

The release number of the bind file is not valid.

Cause: The release number of the bind file is not compatible with the release number of the installed version of the database manager.

The bind file cannot be used with the current version of the database manager. The command cannot be processed.

Action: If possible, repeat the precompile process with the current database manager. Otherwise, use only bind files created with a compatible release level of the database manager.

SQL0029N

INTO clause required.

Cause: Non-cursor SELECT or VALUES statements embedded in an application program must have an INTO clause to denote where the results of the statement are to be placed. Dynamic SELECT statements do not permit the INTO clause.

Action: Add the INTO clause to the SELECT or VALUES statement and precompile the application program again.

sqlcode: -29

sqlstate: 42601

SQL0030N

A source file name was not specified.

Cause: A source file name was not provided to proceed with precompilation.

Precompilation is terminated.

Action: Specify a source file name.

SQL0031C

File "<name>" could not be opened.

Cause: The file "<name>" is required but could not be opened.

Precompilation is terminated.

Action: Ensure that the file name specified is correct and exists in the file system, and that the file permissions are correct.

If installing the sample database, drop it and install the sample database again. If the error continues, reinstall the database manager and then install the sample database.

SQL0032C

Unable to use file "<name>".

Cause: While reading or writing file "<name>", an error was encountered.

Precompilation is terminated.

Action: Retry the precompilation.

SQL0033N

"<name>" is not a valid bind file.

Cause: The bind file "<name>" specified is not a bind file.

Binding is terminated.

Action: Ensure that the correct file name was specified.

SQL0034N

A bind file name was not specified.

Cause: No bind file name was specified to proceed with binding.

Binding is terminated.

Action: Provide a bind file name.

SQL0035N

The file "<name>" cannot be opened.

Cause: The message file "<name>" could not be opened.

Binding or precompilation has been terminated.

Action: Ensure that the system can access the file.

SQL0036N

The syntax of the file name "<name>" is not valid.

Cause: If the file is input to the precompiler, it must have an extension with the correct extension for the language being used. If the file is input to the binder, it must have an extension of *.bnd*. Also, a fully resolved file name which is over the platform maximum length will cause this error.

Precompilation or binding is terminated.

Action: Ensure that the specified file name is correct.

SQL0037W SQL0051N

SQL0037W

The syntax of the message file "<name>" is not valid.

Cause: The message file name "<name>" is not syntactically correct for this function.

The system redirects the output to the standard output device.

Action: Ensure that the specified file name is correct.

SQL0038W

The bind option SQLERROR CONTINUE has been activated since it is required when binding this DB2-supplied list file to DB2/MVS, SQL/DS, or OS/400.

Cause: The SQLERROR CONTINUE bind option is required when binding the following DB2-supplied list files:

- ddcsmvs.lst
- ddcsvm.lst
- ddcsvse.lst
- ddcs400.lst

This option instructs a DRDA server to create a package even if there are SQL statements in a bind file which it considers invalid. Since not all of the DRDA servers support all of the SQL statements contained in the DB2-supplied bind files, the SQLERROR CONTINUE bind option must be used to guarantee that packages are created for all of the bind files in the list file.

Action: No response is necessary. The required bind option, SQLERROR CONTINUE, has been specified for you. To avoid receiving this warning in the future specify the SQLERROR CONTINUE bind option.

SQL0039N

The binder did not finish processing because the bind file is not valid.

Cause: The binder could not process the bind file. The bind file may not be valid because the bind file contents were altered accidentally.

The bind file is not processed.

Action: If possible, repeat the precompile process to create a new bind file. Otherwise, obtain a new copy of the bind file.

SQL0040N

An error occurred on one or more bind files in the list "<name>". The following files were not bound: "<list>".

Cause: For reasons specified in the previous messages in the message file, one or more bind files were not bound. The list of files that were not bound is composed of digits, starting from 1, that indicate the relative position of the unbound files in the list file. "<name>" does not include the path specification of the list file.

Only the first 20 bind files in error are listed in the message. If there were more than 20 bind files in error, ellipses (...) are inserted after the last bind file name in the list.

One or more packages were not created.

Action: Refer to the messages in the message file. Check the list file to verify that it contains valid names. Resubmit the command.

SQL0041N

No attempt was made to bind the files following file number "<number>" in list "<name>" because a fatal error occurred which terminated processing.

Cause: Some errors encountered during the binding process are considered fatal (i.e. system errors, memory errors). If one of these errors occurs while processing a file in a list file, processing terminates. No attempt is made to bind the remaining files in the list file.

Such an error occurred while binding the specified bind file in the list. Note that the digit used to identify the bind file indicates the relative position of the file in the list file.

Action: Refer to the other messages which accompanied this one to resolve the errors which occurred. Resubmit the command.

SQL0051N

The maximum number of SQL statements was exceeded in the program.

Cause: Too many SQL statements are in the program to fit into one package in the database.

Precompilation is terminated.

Action: Either simplify the program, or split the program into smaller, separate programs, or do both.

SQL0053W

No SQL statements were found in the program.

Cause: The specified source file contains no SQL statements.

If binding, an empty package is created.

Action: Ensure that you are precompiling or binding the correct program.

SQL0055N

Source input file is empty.

Cause: The program source input file contained no data.

Precompilation is terminated.

Action: Ensure that the correct input file was specified.

SQL0060W

The "<name>" precompiler is in progress.

Cause: This message is written to the standard output device when the precompiler begins processing. The token "<name>" refers to the particular language precompiler which was invoked.

Processing continues.

Action: No action is required.

SQL0061W

The binder is in progress.

Cause: This message is written to the standard output device when the binder begins processing.

Processing continues.

Action: No action is required.

SQL0062W

Starting INCLUDE of file "<name>".

Cause: An INCLUDE statement has been encountered. The precompiler is now processing the INCLUDE file.

Action: No action is required.

SQL0063W

Completed INCLUDE of file "<name>".

Cause: The precompiler has completed processing the INCLUDE file. It will resume processing the file which contained the INCLUDE statement.

Action: No action is required.

SQL0064N

File "<name>" directly or indirectly INCLUDEs itself.

Cause: A cyclic INCLUDE has been encountered. A precompiler input file cannot INCLUDE itself, nor be INCLUDEd by a file it INCLUDEs.

The specified file is not INCLUDEd.

Action: Check the nesting of the INCLUDE files, and remove any cycles.

SQL0065N

An unexpected end-of-line was encountered in a host variable declaration.

Cause: A host variable declaration has an invalid syntax. An end-of-line was encountered before the declaration was complete.

The statement cannot be processed.

Action: Examine the syntax of the host variable declaration.

SQL0081N

SQLCODE "<sqlcode>" returned during precompilation/bind.

Cause: While precompiling or binding the program, an unexpected SQLCODE "<sqlcode>" was returned from the database manager.

The statement cannot be processed.

Action: Inspect the SQLCODE to determine the problem and take the appropriate action.

SQL0082C   SQL0089N

SQL0082C

An error has occurred which has terminated processing.

Cause: A previous, non-SQL error has caused processing to terminate.

Precompile/bind/rebind is terminated. No package was created.

Action: Correct the situation that caused the previous error and retry the operation.

SQL0083C

A memory allocation error has occurred.

Cause: During processing, there was not enough memory to continue processing.

Action: Possible solutions include:

- Ensure that your system has sufficient real and virtual memory.
- Remove background processes.

SQL0084N

An EXECUTE IMMEDIATE statement contains a SELECT or VALUES statement.

Cause: A SELECT or VALUES statement was used in an EXECUTE IMMEDIATE statement.

The statement cannot be processed.

Action: The implied function is not supported. Prepare the SELECT or VALUES statement. Then use OPEN, FETCH, and CLOSE.

sqlcode: -84

sqlstate: 42612

SQL0085N

The statement name "<name>" is already defined.

Cause: A previous DECLARE statement already used the statement name specified in the current DECLARE statement.

The current DECLARE statement is not processed. The previous DECLARE statement retains use of the statement name.

Action: Use a different name for the statement.

SQL0086C

A memory allocation error has occurred.

Cause: During processing, there was not enough memory to continue processing.

Action: Possible solutions include:

- Verify that your system has the required memory.
- Remove background processes.

SQL0087N

Host variable "<name>" is a structure used where structure references are not permitted.

Cause: When a structure reference is used in an SQL statement, it is treated as if a comma-separated list of its component fields had been used instead. Since a list of host variables is not permitted in SQL statements such as PREPARE, neither is a reference to a structure with multiple fields.

Action: Replace the structure reference with an atomic host variable, or with a fully qualified structure field name.

SQL0088N

Host variable "<name>" is ambiguous.

Cause: A field which occurs in multiple host structures has been used in an SQL statement without qualification. The reference is ambiguous.

Action: Qualify the field reference.

SQL0089N

Processing ended after 100 errors were found.

Cause: The precompiler or binder quits processing after 100 errors were found.

Action: Fix the errors indicated in the message log and resubmit the command.

SQL0091W

Precompilation or binding was ended with "<number-1>" errors and "<number-2>" warnings.

Cause: The precompilation or binding has terminated with the stated number of warnings and errors.

Precompilation or binding is terminated.

Action: If warnings or errors occurred, correct the program, if necessary, and retry the precompilation or binding.

SQL0092N

No package was created because of previous errors.

Cause: Because of previous errors, no package was created.

Action: Correct the errors and retry the precompilation or bind.

SQL0093N

The end-of-input was encountered before the statement terminator for EXEC SQL.

Cause: While processing an SQL statement, the end of source was encountered before the statement terminated.

Precompilation is terminated.

Action: Ensure that the SQL statement terminates properly.

SQL0094N

Binding was ended because of user interrupt request.

Cause: Binding was terminated because the user may have pressed the interrupt key sequence.

Processing is terminated. No package is created.

Action: Resubmit the bind, if needed.

If installing the sample database, drop it and install the sample database again.

SQL0095N

No bind file was created because of previous errors.

Cause: Because of previous errors, no bind file was created.

No bind file is created.

Action: Correct the errors and retry the precompilation.

SQL0100W

No row was found for FETCH, UPDATE or DELETE; or the result of a query is an empty table.

Cause: One of the following conditions is true:

- No row was found that meets the search conditions specified in an UPDATE or DELETE statement.
- The result of a SELECT statement was an empty table.
- A FETCH statement was executed when the cursor was positioned after the last row of the result table.
- The result of the SELECT used in an INSERT statement is empty.

No data was retrieved, updated, or deleted.

Action: No action is required. Processing can continue.

sqlcode: +100

sqlstate: 02000

SQL0101N

The statement is too long or too complex.

Cause: The statement could not be processed because it exceeds a system limit for either length or complexity, or because too many constraints or triggers are involved.

If the statement is one that creates or modifies a packed description, the new packed description may be too large for its corresponding column in the system catalogs.

DataJoiner users should also check to see if the statement:

- Exceeds either a DataJoiner system limit or a data source system limit for length or complexity.
- Violates some other data source specific limit.

The statement cannot be processed.

Note: Where character data conversions are performed for applications and databases running under

SQL0102N   SQL0103N

different codepages, the result of the conversion is exceeding the length limit.

Action: Either:

- Break the statement up into shorter or less complex SQL statements.
- Increase the size of the statement heap (stmtheap) in the database configuration file.
- Reduce the number of check or referential constraints involved in the statement or reduce the number of indexes on foreign keys.
- Reduce the number of triggers involved in the statement.
- DataJoiner users: determine which data source is failing the statement (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and determine the cause of the rejection. If the rejection is coming from DataJoiner, increase the size of the statement heap (stmtheap) in the database configuration file.

sqlcode: -101

sqlstate: 54001

SQL0102N

The string constant beginning with "<string>" is too long.

Cause: One of the following has occurred:

- The comment in the COMMENT ON statement is greater than 254 bytes.

- The application server name specified in the SQL CONNECT statement is greater than 18 characters.

- The string constant beginning with "<string>" has a length greater than 4000 bytes. Character strings with lengths greater than 4000 bytes or graphic strings with lengths greater than 2000 characters can be specified only through assignment from host variables.

- DataJoiner users: the data source-specific limits must not be exceeded in a pass-through session. For example, a character literal larger than 254 bytes included in a statement sent to DB2 for MVS in a pass-through session would cause this error.

It may be that data conversion is occurring and the resultant string is too long. For connections between applications and databases running under different codepages, string constants are converted from the application codepage to the database codepage. In certain situations, graphic string constants may be further converted from the database codepage to a UCS-2 (UNICODE) encoding, for example when the database has been created with an EUC codepage. This means that it is possible to have a resultant string which is longer than the input string.

The statement cannot be processed.

Action: For Comment on Table or Comment on Column, reduce the size of the comment. For the SQL CONNECT statement, reduce the length of the application server name. For other string constants, the requested function is not available interactively. For an error occurring in the context of a non-CONNECT SQL statement embedded in an application program, assign the long string to a host variable and substitute that variable for the string literal in the SQL statement.

DataJoiner users: for a pass-through session, determine what data source is causing the error (see the *Problem Determination Guide* for the failing data sources). Examine the SQL dialect for that data source to determine which specific limit has been exceeded, and adjust the failing statement as needed.

sqlcode: -102

sqlstate: 54002

SQL0103N

The numeric literal "<literal>" is not valid.

Cause: The indicated "<literal>" begins with a digit but is not a valid integer, decimal, or floating point literal.

DataJoiner users: a data source-specific literal representation error has occurred in a pass-through session.

The statement cannot be processed.

Action: Correct the invalid numeric literal. DataJoiner users, if the error occurred in a pass-through session, determine what data source is causing the error (see the *Problem Determination Guide* for the failing data sources). Examine the SQL dialect for that data source to determine which literal representation rule has been violated, and adjust the failing statement as needed.

sqlcode: -103

sqlstate: 42604

SQL0104N

An unexpected token "<token>" was found following "<text>". Expected tokens may include: "<token-list>".

Cause: A syntax error in the SQL statement was detected at the specified token following the text "<text>". The "<text>" field indicates the 20 characters of the SQL statement that preceded the token that is not valid.

As an aid to the programmer, a partial list of valid tokens is provided in the SQLERRM field of the SQLCA as "<token-list>". This list assumes the statement is correct to that point.

The statement cannot be processed.

Action: Examine and correct the statement in the area of the specified token.

sqlcode: -104

sqlstate: 42601

SQL0105N

The string constant beginning with "<string>" is not valid.

Cause: The statement contains a string constant beginning with "<string>" that is not valid.

The statement cannot be processed.

Action: Specify the correct format of the string constant. Check for graphic string, paired delimiters, and an even number of bytes within the string.

DataJoiner users, see the *Problem Determination Guide* to determine which data source is causing the error.

sqlcode: -105

sqlstate: 42604

SQL0107N

The name "<name>" is too long. The maximum length is "<length>".

Cause: The name returned as "<name>" is too long. The maximum length permitted for names of that type is indicated by "<length>".

The names for columns, tables, views, indexes, aliases, and constraints can be a maximum length of 18 bytes. (This does not include any escape characters, if present.)

A maximum of 8 bytes is permitted for a schema name (object qualifier).

Host variable names must not exceed 30 bytes in length.

For the SQL CONNECT statement, an application server name of up to 18 characters in length will be accepted at pre-compilation time. However, at runtime, an application server name which is greater than 8 characters in length will cause an error.

Also, a password of up to 18 characters in length and an authorization ID of up to 8 characters in length will be accepted in the SQL CONNECT statement.

DataJoiner users: if in a pass-through session, a data source-specific limit might have been exceeded.

The statement cannot be processed.

Note: Where character data conversions are performed for applications and databases running under different codepages, the result of the conversion is exceeding the length limit.

Action: Choose a shorter name or correct the spelling of the object name.

DataJoiner users: for a pass-through session, determine what data source is causing the error (see the *Problem Determination Guide* for the failing data sources). Examine the SQL dialect for that data source to determine which specific limit has been exceeded, and adjust the failing statement as needed.

sqlcode: -107

sqlstate: 42622

SQL0108N

The name "<name>" has the wrong number of qualifiers.

Cause: The name "<name>" is improperly qualified.

The object given the name "<name>" can only have one qualifier.

A column name is qualified with a table name, which is either qualified or unqualified, or a correlation name. In some contexts, a column name requires a table name qualifier.

The statement cannot be processed.

Action: Ensure that the name for the object is qualified correctly.

sqlcode: -108

sqlstate: 42601

SQL0109N

The "<clause>" clause is not allowed.

Cause: The indicated clause is not allowed in the context where it appears in the SQL statement.

A subquery, an INSERT statement, or a CREATE VIEW statement cannot have INTO, ORDER BY, or FOR UPDATE clauses. An embedded SELECT statement cannot have ORDER BY or FOR UPDATE clauses. An embedded SELECT statement cannot contain a set operator except in a subquery. SELECT or VALUES statements used in cursor declarations cannot have an INTO clause. A RAISE_ERROR function can only be used as a select list item if it is cast to some data type using the CAST specification.

DataJoiner users: the indicated clause may be missing and may be required. In a pass-through session, a data source-specific restriction might have been violated.

The statement cannot be processed.

Action: Correct the SQL statement by removing the clause.

DataJoiner users: correct the SQL statement by removing or adding the clause. For a pass-through session, determine what data source is causing the error (see the *Problem Determination Guide* for the failing data sources). Examine the SQL dialect for that data source to determine which specific restriction has been violated, and adjust the failing statement as needed.

sqlcode: -109

sqlstate: 42601

SQL0110N

"<string>" is an invalid hexadecimal constant.

Cause: The hexadecimal constant "<string>" is invalid. The problem is one of the following:

- An invalid hexadecimal digit was specified. Only '0 to 9', 'A to F', and 'a to f' are allowed.

- An uneven number of hexadecimal digits was specified.

- More than 8000 hexadecimal digits were specified.

Action: Correct the constant and resubmit the statement.

sqlcode: -110

sqlstate: 42606

SQL0111N

The column function "<name>" does not include a column name.

Cause: The specification of the column function "<name>" (AVG, MIN, MAX, SUM, or COUNT(DISTINCT)) was not specified correctly because such functions must include a column name in the operand.

The statement cannot be processed.

Action: Specify a column name in the expression that is the operand to the column function.

NOTE: This error is only applicable to releases of DB2 prior to Version 2.

sqlcode: -111

sqlstate: 42901

SQL0112N

The operand of the column function "<name>" is another column function or a scalar fullselect, or includes a subquery.

Cause: Only expressions without column functions or scalar fullselects or that do not include a subquery are permitted as operands of a column function.

In a SELECT list, the operand of an arithmetic operator cannot be a column function that includes the DISTINCT keyword.

The statement cannot be processed.

Action: Correct the use of the column function to eliminate the invalid expression and try again.

sqlcode: -112

sqlstate: 42607

SQL0117N

The number of values assigned is not the same as the number of specified or implied columns.

Cause:

- The number of insert values in the value list of the INSERT statement is not the same as the number of columns specified or implied. If no column list is specified, a column list that includes all columns of the table or view is implied.

- The number of values on the right hand side of an assignment in a SET transition-variable statement or

SET clause of an UPDATE statement does not match the number of columns on the left hand side.

The statement cannot be processed.

Action: Correct the statement to specify one value for each of the specified or implied columns.

sqlcode: -117

sqlstate: 42802

SQL0118N

The table or view that is the target of the INSERT, DELETE, or UPDATE statement is also specified in a FROM clause.

Cause: The table or view specified as the target of an INSERT, DELETE, or UPDATE statement also appears in the FROM clause of a subquery within the statement.

The table or view that is the target of an INSERT, UPDATE, or DELETE cannot also be used to supply the values to be inserted or to qualify the rows to be inserted, updated, or deleted.

The statement cannot be processed.

This message is only applicable to Version 1.2 servers and earlier, and hosts accessed through DDCS.

Action: The implied function is not supported. To attempt to obtain the desired result, create a temporary copy of the object table or view and address the subselect to that copy.

sqlcode: -118

sqlstate: 42902

SQL0119N

An expression starting with "<expression-start>" specified in a SELECT clause, HAVING clause, or ORDER BY clause is not specified in the GROUP BY clause or it is in a SELECT clause, HAVING clause, or ORDER BY clause with a column function and no GROUP BY clause is specified.

Cause: The SELECT statement has one of the following errors:

- The identified expression and a column function are contained in the SELECT clause, HAVING clause, or ORDER BY clause but there is no GROUP BY clause
- The identified expression is contained in the SELECT clause, HAVING CLAUSE, or ORDER BY clause but is not in the GROUP BY clause.

The identified expression is an expression that starts with "<expression-start>". The expression may be a single column name.

The statement cannot be processed.

Action: Correct the statement by including the expression in the GROUP BY clause that are in the SELECT clause, HAVING clause, or ORDER BY clause or by removing the column function from the SELECT statement.

sqlcode: -119

sqlstate: 42803

SQL0120N

A WHERE clause, GROUP BY clause, SET clause, or SET transition-variable statement contains a column function.

Cause: A WHERE clause can contain a column function only if that clause appears within a subquery of a HAVING clause and the argument of the column function is a correlated reference to a group. A GROUP BY clause can contain a column function only if the argument of the column function is a correlated reference to a column in a different subselect than the one containing the GROUP BY clause. A SET clause of an UPDATE statement or a SET transition-variable statement can only include a column function within a fullselect on the right hand side of an assignment.

The statement cannot be processed.

Action: Change the statement so that the column function is not used or used only where it is supported.

sqlcode: -120

sqlstate: 42903

SQL0121N

The column "<name>" is specified more than once in the INSERT, UPDATE or SET transition-variable statement.

Cause: The same column "<name>" is specified more than once in the list of columns of an INSERT statement, the left hand side of assignments in the SET clause of an UPDATE statement, or the left hand side of assignments in the SET transition-variable statement. Note that this error may occur when updating or inserting into a view where more than one column of the view is based on the same column of a base table.

The statement cannot be processed.

SQL0122N   SQL0129N

Action: Correct the syntax of the statement so each column name is specified only once.

sqlcode: -121

sqlstate: 42701

SQL0122N

A SELECT statement with no GROUP BY clause contains a column name and a column function in the SELECT clause, or a column name is contained in the SELECT clause but not in the GROUP BY clause.

Cause: The SELECT statement has one of the following errors:

- A column name and a column function are contained in the SELECT clause but there is no GROUP BY clause.
- A column name is contained in the SELECT clause but not in the GROUP BY clause.

The column may be contained in a scalar function.

The statement cannot be processed.

Action: Correct the statement by including the columns in the GROUP BY clause that are in the SELECT clause or by removing the columns from the SELECT clause.

sqlcode: -122

sqlstate: 42803

SQL0123N

The parameter in position "<n>" in the function "<name>" must be a constant or a keyword.

Cause: The parameter in position "<n>" in the function "<name>" is not a constant when it is required to be a constant or a keyword when it is required to be a keyword.

Action: Ensure that each argument of the function conforms to the definition of the corresponding parameter.

sqlcode: -123

sqlstate: 42601

SQL0125N

The column number in the ORDER BY clause is either less than one or greater than the number of columns in the result table.

Cause: The ORDER BY clause in the statement contains a column number that is either less than 1 or greater than the number of columns in the result table (the number of items in the SELECT clause).

The statement cannot be processed.

Action: Correct the syntax of the ORDER BY clause so each column identifier properly identifies a column of the result table.

sqlcode: -125

sqlstate: 42805

SQL0127N

DISTINCT is specified more than once.

Cause: The DISTINCT qualifier cannot be used:

- In both the SELECT clause and a column function
- In two or more column functions in the same SELECT statement.

The statement cannot be processed.

Action: This error is only applicable to releases of DB2 prior to DB2 Version 2 and hosts accessed through DDCS.

sqlcode: -127

sqlstate: 42905

SQL0129N

The statement contains too many table names (the maximum is 15).

Cause: The SQL statement contains too many table names. A single SQL statement can reference up to 15 tables. Each table in any view referenced is included in this limit.

The statement cannot be processed.

Action: Break the SQL statement into two or more simple statements with 15 or fewer table references.

This message is only applicable to Version 1.2 servers and earlier, and hosts accessed through DDCS.

sqlcode: -129

sqlstate: 54004

SQL0130N

The ESCAPE clause is not a single character, or the pattern string contains an invalid occurrence of the escape character.

Cause: The escape character must be a single character no more than two bytes in length. It can only appear in the pattern string if it is followed by itself, a percent sign, or an underscore. For more information about the ESCAPE clause on the LIKE predicate, refer to the *SQL Reference*.

Action: Correct the pattern string or the escape character accordingly.

sqlcode: -130

sqlstate: 22019, 22025

SQL0131N

The operands of a LIKE predicate have incompatible data types.

Cause: If the expression to the left of a LIKE or NOT LIKE is of type character, the expression to the right must be of type character.

If the expression to the left is of type graphic, the expression to the right must be of type graphic.

If the expression to the left is of type BLOB, the expression to the right must be of type BLOB.

The statement cannot be processed.

Action: Correct the expressions of the LIKE predicate to be the same data type.

sqlcode: -131

sqlstate: 42818

SQL0132N

A LIKE predicate or POSSTR scalar function is not valid because the first operand is not a string expression or the second operand is not a string.

Cause: A LIKE predicate or POSSTR scalar function appearing in the statement is not valid because either the first operand is not a string expression or the second operand is not a string.

The operand appearing to the left of a LIKE or NOT LIKE predicate or the first operand of POSSTR must be a string expression. The value appearing to the right of the predicate or the second operand of POSSTR can be one of:

- a constant
- a special register
- a host variable
- a scalar function whose operands are any of the above
- an expression concatenating any of the above

with the restrictions that:

- no element in the expression can be of type LONG VARCHAR, CLOB, LONG VARGRAPHIC, or DBCLOB. In addition it cannot be a BLOB file reference variable.
- the actual length of the expression cannot be more than 4000 bytes.

A LIKE predicate or POSSTR scalar function cannot be used with DATE, TIME, or TIMESTAMP.

The statement cannot be processed.

Action: Check and correct the syntax of LIKE and POSSTR

sqlcode: -132

sqlstate: 42824

SQL0134N

Improper use of a string column, host variable, constant, or function "<name>".

Cause: The use of the string "<name>" is not permitted.

An expression resulting in a string data type with a maximum length greater than 254 bytes is not permitted in:

- A SELECT DISTINCT statement
- A GROUP BY clause
- An ORDER BY clause
- A column function with DISTINCT
- A SELECT or VALUES statement of a set operator other than UNION ALL.

An expression resulting in a LONG VARCHAR or LONG VARGRAPHIC data type is not permitted in:

- A predicate other than EXISTS or NULL
- A column function
- The SELECT clause of a subquery of a predicate other than EXISTS or NULL
- The SELECT clause of a subselect in an INSERT statement
- The value expression of a SET clause in an UPDATE statement unless the expression is a

LONG VARCHAR or LONG VARGRAPHIC host variable
- A SELECT statement of a set operator (except UNION ALL)
- VARGRAPHIC scalar function.

DataJoiner users: in a pass-through session, a data source-specific restriction can cause this error.

The statement cannot be processed.

Action: The requested operation on the string is not supported.

Note: If it is unclear as to how the 254 byte limit is being exceeded, consider that codepage conversion operations may be required to evaluate the string expression. Depending on the source and target codepages, the target may have a greater length attribute than the source. For more information, refer to the SQL Reference for discussions on string restrictions and string conversions.

sqlcode: -134

sqlstate: 42907

SQL0137N

The length resulting from "<operation>" is greater than "<maximum value>".

Cause: The result of concatenating the given operands was longer than what is supported by the result type.

Character string results are limited to 32,700 bytes unless one of the operands is a CLOB and then the limit is 2 gigabytes.

Graphic string results are limited to 16,350 characters unless one of the operands is a DBCLOB and then the limit is 1,073,741,823 (1 less than 1 gigabyte) double byte characters.

Binary string results (operands are BLOB) are limited to 2 gigabytes.

Action: Ensure that the sum of the lengths of the operands does not exceed the supported maximum and try the operation again.

sqlcode: -137

sqlstate: 54006

SQL0138N

The second or third argument of the SUBSTR function is out of range.

Cause: One of the following conditions occurred:
- The second argument of the SUBSTR function is less than 1 or greater than M.
- The third argument of the SUBSTR function is an expression with a value less than 0 or greater than M–N+1.

M is the length of the first argument if it is fixed length or the maximum length of the first argument if it is varying length. N is the value of the second argument.

The statement cannot be processed.

Action: Ensure that the second and third arguments of the SUBSTR function have values that conform to the above rules.

sqlcode: -138

sqlstate: 22011

SQL0139W

A redundant clause appears in the specification for column "<column>".

Cause: The clause in the column specification is redundant.

The statement was processed successfully, but the redundant clause was ignored.

Action: Correct the column specification.

sqlcode: +139

sqlstate: 01589

SQL0142N

The SQL statement is not supported.

Cause: An embedded SQL statement that may be valid for other IBM relational database products is not supported by the database manager.

DataJoiner users: check to see if an SQL statement was directed at a data source that does not support the SQL statement.

The statement cannot be processed.

Action: Change the syntax of the SQL statement or remove the statement from the program.

DataJoiner users: if the reason is unknown, isolate the problem to the data source failing the request (see the

Problem Determination Guide for procedures to follow to identify the failing data source) and examine the SQL dialect for that data source.

SQL0143W

The SQL statement is not supported; invalid syntax is ignored.

Cause: An embedded SQL statement that may be valid for other IBM relational database products is understood but is not supported by the database manager.

The statement may produce inconsistent or unwanted results.

Action: Change the syntax of the SQL statement or remove the statement from the program.

SQL0150N

The view in the INSERT, DELETE, or UPDATE statement is a view for which the requested operation is not permitted.

Cause: The view named in the INSERT, UPDATE, or DELETE statement is defined so that the requested insert, update, or delete operation cannot be performed.

The view is read only if the SELECT statement contains any of the following:

- A DISTINCT keyword
- A column function in the select list
- A GROUP BY or HAVING clause
- A FROM clause that identifies one of the following:
  - More than one table or view
  - A read-only view (READONLY column of SYSCAT.SYSVIEWS is set to 'Y')
  - A table or view that is also identified in the FROM clause of a subquery of the SELECT statement. (NOTE: This only applies to releases prior to DB2 Version 2.)
- A set operator (other then UNION ALL).
- DataJoiner users: a data source specific limitation that causes the column not to be updateable

Note that these conditions do not apply to subqueries of the SELECT statement.

The statement cannot be processed.

Action: The requested function cannot be performed on the view.

DataJoiner users: if the reason is unknown, isolate the problem to the data source failing the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and examine the object definition and the update restrictions for that data source.

sqlcode: -150

sqlstate: 42807

SQL0151N

The column "<name>" cannot be updated.

Cause: The specified column cannot be updated because one of the following was attempted.

- the object table is a view, and the specified column is derived from a scalar function, expression, keyword, constant, or column of a view where that column cannot be updated.

- the specified column is a non-updateable column of a system catalog.

DataJoiner users should check to see if:

- The column is not an updateable column. See section on "System Catalog Views", in the *IBM DataJoiner Application Programming and SQL Reference Supplement* for a description of the system catalog tables and which columns are updateable

- Some other data source specific limitation prevents the column from being updated.

Some DataJoiner data sources do not provide the appropriate values for the "<name>". In these cases the appropriate fields contain values (such as "unknown") that indicate that the actual value is unknown.

The statement cannot be processed.

Action: The requested function is not supported. For a list of updateable catalogs (and the updateable columns) see the *SQL Reference*.

DataJoiner users: if the reason is unknown, isolate the problem to the data source failing the request (see the *Problem Determination Guide*) and examine the definition and the update restrictions for that data source.

sqlcode: -151

sqlstate: 42808

SQL0153N SQL0158N

SQL0153N

The CREATE VIEW statement or common table expression does not include a column list.

Cause: A column list must be specified in a CREATE VIEW statement or a common table expression when:

- any element of the SELECT list in the fullselect is other than a column name and is not named using the AS clause
- any two elements are the same column name that were not renamed using the AS clause.

The statement cannot be processed. For a CREATE VIEW statement, the view was not created.

Action: Provide a column name list in the CREATE VIEW statement or common table expression or name the columns in the SELECT list of the fullselect using the AS clause.

sqlcode: -153

sqlstate: 42908

SQL0155N

A trigger transition table cannot be modified.

Cause: The trigger includes a REFERENCING clause with an OLD_TABLE or NEW_TABLE identified. A DELETE, INSERT or UPDATE triggered SQL statement has used the name specified as the OLD_TABLE or NEW_TABLE as the table to modify.

Action: Remove the DELETE, INSERT or UPDATE triggered SQL statement from the triggered action or change the name of the transition table so that it does not conflict with the table you are attempting to modify.

sqlcode: -155

sqlstate: 42807

SQL0156N

The name used for this operation is not a table.

Cause: The SQL statements ALTER TABLE, DROP TABLE, SET CONSTRAINTS, CREATE TRIGGER, CREATE INDEX, LOCK TABLE, and RENAME TABLE are applicable only to tables, not to views.

DataJoiner users: ALTER TABLE and DROP TABLE are not allowed on nicknames. The RUNSTATS, LOAD and UNLOAD utilities are also applicable only to tables, not to views.

The statement or utility cannot be processed.

Action: Verify that the correct table name is specified in the statement. If an alias name was supplied, ensure that the alias resolves to a table.

DataJoiner users: if the statement is an ALTER TABLE or DROP TABLE statement, verify that the object is not a nickname.

sqlcode: -156

sqlstate: 42809

SQL0157N

"<name>" is not allowed in a FOREIGN KEY clause because it identifies a view.

Cause: The view "<name>" 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.

The statement cannot be processed. The specified table is not created or altered.

Action: Correct the statement to specify a table name in the FOREIGN KEY clause.

If an alias name was supplied, ensure that the alias resolves to a table.

sqlcode: -157

sqlstate: 42810

SQL0158N

The number of columns for the view is not the same as the number of columns in the SELECT clause.

Cause: The number of column names specified for a view in a CREATE VIEW statement must equal the number of elements (for example, column names, SQL functions, and expressions) specified in the AS SELECT clause.

The statement cannot be processed. The specified view is not created.

Action: Correct the syntax of the statement to specify a column name for each column in the view.

sqlcode: -158

sqlstate: 42811

SQL0159N

DROP or COMMENT ON "<object>" identifies a(n) "<object-type>" rather than a(n) "<expected-object-type>".

Cause: The object "<object>" specified as part of a DROP or COMMENT ON statement refers to an object of type "<object-type>" instead of the expected type "<expected-object-type>".

The type of the object provided with a DROP or COMMENT ON statement must match the type identified by the key word used in the statement. For example, if the statement is DROP ALIAS *PBIRD.T1*, then *PBIRD.T1* must be an alias name.

Action: Change the statement to properly match the type of object identified by "<expected-object-type>".

sqlcode: -159

sqlstate: 42809

SQL0160N

The WITH CHECK option is not valid for the specified view.

Cause: The WITH CHECK option cannot be used in a view definition if:

- The view is defined as read only. The view is read only if the SELECT statement contains any of the following. (Note that these conditions do not apply to subqueries of the SELECT statement.)
    – A DISTINCT keyword
    – A column function in the selected list
    – A GROUP BY or HAVING clause
    – A FROM clause that identifies either:
        - More than one table or view
        - A read-only view
    – A set operator (other than UNION ALL).
- The SELECT statement in the CREATE VIEW statement includes a subquery (except for specific statistic columns of some catalog tables).

The statement cannot be processed. The specified view was not created.

Action: Remove the WITH CHECK option or change the view definition so it conforms to the rules above.

sqlcode: -160

sqlstate: 42813

SQL0161N

The resulting row of the INSERT or UPDATE does not conform to the view definition.

Cause: The WITH CHECK option was specified in the view definition that is the object of the INSERT or UPDATE statement. Consequently, all attempts to insert or update rows in that view are checked to ensure that the results conform to the view definition.

The statement cannot be processed. No inserts or updates are performed, and the contents of the view and underlying base table remain unchanged.

Action: Examine the view definition to determine why the requested INSERT or UPDATE was rejected. Note that this may be a data-dependent condition.

Requested INSERT or UPDATE may be trying to put out of range value(s) in the target column(s). For system catalog updates, see the *SQL Reference* for valid ranges of values in various updateable columns of catalogs.

DataJoiner users, if the reason is unknown, isolate the problem to the data source request (see the *Problem Determination Guide*) and examine the definition and the view definition for that data source.

sqlcode: -161

sqlstate: 44000

SQL0170N

The number of arguments for function "<name>" is incorrect.

Cause: The specified scalar function "<name>" has either too few or too many arguments.

The statement cannot be processed.

Action: Ensure that the correct number of arguments is specified for the scalar function.

sqlcode: -170

sqlstate: 42605

SQL0171N

The data type, length or value of argument "<n>" of function "<name>" is incorrect.

Cause: The data type, length or value of argument "<n>" of scalar function "<name>" is incorrect.

The statement cannot be processed.

Action: Ensure the arguments of the scalar function conform to the function rules.

Chapter 7. SQL Messages **89**

SQL0172N • SQL0181N

sqlcode: -171

sqlstate: 42815

SQL0172N

"\<name\>" is not a valid function name.

Cause: The SQL statement includes an unknown scalar function.

The statement cannot be processed.

Action: Ensure that you correctly spell the function name.

sqlcode: -172

sqlstate: 42601

SQL0176N

The second, third or fourth argument of the TRANSLATE scalar function is incorrect.

Cause: The statement is not correct for one or more of the following reasons:

- The translate scalar function does not allow replacement of a character by another character which is encoded using a different number of bytes. For example, a single-byte character cannot be replaced with a double-byte character nor can a double-byte character be replaced with a single-byte character.
- The second and third arguments of the translate scalar function must end with correctly formed characters.
- The fourth argument of the translate scalar function must be a correctly formed single-byte character if the first argument is CHAR or VARCHAR.
- The fourth argument of the translate scalar function must be a correctly formed double-byte character if the first argument is GRAPHIC or VARGRAPHIC.

The statement cannot be processed.

Action: Ensure that the second, third, and fourth arguments of the translate scalar function have correct values.

sqlcode: -176

sqlstate: 42815

SQL0180N

The syntax of the string representation of a datetime value is incorrect.

Cause: The string representation of a date, time, or timestamp value does not conform to the syntax for the specified or implied data type.

The statement cannot be processed.

Action: Ensure that the syntax of the date, time, or timestamp value conforms to the syntax for its data type. If the string is not intended to be a date, time, or timestamp value, ensure that when used, it does not imply that data type.

DataJoiner users: the problem might be due to a date/time representation problem at the DataJoiner data source. If the reason is unknown, isolate the problem to the data source failing the request (see the *Problem Determination Guide*) and examine the date/time representation restrictions for that data source.

sqlcode: -180

sqlstate: 22007

SQL0181N

The string representation of a datetime value is out of range.

Cause: The string representation of a date, time or timestamp value contains a value that is out of range. The correct ranges for date, time, or timestamp values are as follows:

- 0001 to 9999 for years.
- 1 to 12 for months.
- 1 to 31 for days when the month is 1, 3, 5, 7, 8, 10, and 12.
- 1 to 30 for days when the month is 4, 6, 9, 11.
- 1 to 28 for days when the month is 2 in a non-leap year.
- 1 to 29 for days when the month is 2 in a leap year.
- 0 to 24 for hours. If the hour is 24, the other parts of the time are 0. If the format is USA, the hour cannot be greater than 12.
- 0 to 59 for minutes.
- 0 to 59 for seconds.
- 0 to 999999 for microseconds.
- 001 to 365 for the day portion of a Julian calendar date when the year is not a leap year.
- 001 to 366 for the day portion of a Julian calendar date when the year is a leap year.

DataJoiner users: the problem might be due to a date/time representation problem at the DataJoiner data

source. If the reason is unknown, isolate the problem to the data source failing the request (see the *Problem Determination Guide*) and examine the date/time representation restrictions for that data source.

The statement cannot be processed.

Action: Ensure that the value is within the valid range.

sqlcode: -181

sqlstate: 22007

SQL0182N

An expression with a datetime value or a labeled duration is not valid.

Cause: The specified expression contains an improperly used date, time, or timestamp value or labeled duration.

The statement cannot be processed.

Action: Examine the SQL statement to determine the cause of the problem and correct the statement.

sqlcode: -182

sqlstate: 42816

SQL0183N

A datetime arithmetic operation or datetime scalar function has a result that is not within the valid range of dates.

Cause: The result of an arithmetic operation is a date or timestamp duration which is outside the range 0001-01-01 through 9999-12-31.

The statement cannot be executed.

Action: Examine the SQL statement to determine the cause of the problem. If the problem is data-dependent, examine the data processed when the error occurred.

sqlcode: -183

sqlstate: 22008

SQL0187N

A reference to a current date/time special register is invalid.

Cause: An error indication from the operating system was detected while retrieving date/time information.

Action: Ensure that the system TOD clock and timezone settings are correct.

sqlcode: -187

sqlstate: 22506

SQL0191N

MBCS conversion error occurred because of a fragmented MBCS character.

Cause: The data contained an improperly formed multi-byte character. For example, the first byte of a DBCS character was found, but the second character was not.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Correct the input data and try again.

DataJoiner users: if the data is correct, isolate the problem to the data source failing the request (see the *Problem Determination Guide*) and examine the DBCS restrictions of that data source. If the data appears to be correct, contact IBM service for assistance.

sqlcode: -191

sqlstate: 22504

SQL0193N

In an ALTER TABLE statement, the column "<column-name>" has been specified as NOT NULL and either the DEFAULT clause was not specified or was specified as DEFAULT NULL.

Cause: When new columns are added to a table that already exists, a value must be assigned to that new column for all existing rows. By default, the null value is assigned. However, since the column has been defined as NOT NULL, a default value other than null must be defined.

Action: Either remove the NOT NULL restriction on the column or provide a default value other than null for the column.

sqlcode: -193

sqlstate: 42601

SQL0197N   SQL0204N

SQL0197N

A qualified column name is not allowed in the ORDER BY clause.

Cause: The ORDER BY clause of a fullselect that includes a set operator (UNION, EXCEPT, INTERSECT) cannot have qualified column names.

Action: Ensure that all column names in the ORDER BY clause are unqualified.

the statemes qunot be rp7ui.99

sqlstate: 42877

SQL0198N

The statement string of the PREPARE or EXECUTE IMMEDIATE statement is blank or empty.

Cause: The host variable that was the object of the PREPARE or EXECUTE IMMEDIATE statement either contained all blanks or was an empty string.

The PREPARE or EXECUTE IMMEDIATE could not be completed.

Action: Correct the logic of the program to ensure that a valid SQL statement is provided in the operand of the PREPARE or EXECUTE IMMEDIATE statement before it is executed.

sqlcode: -198

sqlstate: 42617

SQL0199N

The use of reserved word "<keyword>" following "<text>" is not valid. Expected tokens may include: "<token-list>".

Cause: A syntax error in the SQL statement was detected at the point in the statement when the reserved word "<keyword>" appears following "<text>". The "<text>" field indicates the 20 characters of the SQL statement that preceded the reserved word. Clauses in the statement may be in the wrong order.

As an aid to the programmer, a partial list of valid tokens is provided in the SQLERRM field of the SQLCA as "<token-list>". This list assumes the statement is correct to that point.

The statement cannot be processed.

Action: Examine the statement in the keyword area. Add a colon or SQL delimiter, if missing. Verify that the clauses are in the correct order. If the reserved word identified in the messages is listed as a reserved word, make the word a delimited identifier.

NOTE: This error is only applicable to releases of DB2 prior to Version 2.

sqlcode: -199

sqlstate: 42601

SQL0203N

win vin be ‰iscable column è\$to Tere ct)è¢ojeld ref>".This ojeld be the roject :99´è¢ã´¸´´t

---

If "<name>" is qualified, then the function does not exist. If "<name>" is unqualified, then a function of this name does not exist in any schema of the current function path.

This return code can be generated for any type of database object.

DataJoiner users: the object identified by "<name>" is not defined in the database or "<name>" is not a nickname in a DROP NICKNAME statement.

Some DataJoiner data sources do not provide the appropriate values for "<name>". In these cases, the message token will have the following format: "OBJECT:<data source> TABLE/VIEW", indicating that the actual value for the specified data source is unknown.

The statement cannot be processed.

Action: Ensure that the object name (including any required qualifiers) is correctly specified in the SQL statement and it exists. For missing data type or function in SOURCE clause, it may be that the object does not exist, OR it may be that the object does exist in some schema, but the schema is not present in your function path.

sqlcode: -204

sqlstate: 42704

SQL0205N

"<name>" is not a column of table "<table-name>".

Cause: No column with the specified "<name>" is defined for the table or view "<table-name>".

DataJoiner users: "<table-name>" could refer to a DataJoiner nickname.

The statement cannot be processed.

Action: Verify that the column and table names (including any required qualifiers) are specified correctly in the SQL statement.

Also, if you receive this error during a REORG or an IMPORT, the column names in the index may violate the database manager naming rules as defined in the *Administration Guide*.

sqlcode: -205

sqlstate: 42703

SQL0206N

"<name>" is not a column in an inserted table, updated table, or any table identified in a FROM clause or is not a valid transition variable for the subject table of a trigger.

Cause: This error can occur in the following cases:

- For an INSERT or UPDATE statement, the specified column is not a column of the table, or view that was specified as the object of the insert or update.
- For a SELECT or DELETE statement, the specified column is not a column of any of the tables or views identified in a FROM clause in the statement.
- For an ORDER BY clause, the specified column is a correlated column reference in a subselect, which is not allowed.
- For a CREATE TRIGGER statement:
  - A reference is made to a column of the subject table without using an OLD or NEW correlation name.
  - The left hand side of an assignment in the SET transition-variable statement in the triggered action specifies an old transition variable where only a new transition variable is supported.

The statement cannot be processed.

Action: Verify that the column and table names are specified correctly in the SQL statement. For a SELECT statement, ensure that all the required tables are named in the FROM clause. For a subselect in an ORDER BY clause, ensure that there are no correlated column references. If a correlation name is used for a table, verify that subsequent references use the correlation name and not the table name.

For a CREATE TRIGGER statement, ensure that only new transition variables are specified on the left hand side of assignments in the SET transition-variable statement and that any reference to columns of the subject table have a correlation name specified.

sqlcode: -206

sqlstate: 42703

SQL0207N   SQL0214N

A column name is not allowed in the ORDER BY clause of a SELECT statement used with a set operator.

Cause: A SELECT statement with a set operator contains an ORDER BY clause, which specifies column names. In this case, the list of columns in the ORDER BY clause must contain only integers.

The statement cannot be processed.

Action: Specify only integers in the list of columns in the ORDER BY clause.

NOTE: This error is only applicable to releases of DB2 prior to Version 2 and hosts accessed through DDCS.

sqlcode: -207

sqlstate: 42706

SQL0208N

The ORDER BY clause is not valid because column "<name>" is not part of the result table.

Cause: The statement is not valid because a column "<name>" specified in the ORDER BY list is not specified in the SELECT list and is not in the result table. Only columns in the result table can be used to order that result when the fullselect of the select-statement is not a subselect.

The statement cannot be processed.

Action: To correct the syntax of the statement, either add the specified column to the result table or delete it from the ORDER BY clause.

sqlcode: -208

sqlstate: 42707

SQL0212N

"<name>" is a duplicate table designator or is specified more than once in the REFERENCING clause of a trigger definition.

Cause: The exposed table, view, alias, or correlation name specified by "<name>" is identical to another exposed table, view, alias, or correlation name in the same FROM clause.

If the statement is a CREATE TRIGGER, the REFERENCING clause may have specified the same name as the subject table or may have the same name

for more than one of the OLD or NEW correlation names or the NEW_TABLE or OLD_TABLE identifiers.

The statement cannot be processed.

Action: Rewrite the FROM clause of the SELECT statement. Associate correlation names with table, view, or alias names so no exposed table, view, alias, or correlation name is identical to any other exposed table, view, alias, or correlation name in the FROM clause.

For a CREATE TRIGGER statement, change the names in the REFERENCING clause so that there are no duplicates.

sqlcode: -212

sqlstate: 42712

SQL0214N

An expression starting with "<expression-start>" in the "<clause-type>" clause is not valid. Reason code = "<reason-code>".

Cause: The expression identified by the first part of the expression "<expression-start>" in the "<clause-type>" clause is not valid for the reason specified by the "<reason-code>" as follows:

1   The fullselect of the select-statement is not a subselect. Expressions are not allowed in the ORDER BY clause for this type of select-statement. This reason code occurs only when "<clause-type>" is ORDER BY.

2   DISTINCT is specified in the select clause and the expression cannot be matched exactly with an expression in the select list. This reason code occurs only when "<clause-type>" is ORDER BY.

3   Grouping is caused by the presence of a column function in the ORDER BY clause. This reason code occurs only when "<clause-type>" is ORDER BY.

4   Expression in a GROUP BY clause cannot be a scalar-fullselect. This reason code occurs only when "<clause-type>" is GROUP BY.

The statement cannot be processed.

Action: Modify the select-statement based on the reason specified by the "<reason-code>" as follows:

1   Remove the expression from the ORDER BY clause. If attempting to reference a column of the result, change the sort key to the simple-integer or simple-column-name form.

2   Remove DISTINCT from the select clause or change the sort key to the simple-integer or simple-column-name form.

3   Add a GROUP BY clause or remove the column function from the ORDER BY clause.

4   Remove any scalar-fullselect from the GROUP BY clause. If grouping is desired on a column of the result that is based on a scalar-fullselect use a nested table expression or a common table expression to first provide a result table with the expression as a column of the result.

sqlcode: -214

sqlstate: 42822

SQL0217W

The statement was not executed as only Explain information requests are being processed.

Cause: The current value of one of the Explain special registers has been set to EXPLAIN. This value allows dynamic SQL statements to be prepared and explained but prevents any dynamic statement from being executed.

Action: Change the value of the appropriate Explain special register to a setting other than EXPLAIN by issuing the appropriate SET statement from the interface or application that is encountering this condition.

sqlcode: +217

sqlstate: 01604

SQL0219N

The required Explain table "<name>" does not exist.

Cause: The Explain facility has been invoked but was unable to find the required Explain table "<name>". The Explain tables must be created prior to invoking Explain.

Action: Create the required Explain tables. The SQL Data Definition Language statements needed to create the Explain tables are available in the file called EXPLAIN.DDL in the misc directory under sqllib.

sqlcode: -219

sqlstate: 42704

SQL0220N

The Explain table "<name>" does not have the proper definition.

Cause: The Explain facility has been invoked but the Explain table "<name>" did not have the expected definition. The definition could be incorrect due to:

- Incorrect number of columns defined
- Incorrect data type assigned to columns

Action: Correct the definitions of the specified Explain table. The SQL Data Definition Language statements needed to create the Explain tables are available in the file called EXPLAIN.DDL in the misc directory under sqllib.

sqlcode: -220

sqlstate: 55002

SQL0236W

SQLDA has only provided "<integer1>" SQLVAR entries. "<integer2>" SQLVAR entries are required for "<integer3>" columns. No SQLVAR entries have been set.

Cause: The value of the SQLN field of the SQLDA should be at least as large as the number of columns in the result set.

The database manager has not set any SQLVAR entries (and the SQLDOUBLED flag has been set to "off" (i.e. to the space character)).

Action: Increase the value of the SQLN field in the SQLDA to the value indicated in the message (making sure the SQLDA is large enough to support that amount) and resubmit the statement.

sqlcode: +236

sqlstate: 01005

SQL0237W

SQLDA has only provided "<integer1>" SQLVAR entries. Since at least one of the columns being described is a distinct type, "<integer2>" SQLVAR entries should have been specified. None of the Secondary SQLVAR entries have been set.

Cause: Since at least one of the columns in the result set is a distinct type, space should be provided for twice as many SQLVAR entries as the number of columns in the result set. The database manager has only set the

SQL0238W   SQL0259N

Base SQLVAR entries (and the SQLDOUBLED flag has been set off (i.e. to the space character).

Action: If there is no need for the additional information about the distinct type(s) in the result set, then no action is required. If the distinct type information is needed, the value of the SQLN field in the SQLDA should be increased to the value indicated in the message (after making sure that the SQDA is large enough to support that amount) and the statement should be resubmitted.

sqlcode: +237

sqlstate: 01594

```
┌─── SQL0238W ─────────────────────────────
│
│ SQLDA has only provided "<integer1>" SQLVAR
│ entries.  Since at least one of the columns being
│ described is a LOB, "<integer2>" SQLVAR
│ entries are required for "<integer3>" columns.
│ No SQLVAR entries have been set.
│
└──────────────────────────────────────────
```

Cause: Since at least one of the columns in the result set is a LOB, space must be provided for twice as many SQLVAR entries as the number of columns in the result set. Note also that one or more of the columns in the result set may be a distinct type.

The database manager has not set any SQLVAR entries (and the SQLDOUBLED flag has been set off (i.e. to the space character).

Action: Increase the value of the SQLN field in the SQLDA to the value indicated in the message (after making sure that the SQDA is large enough to support that amount) and resubmit the statement.

sqlcode: +238

sqlstate: 01005

```
┌─── SQL0239W ─────────────────────────────
│
│ SQLDA has only provided "<integer1>" SQLVAR
│ entries.  Since at least one of the columns being
│ described is a distinct type, "<integer2>"
│ SQLVAR entries are required for "<integer3>"
│ columns.  No SQLVAR entries have been set.
│
└──────────────────────────────────────────
```

Cause: If any of the columns in the result set is a distinct type then space should be provided for twice as many SQLVAR entries as the number of columns in the result set.

The database manager has not set any SQLVAR entries (and the SQLDOUBLED flag has been set off (i.e. to the space character).

Action: If the distinct type information is needed, the value of the SQLN field in the SQLDA should be increased to the value indicated in the message (after making sure the SQLDA is large enough to support that amount) and the statement should be resubmitted. If there is no need for the additional information about the distinct type(s) in the result set, then it is possible to resubmit the statement only providing enough SQLVAR entries to accommodate the number of columns in the result set.

sqlcode: +239

sqlstate: 01005

```
┌─── SQL0257N ─────────────────────────────
│
│ Raw DEVICE containers are currently not
│ supported on this platform.
│
└──────────────────────────────────────────
```

Cause: An attempt was made to use DEVICE containers. These are not currently supported on this platform.

Action: Use FILE containers or system-managed table spaces instead.

sqlcode: -257

sqlstate: 42994

```
┌─── SQL0258N ─────────────────────────────
│
│ Containers cannot be added while a rebalance of
│ the table space is in progress.
│
└──────────────────────────────────────────
```

Cause: Rebalancer is in progress. Details can be found in the system error log and/or the database manager error log.

Action: Wait until the rebalance has completed and then try the operation again.

sqlcode: -258

sqlstate: 55041

```
┌─── SQL0259N ─────────────────────────────
│
│ Container map for table space is too
│ complicated.
│
└──────────────────────────────────────────
```

Cause: The map structure keeps a record of how the table space address space is mapped to the various containers. If this gets too complicated, it does not fit in the table space file.

Details can be found in the system error log and/or the database manager error log.

Action: The table space may need to be rebalanced to more evenly distribute the data among the containers. This may simplify the mapping.

If this does not work, try making as many containers as possible the same size. Existing container sizes can be changed by backing up the table space and then using the database administration utility to change the containers. Restore the table space to the new containers.

sqlcode: -259

sqlstate: 54037

SQL0280W

View or trigger "<name>" has replaced an existing, inoperative view or trigger.

Cause: An existing, inoperative view or trigger "<name>" was replaced by:

- the new view definition as a result of a CREATE VIEW statement

- the new trigger definition as a result of a CREATE TRIGGER statement.

Action: None required.

sqlcode: +280

sqlstate: 01595

SQL0281N

Table space "<tablespace-name>" cannot be altered with additional containers because it is a system managed table space.

Cause: Additional containers cannot be added to a system managed table space. The table space must be managed by the database in order to add additional containers.

Action: Specify a managed by database table space on the ALTER TABLESPACE command.

sqlcode: -281

sqlstate: 42921

SQL0282N

Table space "<tablespace-name>" cannot be dropped because at least one of the tables in it, "<table-name>", has one or more of its parts in another table space.

Cause: A table in the specified table space does not contain all its parts in that table space. The base table, indexes or long data may be in another table space. Dropping this table space will not completely drop the table. This would leave the table in an inconsistent state. This table space cannot be dropped.

Action: Ensure that all objects contained in table space "<tablespace-name>" contain all their parts in this table space before attempting to drop the table space.

This may require dropping the table "<table-name>" before dropping the table space.

sqlcode: -282

sqlstate: 55024

SQL0283N

Temporary table space "<tablespace-name>" cannot be dropped because it is the only temporary table space in the database.

Cause: A database must contain at least one temporary table space. Dropping table space "<tablespace-name>" would remove the last temporary table space from the database.

Action: Ensure there will be another temporary table space in the database before attempting to delete this table space.

sqlcode: -283

sqlstate: 55026

SQL0284N

Table space "<tablespace-name>" following the clause "<clause>" cannot be a "<tablespace-type>" table space.

Cause: The CREATE TABLE statement specified a table space named "<tablespace-name>" following the clause "<clause>" that is not a valid type of table space for this clause.

This can occur in the following situations:

- "<tablespace-name>" was specified on the IN clause and the table space is not a REGULAR table space.

- "<tablespace-name>" was specified on the LONG IN clause and the table space is not a LONG, managed by database table space.
- "<tablespace-name>" was specified on the INDEX IN clause and the table space is not a REGULAR, managed by database table space.

Action: Correct the CREATE TABLE statement to specify a table space with the correct type for the "<clause>" clause.

sqlcode: -284

sqlstate: 42838

SQL0285N

The indexes and/or long columns for table "<table-name>" cannot be assigned to separate table spaces because the primary table space "<tablespace-name>" is a system managed table space.

Cause: If the primary table space is a system managed table space, all table parts must be contained in that table space. A table can have parts in separate table spaces only if the primary table space, index table space and long table space are database managed table spaces.

Action: Either specify a database managed table space for the primary table space, or do not assign the table parts to another table space.

sqlcode: -285

sqlstate: 42839

SQL0286N

No default primary table space exists for the new table.

Cause: The CREATE TABLE statement did not specify a table space. A table space defined by the user and a table space with the name USERSPACE1 do not exist in the database. These are the valid defaults.

Action: Specify a valid table space name in the CREATE TABLE statement or ensure that a valid default table space exists in the database.

sqlcode: -286

sqlstate: 42727

SQL0287N

SYSCATSPACE cannot be used for user objects.

Cause: The CREATE TABLE statement specified a table space named SYSCATSPACE which is reserved for catalog tables.

Action: Specify a different table space name.

sqlcode: -287

sqlstate: 42838

SQL0288N

A long table space cannot be defined using MANAGED BY SYSTEM.

Cause: The table space being defined is for use with large objects and long strings. These can only be stored in table spaces that are defined in database managed space. Thus a long table space cannot be defined to use system managed space.

Action: Remove the keyword LONG or change to MANAGED BY DATABASE in the CREATE TABLESPACE statement.

sqlcode: -288

sqlstate: 42613

SQL0289N

Unable to allocate new pages in a table space.

Cause: One of the following conditions is true:

1. All the containers assigned to this table space are full. This is the most likely cause of the error.
2. The table space object table is full.
3. A rebalance is in progress, but has not progressed far enough to enable the newly added space to be used.

Details can be found in the system error log and/or the database manager error log.

Action: For case 1, add new container(s) to the table space and try the operation again, after the rebalancer has made the new pages available for use. For case 2, drop unnecessary tables from this table space. For case 3, wait for the rebalancer to make more progress.

sqlcode: -289

sqlstate: 57011

SQL0290N

Table space access is not allowed.

Cause: A process attempted to access a table space which is in an invalid state for which the intended access is not allowed.

- If the table space is in a quiesced state, only processes which also hold the table space in a quiesced state are allowed access to the table space.
- If the table space is in any other state, only the process which is performing the action specified is allowed access to the table space.
- A temporary tablespace cannot be dropped which contains active temporary tables.
- The SET CONTAINER api cannot be used to set the container list unless the tablespace is in a "restore pending" state.

Details can be found in the system error log and/or the database manager error log.

Action: Possible actions include:

- If the table space is in a quiesced state, attempt to acquire a quiesced share or quiesced update state on the table space. Or, attempt to quiesce reset the table space.
- If the table space is in any other state, wait until the table space has returned to normal state before attempting to access the table space.

 compact.

Refer to the systems administration guide for further information about the table space states.

sqlcode: -290

sqlstate: 55039

SQL0291N

State transition not allowed on table space.

Cause: An attempt was made to change the state of a table space. Either the new state is not compatible with the current state of the table space, or an attempt was made to turn off a particular state and the table space was not in that state.

Details can be found in the system error log and/or the database manager error log.

Action: Table space states change when a backup is taken, the load completes, the rollforward completes, etc., depending on the current state of the table spaces. Refer to the systems administration guide for further information about the table space states.

sqlcode: -291

sqlstate: 55039

SQL0292N

An internal database file could not be created.

Cause: An internal database file could not be created. Details can be found in the system error log and/or the database manager error log.

Action: Check that the directory containing the file is accessible (e.g. mounted) and writeable by the database instance owner.

sqlcode: -292

sqlstate: 57047

SQL0293N

Error accessing a table space container.

Cause: One of the following conditions is true:

- A container (directory, file or raw device) was not found.
- A container is not tagged as being owned by the proper table space.
- A container tag is corrupt.

Details can be found in the system error log and/or the database manager error log.

Action: Try the following:

1. Check that the directory, file or device exists and that the file system is mounted (if it is on a separate file system). Containers must be readable and writable by the database instance owner.

2. If you have a recent backup, try restoring the table space or database. If that fails because of the bad container and the container is not a DEVICE type, try manually removing the container first.

3. Call your IBM service representative.

sqlcode: -293

sqlstate: 57048

SQL0294N SQL0298N

SQL0294N

The container is already in use.

Cause: Table space containers may not be shared.

Details can be found in the system error log and/or the database manager error log.

Action: Specify a different container for the table space.

sqlcode: -294

sqlstate: 42730

SQL0295N

The combined length for all container names for the table space is too long.

Cause: The total space required to store the list of containers exceeds the space allotted for this table space in the table space file.

Details can be found in the system error log and/or the database manager error log.

Action: Try one or more of the following:

- Use symbolic links, mounted file systems, etc. to shorten the new container names.
- Back up the table space and then use the database administration utility to reduce the number and/or name lengths of the containers. Restore the table space to the new containers.

sqlcode: -295

sqlstate: 54034

SQL0296N

Table space limit exceeded.

Cause: This database contains the maximum number of table spaces. No more can be created.

Details can be found in the system error log and/or the database manager error log.

Action: Delete table spaces that are not being used any more. Combine small table spaces by moving all data into one of them and deleting the other.

sqlcode: -296

sqlstate: 54035

SQL0297N

Pathname for container is too long.

Cause: The full path specifying the container name exceeds the maximum length allowed. If the container was specified as a path relative to the database directory, the concatenation of these two values must not exceed the maximum length.

Details can be found in the system error log and/or the database manager error log.

Action: Shorten the path length.

sqlcode: -297

sqlstate: 54036

SQL0298N

Bad container path.

Cause: The container path violates one of the following requirements:

- Container paths must be valid fully-qualified absolute paths or valid relative paths. The latter are interpreted relative to the database directory.
- The path must be read/write accessible to the instance id (check file permissions on UNIX-based systems).
- Containers must be of the type specified in the command (directory, file or device).
- Containers (directories) in system managed table spaces must be empty when designated as containers and must not be nested underneath other containers.
- The containers for one database must not be located underneath the directory of another database, and they may not be underneath any directory that appears to be for another database. This rules out any directory of the form SQLnnnnn, where 'n' is any digit.
- The container must be within the file size limit for the operating system.

This message will also be returned if any other unexpected error occurred which prevents DB2 from accessing the container.

Details can be found in the system error log and/or the database manager error log.

Action: Specify another container location or change the container to make it acceptable to DB2 (such as changing file permissions) and try again.

sqlcode: -298

sqlstate: 428B2

SQL0299N

Container is already assigned to the table space.

Cause: The container that you are attempting to add has already been assigned to the table space.

Details can be found in the system error log and/or the database manager error log.

Action: Choose another container and try again.

sqlcode: -299

sqlstate: 42731

SQL0301N

The value of a host variable in the EXECUTE or OPEN statement cannot be used because of its data type.

Cause: A host variable could not be used as specified in the statement because its data type is incompatible with the intended use of its value.

This error can occur as a result of specifying an incorrect host variable or an incorrect SQLTYPE value in a SQLDA on an EXECUTE or OPEN statement.

The statement cannot be processed.

Action: Verify that the data types of all host variables in the statement are compatible with the manner in which they are used.

sqlcode: -301

sqlstate: 07006

SQL0302N

The value of a host variable in the EXECUTE or OPEN statement is too large for its corresponding use.

Cause: The value of an input host variable was found to be too large for its use in the SELECT, VALUES, or prepared statement. One of the following occurred:

- The corresponding host variable or parameter marker used in the SQL statement is defined as string, but the input host variable contains a string that is too long.

- The corresponding host variable or parameter marker used in the SQL statement is defined as numeric, but the input host variable contains a numeric value that is too large.
- The terminating NUL character is missing from the C language NUL-terminated character string host variable.
- DataJoiner users: in a pass-through session, a data source-specific restriction might have been violated.

This error occurs as a result of specifying either an incorrect host variable or an incorrect SQLLEN value in an SQLDA on an EXECUTE or OPEN statement.

The statement cannot be processed.

Action: Ensure that the input host variable value is the correct type and length. If the input host variables supply values to parameter markers, match values with the implied data type and length of the parameter marker.

DataJoiner users: for a pass-through session, determine what data source is causing the error (see the *Problem Determination Guide* for procedures to follow to identify the failing data source). Examine the SQL dialect for that data source to determine which specific restriction has been violated, and adjust the failing statement as needed.

sqlcode: -302

sqlstate: 22001, 22003

SQL0303N

A value cannot be assigned to a host variable in the SELECT, VALUES, or FETCH statement because the data types are not compatible.

Cause: An imbedded SELECT or VALUES statement selects into a host variable "<name>", but the data type of the variable is not compatible with the data type of the corresponding SELECT-list or VALUES-list element. Both must be numeric, character, or graphic. For example, if the data type of the column is date or time, the data type of the variable must be character with an appropriate minimum length.

The statement cannot be processed.

Action: Verify that the table definitions are current and that the host variable has the correct data type.

sqlcode: -303

sqlstate: 42806

SQL0304N   SQL0309N

SQL0304N

A value cannot be assigned to a host variable because the value is not within the range of the host variable's data type.

Cause: A FETCH, VALUES, or SELECT into a host variable list failed because the host variable was not large enough to hold the retrieved value.

The statement cannot be processed. No data was retrieved.

Action: Verify that table definitions are current and that the host variable has the correct data type. For the ranges of SQL data types, refer to the *SQL Reference*.

DataJoiner users: for the ranges of SQL data types, refer to the *IBM DataJoiner Application Programming and SQL Reference Supplement*.

sqlcode: -304

sqlstate: 22001, 22003

SQL0305N

The NULL value cannot be assigned to a host variable in the SELECT or FETCH statement because no indicator variable is specified.

Cause: A FETCH or embedded SELECT or VALUES operation resulted in the retrieval of a NULL value to be inserted into a host variable for which no indicator variable was provided. An indicator variable must be supplied if a column can return a NULL value.

The statement cannot be processed. No data was retrieved.

Action: Examine the definition of the FETCH or SELECT object table or the elements of the VALUES list. Correct the program to provide indicator variables for all host variables where NULL values from those columns can be retrieved.

sqlcode: -305

sqlstate: 22002

SQL0306N

The host variable "<name>" is undefined.

Cause: The host variable "<name>" is not declared in any DECLARE SECTION.

The statement cannot be processed.

Action: Ensure that the host variable is declared and the name spelled correctly.

SQL0307N

The host variable "<name>" is already defined.

Cause: The host variable "<name>" has already been declared in a DECLARE SECTION.

The definition is ignored. The previous definition is used.

Action: Ensure that the host variable name is spelled correctly and the name is defined only once for each program.

SQL0308N

The limit on the number of host variables has been reached.

Cause: The limit on the number of host variables is dependent on how many will fit in the HOST_VARS column of SYSPLAN. This limit has been reached.

The remaining variable declarations are ignored.

Action: Either simplify the program, split the program into smaller, separate programs, or do both.

SQL0309N

The value of a host variable in the OPEN statement is NULL, but its corresponding use cannot be NULL.

Cause: The value of an input host variable was found to be NULL but the corresponding use in the SELECT, VALUES, or prepared statement did not specify an indicator variable.

The statement cannot be processed.

Action: Ensure you need a USING clause. Otherwise, ensure an indicator variable is specified only if required.

sqlcode: -309

sqlstate: 07002

SQL0310N

SQL statement contains too many host variables.

Cause: The maximum number of host variables was exceeded in the statement.

The statement cannot be processed.

Action: Ensure the statement has fewer host variables or is less complex.

SQL0311N

The specified length of a host variable is negative.

Cause: When evaluated, the length specification for a host string variable was negative. Either a negative SQLLEN value is used or a varying length string has a negative length.

The statement cannot be processed.

Action: Correct the program to ensure that all host string variable lengths are not negative.

sqlcode: -311

sqlstate: 22501

SQL0312N

The host variable "<host-name>" is used in a dynamic SQL statement, a view definition, or a trigger definition.

Cause: The host variable "<host-name>" appears in the SQL statement, but host variables are not allowed in dynamic SQL statements, in the SELECT statement of a view definition, or in the triggered action of a trigger definition.

The statement cannot be processed.

Action: Use parameter markers (?) instead of host variables for dynamic SQL statements. Do not use host variables and parameter markers in view or trigger definitions.

sqlcode: -312

sqlstate: 42618

SQL0313N

The number of host variables in the EXECUTE or OPEN statement is not equal to the number of input values required.

Cause: The number of host variables specified in the EXECUTE or OPEN statement is not the same as the number of host variables or parameter markers (?) appearing in the SQL statement.

The statement cannot be processed.

Action: Correct the application program so the number of host variables specified in the EXECUTE or OPEN statement and the number of host variables or parameter markers in the SQL statement are the same.

sqlcode: -313

sqlstate: 07001, 07004

SQL0314N

The host variable "<name>" is incorrectly declared.

Cause: The host variable "<name>" is not declared correctly for one of the following reasons:

- The type specified is not one that is supported.
- The length specification is 0, negative, or too large.
- An initializer is used.
- An incorrect syntax is specified.

The variable remains undefined.

Action: Ensure that you correctly specify only the declarations the database manager supports.

SQL0315N

The host variable is incorrectly declared.

Cause: The host variable is not declared correctly for one of the following reasons:

- The type specified is not one that is supported.
- The length specification is 0, negative or too large.
- An incorrect syntax is specified.

The variable remains undefined.

Action: Ensure that you correctly specify only the declarations the database manager supports.

SQL0317N   SQL0332N

SQL0317N

No END DECLARE SECTION was found after a BEGIN DECLARE SECTION.

Cause: The end of input was reached during processing of a DECLARE SECTION.

Precompilation is terminated.

Action: Add an END DECLARE SECTION statement to end a DECLARE SECTION.

SQL0318N

An END DECLARE SECTION was found without a previous BEGIN DECLARE SECTION.

Cause: An END DECLARE SECTION statement was found, but there was no previous BEGIN DECLARE SECTION.

The statement cannot be processed.

Action: Enter a BEGIN DECLARE SECTION before an END DECLARE SECTION.

SQL0324N

The "<usage>" variable "<name>" is the wrong type.

Cause: Either the INDICATOR variable "<name>" is not a small integer or the STATEMENT variable "<name>" is not a character data type.

The statement cannot be processed.

Action: Ensure that the variable is the correct type and is specified correctly.

SQL0332N

There is no available conversion for the source code page "<code page>" to the target code page "<code page>". Reason code "<reason-code>".

Cause: No conversion of data is supported from the source code page to the target code page. This error can occur under the following conditions:

- An error has occurred executing an SQL statement. The data cannot be processed by the database manager.

- An error has occurred when trying to import or export a WSF or an IXF file. The import or export will fail.

- DataJoiner users: the data source does not support the specified code page conversion.

The reason codes are as follows:

1   source and target code page combination is not supported by the database manager.

2   source and target code page combination is either not supported by the database manager or by the operating system character conversion utility on the client node.

3   source and target code page combination is either not supported by the database manager or by the operating system character conversion utility on the server node.

Action: Possible solutions include:

- Verify that data conversion between the source and target code page is supported by the database manager. For database manager code page support, check the *Installation and Operation Guide*. If Distributed Database Connection Services (DDCS) is being used, check the DDCS Installation and Configuration Guide.

- Data conversion for some code page combinations may be supported through database manager install options, depending on the language group of the source or target code page. Ensure that the appropriate ones are installed and accessible to the database manager and the client application, as indicated by the reason code returned. For a list of install options, consult the *Installation and Operation Guide* or the DDCS Installation and Configuration Guide.

- Data conversion for some code page combinations may be supported by the database manager through an operating system-provided character conversion utility, depending on the language group of the source or target code page. Check your operating system documentation for a list of supported conversions and ensure that the appropriate ones are installed and accessible to the database manager and the client application, as indicated by the reason code returned. For a list of operating system conversion utilities used, consult the *Installation and Operation Guide* or the DDCS Installation and Configuration Guide.

- Change the unsupported code page(s) to one of the documented pairs.

DataJoiner users: for DataJoiner code page support, see "National Language Support", in the *IBM DataJoiner Planning, Installation, and Configuration Guide*.

sqlcode: -332

sqlstate: 57017

SQL0334N

Overflow occurred while performing conversion from codepage "<source>" to codepage "<target>". The maximum size of the target area was "<max-len>". The source string length was "<source-len>" and its hexadecimal representation was "<string>".

Cause: During the execution of the SQL statement, a code page conversion operation has resulted in a string that is longer than the maximum size of the target object.

Action: Modify the data to avoid the overflow condition, depending on the circumstances, by:

- decreasing the length of the source string or increasing the size of the target object (please refer to the note below),
- altering the operation, or,
- ensuring that the application codepage and the database codepage are the same. This eliminates the need for codepage conversions for most connections.

Note: Automatic promotion of character or graphic string data types will not occur as part of character conversion. If the resultant string length exceeds the maximum length of the data type of the source string then an overflow has occurred. To correct this situation either change the data type of the source string or use data type casting to allow for an increase in the string length due to conversion.

sqlcode: -334

sqlstate: 22524

SQL0340N

The common table expression "<name>" has the same identifier as another occurrence of a common table expression definition within the same statement.

Cause: The common table expression name "<name>" is used in the definition of more than one common table expressions in the statement. The name used to describe a common table expression must be unique within the same statement.

The statement cannot be processed.

Action: Change the name of one of the common table expressions.

sqlcode: -340

sqlstate: 42726

SQL0341N

A cyclic reference exists between the common table expressions "<name1>" and "<name2>".

Cause: The common table expression "<name1>" refers to "<name2>" in a FROM clause within its fullselect and "<name2>" refers to "<name1>" in a FROM clause within its fullselects. Such forms of cyclic references are not allowed.

The statement cannot be processed.

Action: Remove the cyclic reference from one of the common table expressions.

sqlcode: -341

sqlstate: 42835

SQL0342N

The common table expression "<name>" cannot use SELECT DISTINCT and must use UNION ALL because it is recursive.

Cause: There are two possible explanations:

- A fullselect within the common table expression "<name>" cannot start with SELECT DISTINCT because the common table expression is recursive.
- A fullselect within the common table expression "<name>" specified UNION instead of UNION ALL as required for recursive common table expressions.

The statement cannot be processed.

Action: Remove the keyword DISTINCT from the common table expression, add the keyword ALL following UNION, or remove the recursive reference within the common table expression.

SQL0343N SQL0346N

SQL0343N

The column names are required for the recursive common table expression "\<name\>".

Cause: The recursive common table expression "\<name\>" must include the specification of the column names following the identifier of the common table expression.

The statement cannot be processed.

Action: Add column names following the identifier of the common table expression.

sqlcode: -343

sqlstate: 42908

SQL0344N

The recursive common table expression "\<name\>" has mismatched data types, lengths or code pages for column "\<column-name\>".

Cause: The recursive common table expression "\<name\>" has a column "\<column-name\>" that is referred to in the iterative fullselect of the common table expression. The data type, length and code page are set based on the initialization fullselect for this column. The result of the expression for the column "\<column-name\>" in the iterative fullselect has a different data type, length or code page that may result in failure to assign the value for the column.

The statement cannot be processed.

Action: Correct the column used in the fullselects of the recursive common table expression so that the initialization column matches the iterative columns.

sqlcode: -344

sqlstate: 42825

SQL0345N

The fullselect of the recursive common table expression "\<name\>" must be the UNION of two or more fullselects and cannot include column functions, GROUP BY clause, or HAVING clause.

Cause: The common table expression "\<name\>" includes a reference to itself and therefore:

- must be the union of two or more fullselects
- cannot include a GROUP BY clause

- cannot have column functions
- and cannot include a HAVING clause.

The statement cannot be processed.

Action: Change the common table expression by:

- making it a union of two or more fullselects
- removing any column functions, GROUP BY clause, or HAVING clause
- removing the recursive reference.

sqlcode: -345

sqlstate: 42836

SQL0346N

An invalid reference to common table expression "\<name\>" occurs in the first fullselect, as a second occurrence in the same FROM clause, or in the FROM clause of a subquery.

Cause: The common table expression "\<name\>" includes an invalid reference to itself as described by one of the following.

- A recursive reference in the first fullselect before the UNION ALL set operator. The first fullselect must be an initialization and cannot include a recursive reference.

- More than one reference to the same common table expression in the same FROM clause. Such references are not permitted in recursive common table expression.

- A recursive reference in the FROM clause of a subquery. A recursion cycle cannot be defined using a subquery.

The statement cannot be processed.

Action: Change one of the following:

- the fullselect prior to the union operator so that it does not include a recursive reference

- the FROM clause containing more than one reference to the same common table expression to just one reference

- the FROM clause of the subquery so that it does not reference the common table expression.

sqlcode: -346

sqlstate: 42836

SQL0347W

**The recursive common table expression
"<name>" may contain an infinite loop.**

Cause: The recursive common table expression called
"<name>" may not complete. This warning is based on
not finding specific syntax as part of the iterative portion
of the recursive common table expression. The
expected syntax includes:

- incrementing an INTEGER column in the iterative
  select list by 1.
- a predicate in the where clause of the iterative
  portion of the form "counter_col < constant" or
  "counter_col < :hostvar".

The absence of this syntax in the recursive common
table expression may result in an infinite loop. The data
or some other characteristic of the recursive common
table expression may allow the successful completion of
the statement anyway.

Action: To prevent an infinite loop, include the
expected syntax as described.

sqlcode: +347

sqlstate: 01605

SQL0350N

**LOB column "<column-name>" cannot be used
in an index, a key, or a constraint.**

Cause: Even in cases where it would not violate the
maximum size of an index, key, or constraint, a LOB
column or a distinct type column based on a LOB
column cannot be used in an index, key or constraint.

The statement cannot be processed.

Action: Remove the LOB column from the specification
of the index, key, or constraint.

sqlcode: -350

sqlstate: 42962

SQL0351N

**LOB data in position "<position-number>" of the
select-list cannot be sent using DRDA protocol.**

Cause: LOB data types are not defined in DRDA.
Hence, LOB data or distinct types based on LOB data
cannot be flowed using the DRDA protocol.
"<position-number>" is the position of the element with a
LOB data type in the select-list.

The statement cannot be processed.

Action: Make sure that LOB data is not being retrieved
when using DRDA by excluding them in the select-list, or
select smaller pieces of the LOB by using
CHAR(SUBSTR(CLOB)).

sqlcode: -351

sqlstate: 56084

SQL0352N

**LOB data in position "<position-number>" of the
input-list cannot be sent using DRDA protocol.**

Cause: LOB data types are not defined in DRDA.
Hence, LOB data cannot be flowed using the DRDA
protocol. "<position-number>" is the position of the
element with a LOB data type in the input-list.

The statement cannot be processed.

Action: Make sure that LOB data is not being sent
when using DRDA by excluding them in the input-list.

sqlcode: -352

sqlstate: 56084

SQL0355N

**The column "<column-name>", as defined, is too
large to be logged.**

Cause: Large Object data types (BLOB, CLOB, and
DBCLOB) may be created in sizes up to 2 gigabytes
(2147483647 bytes). Logging of data values is allowed
only upon objects which are less than or equal to than 1
gigabyte (1073741823 bytes) in size. Therefore, large
objects greater than 1 gigabyte in size can not be
logged.

Action: Either explicitly indicate logging of data is not
required, by using the NOT LOGGED phrase during
column creation, or reduce the maximum size of the
column to 1 gigabyte or below.

sqlcode: -355

SQL0401N SQL0404N

sqlstate: 42993

SQL0401N

The data types of the operands for the operation "<operator>" are not compatible.

Cause: The operation "<operator>" appearing within the SQL statement has a mixture of numeric and nonnumeric operands, or the operation operands are not compatible.

DataJoiner users: this data type violation can be at the data source or at DataJoiner.

Some DataJoiner data sources do not provide the appropriate values for "<operator>". In these cases the message token will have the following format: "<data source>:UNKNOWN", indicating that the actual value for the specified data source is unknown.

The statement cannot be processed.

Action: Check all operand data types to ensure that they are comparable and compatible with the statement usage.

If all the SQL statement operands are correct and accessing a view, check the data types of all the view operands.

DataJoiner users: if the reason is unknown, isolate the problem to the data source failing the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and examine the data type restrictions for that data source.

sqlcode: -401

sqlstate: 42818

SQL0402N

The data type of an operand of an arithmetic function or operation "<operator>" is not numeric.

Cause: A nonnumeric operand is specified for the arithmetic function or operator "<operator>".

The statement cannot be processed.

Action: Correct the SQL statement syntax so all specified function or operator operands are numeric.

DataJoiner users: if the reason is unknown, isolate the problem to the data source failing the request (see the

Problem Determination Guide for procedures to follow to identify the failing data source) and examine the operators applied to that data source.

sqlcode: -402

sqlstate: 42819

SQL0403W

The newly defined alias "<name>" resolved to the object "<name2>" which is currently undefined.

Cause: The alias <name> has been defined upon:

- a table or view which currently is not defined in the database
- another alias which resolves to a table or view which is not defined in the database.

The object <name2> is the undefined object. This object must exist before any SQL statement (other than CREATE ALIAS) can successfully use the newly created alias. The specified alias <name> is created.

An inoperative view is considered to be undefined for the purposes of creating an alias.

Action: Ensure that the undefined object <name2> is defined prior to using the newly created alias in an SQL statement (other than CREATE ALIAS).

sqlcode: +403

sqlstate: 01522

SQL0404N

A string in the UPDATE or INSERT statement is too long for column "<name>".

Cause: An INSERT or UPDATE statement specifies a value that is longer than the maximum-length string that can be stored in the column indicated.

The statement cannot be processed.

Note: "<name>" may or may not be returned in SQLCA, depending on the INSERT or UPDATE statement syntax.

Action: Check the length of the object column and correct the program or SQL statement so the insert or update string does not exceed the maximum length.

sqlcode: -404

sqlstate: 22001

SQL0405N

The numeric literal "<literal>" is not valid because its value is out of range.

Cause: The specified numeric literal is not in the acceptable range.

The proper ranges for SQL values are as follows:

- For FLOAT values:
    - 0
    - In the positive range of +2.225E-307 to +1.79769E+308
    - In the negative range of -1.79769E+308 to -2.225E-307.
- -99999999999999999999999999999999. to 99999999999999999999999999999999. for DECIMAL values
- -2147483647 to 2147483647 for INTEGER values
- -32768 to +32767 for small integer (SMALLINT) values.

In certain contexts the literal may have further restrictions that result in this error. For more information, look up the statement or clause in the *SQL Reference*.

DataJoiner users: this range violation can be at the data source or at DataJoiner. The proper ranges for SQL values located in data source tables depend on the data source. To determine the correct range, see the appropriate documentation for the data source. Some DataJoiner data sources do not provide the appropriate values for "<literal>". In these cases the message token will have the following format: "<data source>:UNKNOWN", indicating that the actual value for the specified data source is unknown.

The statement cannot be processed.

Action: Reduce the literal value to the appropriate size.

DataJoiner users: if the reason is unknown, isolate the problem to the data source failing the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and examine the data range restrictions for that data source.

sqlcode: -405

sqlstate: 42820

SQL0406N

A numeric value in the UPDATE or INSERT statement is not within the range of its target column.

Cause: The value of a host variable or a numeric value calculated during processing of the UPDATE or INSERT SQL statement is outside the target column range. This problem may be caused by the values occurring in the object column, the SQL operation being performed on those values, or both.

The statement cannot be processed.

Action: See the explanation of message SQL0405 for ranges allowed for numeric data types.

NOTE: For system catalog updates, see the *SQL Reference* for valid ranges in various columns of updateable catalogs.

DataJoiner users: for the ranges of SQL data types, refer to the *IBM DataJoiner Application Programming and SQL Reference Supplement*.

sqlcode: -406

sqlstate: 22003

SQL0407N

Assignment of a NULL value to a NOT NULL column "<name>" is not allowed.

Cause: One of the following occurred:

- The update or insert value was NULL, but the object column was declared as NOT NULL in the table definition. Consequently:
    - NULL values cannot be inserted into that column.
    - An update cannot set values in that column to NULL.
    - A SET transition-variable statement in a trigger cannot set values in that column to NULL.
- The update or insert value was DEFAULT, but the object column was declared as NOT NULL without WITH DEFAULT in the table definition. Consequently:
    - A default value of NULL cannot be inserted into that column.
    - An update cannot set default values of NULL in that column.
    - A SET transition-variable statement in a trigger cannot set default values of NULL in that column.

- The column name list for the INSERT statement omits a column declared NOT NULL and without WITH DEFAULT in the table definition.
- The view for the INSERT statement omits a column declared NOT NULL and without WITH DEFAULT in the base table definition.

DataJoiner users: this situation can be detected by DataJoiner or by the data source. Some DataJoiner data sources do not provide the appropriate values for "<name>". In these cases the message token will have the following format: "<data source>:UNKNOWN", indicating that the actual value for the specified data source is unknown.

The statement cannot be processed.

Note: Under some circumstances, the token "<name>" may not be filled in (sqlerrmc field of the SQLCA not filled in).

Action: Correct the SQL statement after examining the object table definition to determine which columns of the table have the NOT NULL attribute and do not have the WITH DEFAULT attribute.

DataJoiner users: if the reason is unknown, isolate the problem to the data source failing the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and examine the object definition for that data source. Remember that the defaults (NULL and NOT NULL) are not necessarily the same between data sources.

sqlcode: -407

sqlstate: 23502

SQL0408N

An UPDATE, INSERT, or SET transition-variable value is not compatible with the data type of its target column "<name>".

Cause: The data type of the value to be inserted into or set in the column "<name>" by an INSERT, UPDATE, or SET transition-variable statement is incompatible with the declared data type of that column. Both must be:

- Numeric
- Character
- Graphic
- Dates or character
- Times or character
- Timestamps or character.
- the same distinct type

The statement cannot be processed.

Action: Examine the current definition for the object table. Ensure the host variable or literal value assigned to the specified column is the proper data type.

sqlcode: -408

sqlstate: 42821

SQL0409N

The operand of a COUNT function is not valid.

Cause: As specified in the SQL statement, the operand of the COUNT function does not conform to the rules of SQL syntax. Only COUNT(*) and COUNT(DISTINCT *column*) are allowed.

The statement cannot be processed.

Action: Specify COUNT(*) or COUNT(DISTINCT column).

NOTE: This message is only applicable to versions of DB2 prior to Version 2.

sqlcode: -409

sqlstate: 42607

SQL0410N

The floating point literal "<literal>" contains more than 30 characters.

Cause: The specified floating point literal is more than 30 characters in length, excluding leading zeros. A floating point literal has a maximum length of 30 characters.

The statement cannot be processed.

Action: Shorten the specified literal.

sqlcode: -410

sqlstate: 42820

SQL0412N

The SELECT clause of a subquery specifies multiple columns.

Cause: In the context of the SQL statement, the subquery can have only one column specified in its SELECT clause.

The statement cannot be processed.

Action: Specify only one column in the SELECT clause of the subquery.

sqlcode: -412

sqlstate: 42823

SQL0413N

Overflow occurred during numeric data type conversion.

Cause: During processing of the SQL statement, an overflow condition arose when converting from one numeric type to another. Numeric conversion is performed according to the standard rules of SQL.

DataJoiner users: numeric conversion can occur at DataJoiner, at data sources, or both.

The statement cannot be processed. No data was retrieved, updated, or deleted.

Action: Examine the syntax of the SQL statement to determine the cause of the error. If the problem is data-dependent, it may be necessary to examine the data processed at the time of the error.

DataJoiner users: if the reason is unknown, isolate the problem to the data source failing the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and examine the data range restrictions for that data source.

sqlcode: -413

sqlstate: 22003

SQL0415N

The data types of corresponding columns in the SELECT or VALUES statements connected by a set operator are not compatible.

Cause: A set operator connects SELECT or VALUES statements that contain columns that are incompatible for one of the following reasons:

- Both columns are not characters
- Both columns are not numbers
- Both columns are not dates
- Both columns are not times
- Both columns are not timestamps
- Both columns are not graphic
- Both columns are not the same user-defined distinct type.

If the data type of a column is character, date, time, or timestamp the corresponding column may be a literal string.

The statement cannot be processed.

Action: Correct the column names used in the SELECT statements or the expressions in the VALUES statements so that all corresponding columns are compatible types.

sqlcode: -415

sqlstate: 42825

SQL0416N

You cannot specify a result column longer than 254 bytes in the SELECT or VALUES statements connected by a set operator other than UNION ALL.

Cause: One of the SELECT or VALUES statements connected by a set operator specifies a result column that is longer than 254 bytes. VARCHAR or VARGRAPHIC result columns longer than 254 bytes can be used only with the UNION ALL set operator.

The statement cannot be processed.

Action: Either use the UNION ALL operator instead of UNION, or remove the result columns longer than 254 bytes from the SELECT or VALUES statements.

sqlcode: -416

sqlstate: 42907

SQL0417N

A statement string to be prepared contains parameter markers as the operands of the same operator.

Cause: The statement string specified as the object of a PREPARE or EXECUTE IMMEDIATE contains a predicate or expression in which parameter markers have been used as operands of the same operator without a CAST specification. For example:

    ? > ?

The statement cannot be processed.

Action: This syntax is not supported. Use a CAST specification to give at least one of the parameter markers a data type.

sqlcode: -417

sqlstate: 42609

SQL0418N   SQL0421N

SQL0418N

A statement contains a use of a parameter marker that is not valid.

Cause: Untyped parameter markers cannot be used:

- in a SELECT list
- as the sole argument of a datetime arithmetic operation
- in some cases as the sole argument of a scalar function

Parameter markers can never be used:

- in a statement that is not a prepared statement
- in the fullselect of a CREATE VIEW statement
- in the triggered action of a CREATE TRIGGER statement

The statement cannot be processed.

Action: Correct the syntax of the statement. If untyped parameter markers are not allowed, use the CAST specification to give the parameter marker a data type.

sqlcode: -418

sqlstate: 42610

SQL0419N

A decimal divide operation is not valid because the result would have a negative scale.

Cause: A specified decimal division is not valid because it will result in a negative scale.

The formula used internally to calculate the scale of the result for decimal division is:

```
Scale of result = 31 - np + ns - ds
```

where np is the precision of the numerator, ns is the scale of the numerator, and ds is the scale of the denominator.

DataJoiner users: the specified decimal division results in an invalid scale for that data source.

The statement cannot be processed.

Action: Examine and ensure the precision and scale of all columns that may participate in a decimal division. Note that an integer or small integer value may be converted to a decimal for this calculation.

DataJoiner users: if the reason is unknown, isolate the problem to the data source failing the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and examine the data range restrictions for that data source.

sqlcode: -419

sqlstate: 42911

SQL0420N

Invalid character found in a character string argument of the function "<function-name>".

Cause: The function "<function-name>" has a character string argument that contains a character that is not valid in a numeric SQL constant. The function may have been called as a result of using the CAST specification with "<function-name>" as the target data type. The function or data type used in the SQL statement may be a synonym for "<function-name>".

If a decimal character is specified in the DECIMAL function then that is the character that must be used in place of the default decimal character.

Action: Ensure that the character strings that are being converted to numeric types contain only characters that are valid in numeric SQL constants, using the decimal character, if specified.

sqlcode: -420

sqlstate: 22018

SQL0421N

The operands of a set operator or a VALUES clause do not have the same number of columns.

Cause: The operands of a set operator such as UNION, EXCEPT, or INTERSECT must have the same number of columns. The rows in a VALUES clause must have the same number of columns.

The statement cannot be processed.

Action: Change the SQL statement so each operand, or each row of a VALUES clause, has exactly the same number of columns.

sqlcode: -421

sqlstate: 42826

SQL0423N

LOB locator variable "<variable-position>" does not currently represent any value.

Cause: A locator variable is in error. Either it has not had a LOB value assigned to it, or the locator associated with the variable has been freed.

If "<variable-position>" is provided, it gives the ordinal position of the variable in error in the set of variables specified. Depending on when the error is detected, the database manager may not be able to determine "<variable-position>".

Action: Correct the program so that the LOB locator variables used in the SQL statement have valid LOB values before the statement is executed. A LOB value can be assigned to a locator variable by means of a SELECT INTO statement, a VALUES INTO statement, or a FETCH statement.

sqlcode: -423

sqlstate: 0F001

SQL0426N

Dynamic commit invalid for application execution environment.

Cause: An application executing in a CONNECT TYPE 2 environment or Distributed Transaction Processing (DTP) environment such as CICS has attempted to execute an SQL dynamic COMMIT statement. The SQL dynamic COMMIT statement cannot be executed in this environment.

Action:

- Use the commit statement provided by the DTP environment to perform the commit. For example, in a CICS environment this would be the CICS SYNCPOINT command.

- If this statement was executed within a stored procedure, remove the statement entirely.

sqlcode: -426

sqlstate: 2D528

SQL0427N

Dynamic rollback invalid for application execution environment.

Cause: An application executing in a CONNECT TYPE 2 environment or Distributed Transaction Processing (DTP) environment such as CICS has attempted to execute an SQL dynamic ROLLBACK statement. The SQL dynamic ROLLBACK statement cannot be executed in this environment.

Action:

- Use the rollback statement provided by the DTP environment to perform the rollback. For example, in a CICS environment this would be the CICS SYNCPOINT ROLLBACK command.

- If this statement was executed within a stored procedure, remove the statement entirely.

sqlcode: -427

sqlstate: 2D529

SQL0428N

DISCONNECT cannot be issued if a connection it is directed against has executed SQL within the unit of work.

Cause: DISCONNECT will not be executed against a connection in the following cases:

- If SQL has been issued against the connection within the unit of work.

- If a WITH HOLD cursor is currently open for the connection.

Note that DISCONNECT ALL causes the DISCONNECT to be directed against all connections, so the request will fail if any of the connections violate the above restrictions.

Action: Possible actions:

- Issue the DISCONNECT statement before any SQL statements have been issued to the database.

- COMMIT or ROLLBACK the unit of work before issuing the DISCONNECT.

- Close WITH HOLD cursors before committing a unit of work if a DISCONNECT is required for connections for which the cursors were open.

sqlcode: -428

sqlstate: 25501

SQL0429N   SQL0433N

SQL0429N

**The maximum number of concurrent LOB
locators has been exceeded.**

Cause: A maximum of 32,000 concurrent LOB locators
per unit of work, is supported by DB2.

Action: Modify the program so that it requires fewer
concurrent LOB locators and try again.

sqlcode: -429

sqlstate: 54028

SQL0430N

**User defined function "<function-name>"
(specific name "<specific-name>") has
abnormally terminated.**

Cause: An abnormal termination has occurred while
the named UDF was in control.

Action: The UDF needs to be fixed. Contact the author
of the UDF or your database administrator. Until it is
fixed, the UDF should not be used.

sqlcode: -430

sqlstate: 38503

SQL0431N

**User defined function "<function-name>"
(specific name "<specific-name>") has been
interrupted by the user.**

Cause: A user/client interrupt has occurred while the
named UDF was in control.

Action: This could indicate some problem in the UDF,
such as an infinite loop or wait. If the problem persists,
(i.e. the need to interrupt results in the same error
condition), then contact the author of the UDF or your
database administrator. Until the problem is fixed, the
UDF should not be used.

sqlcode: -431

sqlstate: 38504

SQL0432N

**A parameter marker cannot have the user
defined type name "<udt-name>".**

Cause: A parameter marker in the statement has been
determined as having the user-defined type
"<udt-name>" based on the context in which it is used.
A parameter marker cannot have a user-defined type as
its data type unless it is part of an assignment (VALUES
clause of INSERT or SET clause of UPDATE) or it is
being explicitly cast to a user-defined distinct data type
using the CAST specification.

The statement cannot be processed.

Action: Use an explicit cast to the user-defined distinct
data type for the parameter marker or cast the columns
that are user-defined distinct data types to their
corresponding source data type.

sqlcode: -432

sqlstate: 42841

SQL0433N

Value "<value>" is too long.

Cause: The value "<value>" required truncation by a
system (built-in) cast or adjustment function, which was
called to transform the value in some way. The
truncation is not allowed where this value is used.

The value being transformed is one of the following:

- an argument to a user defined function (UDF)
- an input to the SET clause of an UPDATE
  statement
- a value being INSERTed into a table
- an input to a cast or adjustment function in some
  other context.
- a recursively referenced column whose data type
  and length is determined by the initialization part of
  recursion and may grow in the iterative part of the
  recursion.

The statement has failed.

Action: If "<value>" is a literal string in the SQL
statement, it is too long for its intended use.

If "<value>" is not a literal string, examine the SQL
statement to determine where the transformation is
taking place. Either the input to the transformation is too
long, or the target is too short.

Correct the problem and rerun the statement.

sqlcode: -433

sqlstate: 22001

SQL0434W

An unsupported value for clause "<clause>" has been replaced by the value "<value>".

Cause: The value that was specified for clause "<clause>" is not supported and has been replaced with the identified supported value "<value>".

Action: No change is required if the selected value is acceptable. Otherwise, specify a value that is valid for "<clause>".

sqlcode: +434

sqlstate: 01608

SQL0435N

An invalid SQLSTATE "<sqlstate>" is specified in the function RAISE_ERROR.

Cause: The SQLSTATE specified in the RAISE_ERROR function does not conform to the rules for an application defined SQLSTATE.

Action: Correct the SQLSTATE specified in the function RAISE_ERROR. The SQLSTATE must be a character string containing exactly 5 characters. It must be of type CHAR defined with a length of 5, or type VARCHAR defined with a length of 5 or greater. The SQLSTATE value must follow the rules for application-defined SQLSTATEs as follows:

- Each character must be from the set of digits ('0' through '9') or non-accented upper case letters ('A' through 'Z')
- The SQLSTATE class (first two characters) cannot be '00', '01', or '02' since these are not error classes.
- If the SQLSTATE class (first two characters) starts with the character '0' through '6' or 'A' through 'H', then the subclass (last three characters) must start with a character in the range 'I' through 'Z'
- If the SQLSTATE class (first two characters) starts with the character '7', '8', '9' or 'I' though 'Z', then the subclass (last three characters) can be any of '0' through '9' or 'A' through 'Z'.

sqlcode: -435

sqlstate: 428B3

SQL0436N

The terminating NULL character is missing from the C language NULL-terminated character string host variable.

Cause: The value of an input host variable code in the C programming language requires a NULL-terminator character at the end of the string.

The statement cannot be processed.

Action: Ensure that the value of the input host variable is terminated by the NULL-terminator character.

sqlcode: -436

sqlstate: 22024

SQL0437W

Performance of this complex query may be sub-optimal. Reason code: "<reason-code>".

Cause: The statement may achieve sub-optimal performance since the complexity of the query requires resources that are not available or optimization boundary conditions were encountered. The following is a list of reason codes:

1 The join enumeration method was altered due to memory constraints
2 The join enumeration method was altered due to query complexity
3 Optimizer cost underflow
4 Optimizer cost overflow
5 Query optimization class was too low
6 Optimizer ignored an invalid statistic

The statement will be processed.

Action: One or more of the following:

- Increase the size of the statement heap (stmtheap) in the database configuration file. (Reason code 1)
- Break the statement up into less complex SQL statements. (Reason codes 1,2,3,4)
- Ensure predicates do not over-specify the answer set (Reason code 3)
- Change the current query optimization class to a lower value (Reason codes 1,2)
- Issue Runstats for the tables involved in the query (Reason codes 3,4)
- Change the current query optimization class to a higher value (Reason code 5)

sqlcode: +437

sqlstate: 01602

SQL0438N   SQL0441N

SQL0438N

**Application raised error with diagnostic text:
"<text>".**

Cause: This error occurred as a result of execution of
the RAISE_ERROR function or the SIGNAL SQLSTATE
statement in a trigger.

Action: See application documentation.

sqlcode: -438

sqlstate: application-defined

SQL0439N

**User defined function "<function-name>" is
indirectly implemented by function
"<source-function>" which resulted in error
"<sqlcode>".**

Cause: The function "<function-name>" was referenced
in the user's statement. However, because the
SOURCE clause was used in the definition of this
function, it has turned out that function
"<source-function>" actually implements the function. (It
may be a direct or an indirect definition path from
"<function-name>" to "<source-function>".) At compile
time, the encapsulator (DB2 code which acts on behalf
of a function) for "<source-function>" has returned the
error identified by "<sqlcode>".

Action: The actual error situation needs to be
understood better before corrective action can be taken.
Look up the explanation for "<sqlcode>". If
"<source-function>" is a built-in function, the "<sqlcode>"
should indicate the problem, as in the case where a
built-in function is directly referenced in the user's
statement. If "<source-function>" is a user defined
function, the message most likely indicates a problem
with one of the arguments or with the result from the
function.

Correct the problem and try again.

sqlcode: -439

sqlstate: 428A0

SQL0440N

**No function by the name "<function-name>"
having compatible arguments was found in the
function path.**

Cause: This occurs in a reference to function
"<function-name>", when the database manager cannot
find a function it can use to implement the reference.
There are several reasons why this could occur:

- "<function-name>" was either incorrectly specified or
  does not exist in the database.

- A qualified reference was made, and the qualifier
  was incorrectly specified.

- The user's function path does not contain the
  schema to which the desired function belongs, and
  an unqualified reference was used.

- The wrong number of arguments were included.

- The right number of arguments were included, but
  the data types of one or more of the arguments is
  incorrect.

- The function does not exist in the database with a
  create timestamp earlier that the time the package
  was bound (applies to static Data Definition
  Language (DDL) statements).

Action: Fix the problem and retry. This could involve
catalog access, change to the statement, the addition of
new functions, and/or change to the function path.

sqlcode: -440

sqlstate: 42884

SQL0441N

**Invalid use of DISTINCT with scalar function
"<function-name>".**

Cause: The keyword DISTINCT was detected within
the parentheses in a reference to function
"<function-name>", and the function has been resolved
as a scalar function. Use of the keyword DISTINCT with
a scalar function is invalid.

Action: If a scalar function is being used, then remove
the keyword DISTINCT. It is invalid for a scalar function.

If a column function is being used, then there is a
problem with function resolution. Check your function
path to see if the desired function is in one of the
schemas, and also check the SYSFUNCTIONS catalog
for the spelling of the function name and the number and
types of parameters.

Correct the error and try again.

sqlcode: -441

sqlstate: 42601

SQL0442N

Error in referring to function "<function-name>". The maximum number of allowable arguments (90) has been exceeded.

Cause: Too many arguments were specified in the reference to function "<function-name>". The maximum allowable is 90.

Action: Correct the statement by ensuring that the correct number of arguments has been used and try again.

sqlcode: -442

sqlstate: 54023

SQL0443N

User defined function "<function-name>" (specific name "<specific-name>") has returned an error SQLSTATE with diagnostic text "<text>".

Cause: An SQLSTATE of the form 38xxx was returned to DB2 by User Defined Function (UDF) "<function-name>" (specific name "<specific-name>"), along with message text "<text>".

Action: The user will need to understand the meaning of the error. See your Database Administrator, or the author of the UDF.

Errors that are detected by the IBM supplied functions in the SYSFUN schema all return the SQLSTATE 38552. The message text portion of the message is of the form:

`SYSFUN:nn`

where nn is a reason code meaning:

01 Numeric value out of range
02 Division by zero
03 Arithmetic overflow or underflow
04 Invalid date format
05 Invalid time format
06 Invalid timestamp format
07 Invalid character representation of a timestamp duration
08 Invalid interval type (must be one of 1, 2, 4, 8, 16, 32, 64, 128, 256)
09 String too long
10 Length or position in string function out of range

11 Invalid character representation of a floating point number

sqlcode: -443

sqlstate: 38xxx (the SQLSTATE returned by the UDF).

SQL0444N

User defined function "<function-name>" (specific name "<specific-name>") is implemented with code in library or path "<library-or-path>", function "<function-code-id>" which cannot be accessed. Reason code: "<code>".

Cause: The DBMS is trying to access the body of the code that implements user defined function "<function-name>" (specific name "<specific-name>"), and cannot access it for the reason given by reason code "<code>" (the codes are listed below). The file implementing the function is identified by "<library-or-path>", and the function by "<function-code-id>".

(Note that these last two tokens may be truncated, due to limitations in the total token length which can be provided. If this happens, then the definition of the function in the catalogs may need to be accessed in order to determine the full library or path and function code id that were defined for the function.)

Action: Given for each reason code:

1   Path name "<library-or-path>" is longer than the maximum (255 bytes). Either the function definition needs to be changed to specify a shorter path, or the DB2 instance path name is too long. Look at the catalog definition to determine which is the case. It may be necessary to move the function body to a directory with a shorter path name.

2   The DB2 instance path name could not be retrieved from DB2. See your system administrator.

3   The path "<library-or-path>" could not be found. See the function creator or your database administrator. The function definition or the location of the function itself will need to be corrected.

4   The file in "<library-or-path>" could not be found. See the function creator or your database administrator. The function definition or the location of the function may need to be corrected, or the function may need to be re-linked.

5   There was insufficient memory to load the library containing the function. Contact the function creator or your database administrator to make sure the library was correctly linked. The system configuration may need to be changed to make more memory available to DB2.

6   The function "<function-code-id>" could not be found in the module named. See the function creator or your database administrator. The function definition or the function itself will need to be corrected.

7   The symbol given as the function name ("<function-code-id>") is not the name of a valid function in the named library. See the function creator or your database administrator. The function definition or the function itself will need to be corrected.

8   The "load" system function has failed, for other than the reasons given above. It may be that the module was not linked at all, or not linked correctly.

9   There was insufficient memory to resolve the function name "<function-code-id>" in the library identified in "<library-or-path>". Contact the function creator or your database administrator to make sure the library containing the function was correctly linked. The system configuration may need to be changed to make more memory available to DB2 Server.

10 The loadquery system call has failed. This can only happen on unix-based systems, and is a symptom that the database manager itself has not been installed correctly. Contact your system administrator.

11 The agent process is searching for a particular database manager function that should be in the libdb2.a library, and it cannot find it. This can only happen on unix-based systems, and is a symptom that the database manager is not correctly installed. Contact your system administrator.

other   An unidentified system failure has occurred. Note the code, and see your system administrator.

DataJoiner users: if this user defined function is strictly used to represent a remote user defined function (and thus no code is required to reside on DataJoiner), you may want to consider modifying the SQL statement or

statistics to enable this function to be evaluated on the remote data source.

sqlcode: -444

sqlstate: 42724

SQL0445W

Value "<value>" has been truncated.

Cause: The value "<value>" was truncated by a system (built-in) cast or adjustment function, which was called to transform the value in some way. This is a warning situation.

The value being transformed is the output of a user-defined function (UDF), and is being transformed because of a CAST FROM specification in the UDF definition, or because the UDF is sourced on another function and the result needed to be transformed.

Action: Ensure that the output is as expected and that the truncation has not caused any unexpected consequences.

sqlcode: +445

sqlstate: 01004

SQL0447W

The statement contains redundant specifications involving the clause "<clause>".

Cause: The "<clause>" keyword was included more than once in the statement. This is a warning condition.

Action: If the redundancy was intentional, or if it is determined that no harm has resulted, then no response is necessary. The type of "harm" alluded to could include, for example, the omission of some other desired keyword.

sqlcode: +447

sqlstate: 01589

SQL0448N

Error in CREATE statement for user defined function "<function-name>". The maximum number of allowable parameters (90) has been exceeded.

Cause: Too many parameters were specified in the CREATE statement for function "<function-name>".

Action: Change the CREATE statement to include fewer parameters.

sqlcode: -448

sqlstate: 54023

SQL0449N

**The CREATE for user defined function
"<function-name>", contains an invalidly
formatted library/function identification in the
EXTERNAL NAME clause.**

Cause: An error was found in the EXTERNAL NAME
clause of the CREATE statement for user defined
function (UDF) "<function-name>". The rules for the
library/function identification are as follows:

The name takes the form '<a>!<b>' or '<a>'. No blanks
are permitted within the single quotes. <a> is one of the
following:

- the full-path identification of a file (e.g. (on AIX)
  /u/slick/udfs/math or (on OS/2) d:\myfunc\math)
- a file name (e.g. math) assumed to be in 'function'
  directory in the sqllib directory.

If <b> is omitted, the default is the entry point defined
when the named file was linked. If <b> is present, it
identifies the entry point (function) within <a> that will be
invoked as the body of the UDF.

Action: Correct the problem and try again. A possible
cause is the inclusion of a blank, or having the '!' at the
beginning or end of the name.

sqlcode: -449

sqlstate: 42878

SQL0450N

**User-defined function "<function-name>"
(specific name "<specific-name>") has
generated a result value, SQLSTATE value,
message text, or scratchpad which is too long.**

Cause: Upon return from user-defined function (UDF)
"<function-name>" (specific name "<specific-name>"),
DB2 has detected that more bytes were returned than
were allocated for one of the following:

- the result value (based upon the function definition)
- the SQLSTATE value (6 bytes including the null
  terminator)
- the message text (71 bytes including the null
  terminator).
- the scratchpad content (100 bytes, plus 4 byte
  binary length preceding it).

This is not permitted.

Action: See your Database Administrator, or the author
of the UDF.

sqlcode: -450

sqlstate: 39501

SQL0451N

**The "<data-item>" definition, in the CREATE for
user defined function "<function-name>",
contains a data type "<type>" that is not
appropriate for an external function written in
the given language.**

Cause: An error was made in the "<data-item>" part of
the CREATE statement for the user defined function
(UDF) "<function-name>". The user's CREATE
statement contained the invalid type "<type>", or it
contained a user defined type (UDT) which is based on
the invalid type "<type>".

"<data-item>" is a token that identifies the area of the
problem in the CREATE statement. For example,
"PARAMETER 2" or "RETURNS" or "CAST FROM".

Action: Determine which situation has occurred, and
take corrective action. Possible corrective actions
include:

- Changing the function definition to a supported type,
  e.g. from DECIMAL to FLOAT. This may involve a
  change in the UDF body itself, and may also involve
  use of cast functions upon use of the function.

- Creating a new (suitably based) user defined type,
  or changing the definition of an existing UDT.

sqlcode: -451

sqlstate: 42815

SQL0452N

**Unable to access the file referenced by host
variable "<variable-position>". Reason code:
"<reason-code>".**

Cause: An error was encountered attempting to access
or while accessing the file referenced by the "nth" host
variable, where n = "<variable-position>", for the reason
given by "<reason-code>". <variable-position> is set to
0 if the host variable position could not be determined.
The possible reason codes are as follows:

- 01 - The file name length is invalid or the file name
  and/or the path has an invalid format.

- 02 - The file option is invalid. It must have one of
  the following values:

SQL0453N SQL0454N

```
SQL_FILE_READ      -read from an existing file
SQL_FILE_CREATE    -create a new file for write
SQL_FILE_OVERWRITE -overwrite an existing file.
                    If the file does not exist,
                    create the file.
SQL_FILE_APPEND    -append to an existing file.
                    If the file does not exist,
                    create the file.
```

- 03 - The file cannot be found.

- 04 - The SQL_FILE_CREATE option was specified for a file with the same name as an existing file.

- 05 - Access to the file was denied. The user does not have permission to open the file.

- 06 - Access to the file was denied. The file is in use with incompatible modes. Files to be written to are opened in exclusive mode.

- 07 - Disk full was encountered while writing to the file.

- 08 - Unexpected end of file encountered while reading from the file.

- 09 - A media error was encountered while accessing the file.

- 10 - An incomplete or invalid multibyte character was encountered while reading from the file.

- 11 - An error was encountered while converting data from the file code page to the application's graphic character code page.

Action:

For reason code 01, correct the file name length, file name and/or path.

For reason code 02, specify a valid file option.

For reason code 03, ensure that the file specified exists before attempting to access the file.

For reason code 04, either delete the file if it is no longer required or specify a file name that does not currently exist.

For reason code 05, ensure that the user has access (correct file permissions) to the file.

For reason code 06, either use a different file or, if the file must be accessed, modify the application to ensure that the file is not accessed concurrently.

For reason code 07, delete unnecessary files to free up disk space or specify a file that resides on another drive/file system with sufficient disk space. Also, ensure that no operating system or user file size limits were reached. If your application code page uses a multibyte encoding scheme it is possible that only part of the last

character was written, ensure that the file contains only fully formed characters.

For reason code 08, if a file is to be used for input, ensure that the file is not modified before the entire file has been read.

For reason code 09, correct all errors on the media where the file resides.

For reason code 10, ensure that the file contains valid multibyte characters based on the code page of the application or submit the request while running under the same code page as the contents of the file.

For reason code 11, ensure that character conversion support between the code page of the file, for example Japanese EUC, and the graphic code page of the application, for example UCS-2, is installed.

sqlcode: -452

sqlstate: 428A1

SQL0453N

A problem has been identified with the RETURNS clause in the CREATE statement for user defined function "<function-name>".

Cause: A problem casting the result of user defined function (UDF) "<function-name>" has been identified. The CAST FROM data type is not castable to the RETURNS data type, and it should be. See the *SQL Reference* for details on casting between data types.

Action: Change the RETURNS or CAST FROM clause so that the CAST FROM data type is castable to the RETURNS data type.

sqlcode: -453

sqlstate: 42880

SQL0454N

The signature provided in the CREATE statement for user defined function "<function-name>" matches the signature of some other function already existing in the schema.

Cause: The signature consists of the function name ("<function-name>"), the number of parameters defined for the function, and an ordered list of the types of the parameters (without regard to any parameters of the types). In this case there is a function already in the schema having the same signature as the function being created.

Action: Determine if the existing function already provides the functionality desired. If not, then the new function's signature will have to be changed (e.g. change the function name).

sqlcode: -454

sqlstate: 42723

SQL0455N

In the CREATE for user defined function "<function-name>", the schema name "<schema-name1>" provided for the SPECIFIC name does not match the schema name "<schema-name2>" of the function.

Cause: If the SPECIFIC name is specified as a two part name, the "<schema-name1>" portion must be the same as the "<schema-name2>" portion of the "<function-name>". Note that the "<schema-name2>" portion of "<function-name>" may have been specified directly or it may have defaulted to the authorization ID of the statement.

Action: Correct the statement and try again.

sqlcode: -455

sqlstate: 42882

SQL0456N

In the CREATE of user defined function "<function-name>", the SPECIFIC name "<specific-name>" already exists in the schema.

Cause: The user has supplied an explicit SPECIFIC name "<specific-name>" in the CREATE statement for function "<function-name>", but this name already exists as the SPECIFIC name for a function in the schema.

Action: Choose a new SPECIFIC name and try again.

sqlcode: -456

sqlstate: 42710

SQL0457N

A function or distinct type cannot be called "<name>" since it is reserved for system use.

Cause: The user defined function or distinct type cannot be created because the name selected is reserved for use by the system. Names that cannot be used for functions or distinct types include:

```
"=","<",">",">=","<=","&=","&>",,"&<",
"!=","!>","!<","<>", SOME, ANY, ALL, NOT,
AND, OR, BETWEEN, NULL, LIKE, EXISTS, IN,
UNIQUE, OVERLAPS, SIMILAR, and MATCH.
```

Action: Select a name for the function or distinct type that is not reserved for system use.

sqlcode: -457

sqlstate: 42939

SQL0458N

In a reference to function "<function-name>" by signature, a matching function could not be found.

Cause: In a reference to function "<function-name>" by signature, no matching function could be found. If a data type is used that can accept a parameter (e.g. CHAR(12)), then the type parameter is optional, i.e. you can specify the parameter (CHAR(12) or omit it (CHAR()). If you specify the parameter, then the DBMS will only accept an exact match on the data type AND the data type parameter. If you omit the parameter, then the DBMS will accept a match on data type only. The CHAR() syntax provides a way to say "don't care about data type parameters in finding a matching function".

Note also that in the DROP FUNCTION and COMMENT ON FUNCTION statements, an unqualified reference is qualified with the statement authorization ID, and this is the schema where the problem can be found. In the SOURCE clause of a CREATE function, the qualification comes from the current function path. In this case, there is no matching function in the entire path.

Action: Possible responses include:

- Changing the function path to include the correct schema.
- Removing parameters from the specifications of data types.
- Using a SPECIFIC name to refer to the function instead of a signature.

sqlcode: -458

sqlstate: 42883

SQL0461N   SQL0465N

SQL0461N

Action: Contact your Database or System administrator.

sqlcode: -465

sqlstate: 58032

SQL0466W

There is one or more result sets associated with a stored procedure.

Cause: This message is returned as a result of issuing a CALL SQL statement. It indicates that the stored procedure associated with the CALL SQL statement has one or more result sets associated with it.

Action: None required.

sqlcode: +466

sqlstate: 01610

SQL0467W

Another result set exists for a stored procedure.

Cause: This message is returned as a result of closing a cursor. It indicates that another result set exists for a stored procedure.

Action: None required.

sqlcode: +467

sqlstate: 01611

SQL0473N

A user defined data type having the same name as a system predefined type cannot be created.

Cause: The name of a data type to be created has an unqualified name that is the same as a system-predefined data type or is BOOLEAN. This is not allowed. Adding delimiters does not make the name valid.

The statement could not be processed.

Action: Correct the statement to use another identifier.

sqlcode: -473

sqlstate: 42918

SQL0475N

The result type "<type-1>" of the SOURCE function cannot be cast to the RETURNS type "<type-2>" of the user defined function "<function-name>".

Cause: In order for the CREATE of a sourced user defined function (UDF) to be valid, the result type ("<type-1>") of the source function must be castable to the RETURNS type ("<type-2>") of the function being created. There is no supported cast between these data types. See the *SQL Reference* for details on casting between data types.

Action: Change the RETURNS data type or the SOURCE function identification so that the result type of the SOURCE function is castable to the RETURNS data type.

sqlcode: -475

sqlstate: 42866

SQL0476N

Reference to function "<function-name>" was made without a signature, but the function is not unique in its schema.

Cause: References to a function without a signature is permitted, but the named function "<function-name>" must be unique in its schema and is not.

Note that in the DROP FUNCTION and COMMENT ON FUNCTION statements, an unqualified reference is qualified with the statement authorization ID, and this is the schema where the problem can be found. In the SOURCE clause of a CREATE FUNCTION, the qualification comes from the current function path. In this case, the first schema in the path containing a function with this name had other functions by the same name.

DataJoiner users: if the statement is a CREATE FUNCTION MAPPING statement, this error indicates that an attempt was made to create a function mapping from one remote function to more than one local function.

Action: Correct the reference by one of the following:

- completing the signature
- using the SPECIFIC name of the desired function
- changing the function path

and try again.

sqlcode: -476

SQL0478N   SQL0487N

sqlstate: 42725

SQL0478N

The object type "<object-type1>" cannot be dropped because there is an object "<object-name>", of type "<object-type2>", which depends on it.

Cause: The base object, which is an object of type "<object-type-1>" (function or distinct type) cannot be dropped because another object depends on it. The dependent object "<object-name>" (which is an object of type "<object-type2>") may be a table (dependent via constraint), a view (dependent via view definition), or another function (dependent because it is sourced on this function, or because it depends on this function to cast a parameter or result). CASCADE semantics are not employed on dependent objects when attempting to drop a function.

It may be that the dependency is indirect. That is, the named object is dependent on another object which is dependent on the object being dropped.

For example:

```
- function F1 is sourced on function F2
- table V1 is defined using F1
- an attempt to drop F2 will fail due to
  the direct dependency on F1 on F2 and the
  indirect dependency of V1 on F2
```

Action: Either do not drop this function or drop the dependent objects first.

sqlcode: -478

sqlstate: 42893

SQL0483N

In the CREATE for user defined function "<function-name>" statement, the number of parameters does not match the number of parameters of the SOURCE function.

Cause: An attempt is being made to CREATE a user defined function "<function-name>" which is sourced on another function. One of the following situations has been identified:

- The SOURCE clause uses a function-name (input parameter list) to identify the source function, and

the number of types in the list is different from the number of parameters of the function being created.

- The SOURCE clause uses different syntax to identify the source function, and the number of types of that function is different from the number of parameters of the function being created.

Action: The number of parameters for the SOURCE function and for the function being created must be the same. The identification of the SOURCE function needs to be changed, to

- fix the input parameter list
- correct the function name or function specific name to identify the proper function.

It is also possible that the function path needs to be corrected in order for correct function resolution to occur.

sqlcode: -483

sqlstate: 42885

SQL0486N

The BOOLEAN data type is currently only supported internally.

Cause: One or more of the data types in the statement is BOOLEAN. This is not supported in the current version of DB2.

Action: Change the data type(s) then resubmit the statement.

sqlcode: -486

sqlstate: 42991

SQL0487N

User defined function "<function-name>" (specific name "<specific-name>") attempted to execute an SQL statement.

Cause: The program used to implement the body of a user defined function is not allowed to execute SQL statements. This user defined function "<function-name>" (specific name "<specific-name>") contains SQL statements.

Action: Remove any SQL statements then recompile the program.

sqlcode: -487

sqlstate: 38502

The function "<function-name>" in a SELECT or VALUES list item has produced a BOOLEAN result.

Cause: The function "<function-name>" is defined for use as a predicate, returning a boolean result. Such a result is not valid in a select list.

The statement cannot be processed.

Action: Correct the function name or remove the use of the function.

sqlcode: -489

sqlstate: 42844

─── **SQL0491N** ───

The CREATE statement for user defined function "<function-name>" must have a RETURNS clause, and either the EXTERNAL clause (with other required keywords) or the SOURCE clause.

Cause: A required clause is missing in the CREATE for function "<function-name>". If EXTERNAL was specified, the following must also be specified: LANGUAGE, PARAMETER STYLE, VARIANT or NOT VARIANT, NO SQL, and EXTERNAL ACTION or NO EXTERNAL ACTION.

Action: Add the missing clause, then try again.

sqlcode: -491

sqlstate: 42601

─── **SQL0492N** ───

There is a problem in the CREATE for user defined function "<function-name>", with parameter number "<number>". It may involve a mismatch with a SOURCE function.

Cause: The parameter in position "<number>" of function "<function-name>" is in error and the CREATE cannot be performed. The parameter in position "<number>" of the source function is not castable to the corresponding parameter of the function being created.

Action: Possible corrections include:

- Identify a different source function.

- Change the data type of the parameter of the function being created so that the data type of the source function can be cast to this data type.

sqlcode: -492

sqlstate: 42879

─── **SQL0493N** ───

User defined function "<function-name>" (specific name "<specific-name>") has returned a date, time or timestamp value which is syntactically or numerically invalid.

Cause: The body of user defined function (UDF) "<function-name>" (specific name "<specific-name>") has returned an invalid date, time or timestamp value.

An example of a syntactically invalid date value is '1994-12*25': the '*' should be a '-'. An example of a numerically invalid time value is '11.71.22': there is no 71st minute in the hour.

Action: The UDF needs fixing. See your DBA, or the author of the UDF.

sqlcode: -493

sqlstate: 22007

─── **SQL0499N** ───

Cursor "<cursor-name>" has already been assigned to this or another result set from procedure "<procedure-name>".

Cause: An attempt was made to assign a cursor to a result set but multiple cursors have been allocated for procedure "<procedure-name>".

Action: Determine if the target result set was previously assigned to a cursor. If multiple cursors have been allocated for procedure "<procedure-name>" ensure that only one cursor is used to process the result sets of a stored procedure.

sqlcode: -499

sqlstate: 24516

─── **SQL0501N** ───

The cursor specified in a FETCH or CLOSE statement is not open.

Cause: The program attempted to either: (1) FETCH using a cursor, or (2) CLOSE a cursor at a time when the specified cursor was not open.

The statement cannot be processed.

Action: Check for a previous message (SQLCODE) that may have closed the cursor. Note that after the cursor is closed, any fetches or close cursor statements receive SQLCODE -501.

SQL0502N SQL0507N

If no previous SQLCODEs have been issued, correct the application program to ensure that the cursor is open when the FETCH or CLOSE statement is executed.

sqlcode: -501

sqlstate: 24501

SQL0502N

The cursor specified in an OPEN statement is already open.

Cause: The program attempted to execute an OPEN statement for an open cursor.

The statement cannot be processed. The cursor was unchanged.

Action: Correct the application program to ensure it does not attempt to execute an OPEN statement for a cursor already open.

sqlcode: -502

sqlstate: 24502

SQL0503N

A column cannot be updated because it is not identified in the FOR UPDATE clause of the SELECT statement of the cursor.

Cause: Using a cursor, the program attempted to update a value in a table column that was not identified in the FOR UPDATE clause in the cursor declaration or the prepared SELECT statement.

Any column to be updated must be identified in the FOR UPDATE clause of the cursor declaration.

The statement cannot be processed.

Action: Correct the application program. If the column requires updating, add its name to the FOR UPDATE clause of the cursor declaration.

sqlcode: -503

sqlstate: 42912

SQL0504N

The cursor "<name>" is not defined.

Cause: An UPDATE or DELETE WHERE CURRENT OF "<name>" was specified, but the cursor "<name>" was not declared in the application program.

The statement cannot be processed.

Action: Ensure the completeness of the application program and correct spelling errors in the cursor names.

sqlcode: -504

sqlstate: 34000

SQL0505N

The cursor "<name>" is already defined.

Cause: The cursor name specified in the DECLARE statement has already been declared.

The statement cannot be processed.

Action: Ensure the name is spelled correctly.

SQL0507N

The cursor specified in the UPDATE or DELETE statement is not open.

Cause: The program attempted to execute an UPDATE or DELETE WHERE CURRENT OF cursor statement when the specified cursor was not open.

The statement cannot be processed. No update or delete was performed.

Action: Check for a previous message (SQLCODE) that may have closed the cursor. Note that after the cursor is closed, any fetches or close cursor statements receive SQLCODE -501 and any updates or deletes receive SQLCODE -507. Correct the logic of the application program to ensure that the specified cursor is open at the time the UPDATE or DELETE statement is executed.

sqlcode: -507

sqlstate: 24501

SQL0508N

The cursor specified in the UPDATE or DELETE statement is not positioned on a row.

Cause: The program attempted to execute an UPDATE or DELETE WHERE CURRENT OF cursor statement when the specified cursor was not positioned on an object table row. The cursor must be positioned on the row to be updated or deleted.

The cursor is no longer positioned on a row if the row is deleted.

DataJoiner users: the record in a remote data source has been updated and/or deleted by another application (or a different cursor within this application) and the record no longer exists.

The statement cannot be processed. No data is updated or deleted.

Action: Correct the logic of the application program to ensure that the cursor is correctly positioned on the intended row of the object table before the UPDATE or DELETE statement is executed. Note that the cursor is not positioned on a row if FETCH returned message SQL0100W (SQLCODE = 100).

sqlcode: -508

sqlstate: 24504

SQL0509N

The table specified in the UPDATE or DELETE statement is not the same table specified in the SELECT for the cursor.

Cause: The program attempted to execute an UPDATE or DELETE WHERE CURRENT OF cursor statement where the table named did not match the name of the table specified in the SELECT statement that declared the cursor.

The statement cannot be processed.

Action: Correct the application program to ensure that the table identified in the UPDATE or DELETE statement is the same table identified in the cursor declaration.

sqlcode: -509

sqlstate: 42827

SQL0510N

UPDATE or DELETE is not allowed against the specified cursor.

Cause: The program attempted to execute an UPDATE or DELETE WHERE CURRENT OF cursor statement against a table or view definition that does not permit the requested update or delete operation. For example, this error can occur in a delete from a read-only view or in an update where the cursor was not defined with the FOR UPDATE clause.

On the database manager the view is read-only if the SELECT statement contains any of the following:

- The DISTINCT keyword
- A column function in the SELECT list
- A GROUP BY or HAVING clause
- A FROM clause that identifies one of the following:
    - More than one table or view
    - A read-only view (READONLY column of SYSCAT.SYSVIEWS is set to 'Y')
    - A table or view that is also identified in the FROM clause of a subquery of the SELECT statement. (NOTE: This is a restriction only in releases of DB2 prior to Version 2).
- A set operator (other than UNION ALL).

Note that these conditions do not apply to subqueries of the SELECT statement.

The cursor is declared with a FOR FETCH ONLY or ORDER BY clause.

The cursor is ambiguous and the BLOCKING ALL bind option was specified.

The statement cannot be processed.

Action: If the database manager is failing the statement and the cursor is based on a read-only SELECT or VALUES statement, do not issue any update or delete statements against it.

If the database manager is failing the statement and the cursor is not based on a read-only SELECT or VALUES statement and is defined with a FOR FETCH ONLY or ORDER BY clause, either remove this clause from the cursor definition or do not issue any update or delete statements.

If the database manager is failing the statement and the cursor cannot be determined to be either fetch only or updateable from its definition or context, rebind the program with either the BLOCKING NO or BLOCKING UNAMBIG bind option.

DataJoiner users: isolate the problem to the data source failing the request (see the *Problem Determination Guide*

for procedures to follow to identify the failing data source). If a data source is failing the request, examine the restrictions for that data source to determine the cause of the problem and its solution. If the restriction exists on a data source, see the SQL reference manual for that data source to determine why the object is not updatable.

sqlcode: -510

sqlstate: 42828

SQL0511N

The FOR UPDATE clause is not allowed because the table specified by the cursor cannot be modified.

Cause: The result table of the SELECT or VALUES statement cannot be updated.

On the database manager, the result table is read-only if the cursor is based on a VALUES statement or the SELECT statement contains any of the following:

- The DISTINCT keyword
- A column function in the SELECT list
- A GROUP BY, HAVING, or ORDER BY clause
- A FROM clause that identifies one of the following:
    - More than one table or view
    - A read-only view
    - A table or view that is also identified in the FROM clause of a subquery of the SELECT statement. (NOTE: This is only a restriction on version of DB2 prior to Version 2).
- A set operator (other than UNION ALL).

Note that these conditions do not apply to subqueries of the SELECT statement.

The statement cannot be processed.

Action: Do not perform updates on the result table as specified.

DataJoiner users: isolate the problem to the data source failing the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source). If a data source is failing the request, examine the restrictions for that data source to determine the cause of the problem and its solution. If the restriction exists on a data source, see the SQL reference manual for that data source to determine why the object is not updatable.

sqlcode: -511

sqlstate: 42829

SQL0513W

The SQL statement will modify an entire table or view.

Cause: The UPDATE or DELETE statement does not contain a WHERE clause so all rows of the table or view are modified if this statement is executed.

The statement is accepted.

DataJoiner users: not all data sources report this warning condition. DataJoiner attempts to issue this warning whenever the condition exists, but there is no guarantee that DataJoiner can always detect this condition. Do not rely on this warning to preclude UPDATE/DELETE operations from affecting an entire table or view.

Action: Ensure that you intend to modify the entire table or view.

SQL0514N

The cursor "<name>" is not in a prepared state.

Cause: The application program tried to use a cursor "<name>" that is not in a prepared state. The cursor is associated with a statement that either (1) was never prepared, (2) was made not valid by a ROLLBACK, or (3) was made not valid by either an explicit or implicit rebind of the package.

The statement cannot be processed.

Action: For case (1), prepare the statement named in the DECLARE CURSOR statement for "<name>" before you try to open the cursor. For case (2), do not issue a ROLLBACK until after you finish using the cursor. For case (3), the prepare for the cursor must be re-issued.

sqlcode: -514

sqlstate: 26501

SQL0516N

The DESCRIBE statement does not specify a prepared statement.

Cause: The statement name in the DESCRIBE statement must specify a statement that was prepared in the same database transaction.

The statement cannot be processed.

Action: Verify that the statement name specifies a statement that has been prepared.

sqlcode: -516

sqlstate: 26501

SQL0517N

The cursor "<name>" identifies a prepared statement that is not a SELECT or VALUES statement.

Cause: The cursor "<name>" could not be used as specified because the prepared statement named in the cursor declaration was not a SELECT or VALUES statement.

The statement cannot be processed.

Action: Verify that the statement name is specified correctly in the PREPARE and the DECLARE CURSOR for cursor "<name>" statements. Or correct the program to ensure that only prepared SELECT or VALUES statements are used in association with cursor declarations.

sqlcode: -517

sqlstate: 07005

SQL0518N

The statement named in the EXECUTE statement is not in a prepared state or is a SELECT or VALUES statement.

Cause: The application program tried to EXECUTE a statement that (1) was never prepared, (2) was made not valid by a ROLLBACK, (3) is a SELECT or VALUES statement, or (4) was made not valid by either an explicit or implicit rebind of the package.

The statement cannot be processed.

Action: For case (1), prepare the statement before attempting the EXECUTE. For case (2), either do not issue a ROLLBACK until after you finish using the prepared statement, or prepare the statement again after the ROLLBACK. For case (3), ensure that the statement is not a SELECT or VALUES statement. For case (4), the prepare for the cursor must be re-issued.

sqlcode: -518

sqlstate: 07003

SQL0519N

The PREPARE statement identifies the SELECT or VALUES statement of the open cursor "<name>".

Cause: The application program attempted to prepare the SELECT or VALUES statement for the specified cursor when that cursor is already open.

The statement cannot be prepared. The cursor was not affected.

Action: Correct the application program so it does not attempt to prepare the SELECT or VALUES statement for a cursor that is open.

sqlcode: -519

sqlstate: 24506

SQL0530N

The insert or update value of the FOREIGN KEY "<constraint-name>" is not equal to any value of the primary key of the parent table.

Cause: An UPDATE or INSERT operation attempted to place a value in a foreign key of the object table but this value is not equal to any value of the primary key of the parent table.

When a row is inserted into a dependent table, the insert value of a foreign key must be equal to the value of the primary key of any row of the parent table of the associated relationship.

Similarly, when the value of a foreign key is updated, the update value of a foreign key must be equal to the value of the primary key of any row of the parent table of the associated relationship.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

Some DataJoiner data sources do not provide the appropriate values for "<constraint name>". In these cases the message token will have the following format: "<data source>:UNKNOWN", indicating that the actual value for the specified data source is unknown.

The UPDATE or INSERT statement could not be executed. The contents of the object table are unchanged.

Action: Examine the insert or update value of the foreign key first, and then compare it with each of the

SQL0531N   SQL0533N

primary key values of the parent table to determine and correct the problem.

sqlcode: -530

sqlstate: 23503

SQL0531N

The primary key in a parent row of relationship "<constraint-name>" cannot be updated.

Cause: An UPDATE operation attempted to update a primary key in the row of the object table but the primary key in the specified row has dependent rows associated with it in the "<constraint-name>" association.

The value of a primary key in a parent row cannot be updated if the parent row has any dependent rows.

The UPDATE statement could not be executed. The contents of the object table are unchanged.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

Some DataJoiner data sources do not provide the appropriate values for "<constraint name>". In these cases the message token will have the following format: "<data source>:UNKNOWN", indicating that the actual value for the specified data source is unknown.

Action: Examine the primary key of the object table and the foreign key of the dependent table to determine if the value of the specified row of the primary key should be changed. If this does not show the problem, examine the contents of the object table and the dependent table to determine and correct the problem.

sqlcode: -531

sqlstate: 23001, 23504

SQL0532N

A parent row cannot be deleted because the relationship "<constraint-name>" restricts the deletion.

Cause: A DELETE operation attempted to delete a specified row of the object table but the primary key in the specified row has dependent rows in the "<constraint-name>" association and the delete rule of NO ACTION or RESTRICT is specified for the relationship.

A row of a table cannot be deleted if it has a dependent relationship with a delete rule of NO ACTION or RESTRICT, or if the deletion cascades to a row which has a dependent relationship with a delete rule of NO ACTION or RESTRICT.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

Some DataJoiner data sources do not provide the appropriate values for "<constraint name>". In these cases the appropriate fields contain values (such as "unknown") that indicate that the actual value is unknown.

The DELETE statement could not be executed. The contents of the table are unchanged.

Action: Examine the delete rule for all descendent tables to determine and correct the problem. The specific tables involved can be determined from the relationship "<constraint-name>".

sqlcode: -532

sqlstate: 23001, 23504

SQL0533N

The INSERT statement is not valid because a relationship restricts the result of the fullselect to one row.

Cause: An INSERT operation with a fullselect attempted to insert multiple rows into a table that is a parent and a dependent in the same relationship of a referential constraint.

The fullselect of the INSERT operation should return no more than one row of data.

The INSERT statement could not be executed. The contents of the object table are unchanged.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

Action: Examine the search condition of the fullselect to ensure selection of no more than one row of data.

sqlcode: -533

sqlstate: 21501

SQL0534N

Invalid multiple-row update.

Cause: An UPDATE operation attempted to perform a multiple-row update of a column included in a primary key or unique index.

Multiple-row updates of columns of a primary key or unique index are not supported.

The UPDATE statement could not be executed. The contents of the table are unchanged.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

Action: Ensure the search condition of the UPDATE statement selects only one object table row to update.

sqlcode: -534

sqlstate: 21502

SQL0535N

The DELETE statement is not valid because a self-referencing relationship restricts the deletion to one row.

Cause: A DELETE operation with a WHERE clause attempted to delete multiple rows from a table which is a parent and a dependent in the same relationship of a referential constraint with a RESTRICT or SET NULL delete rule.

The WHERE clause of the DELETE operation should select no more than one row of data.

The DELETE statement could not be executed. The contents of the object table are unchanged.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

Action: Examine the WHERE clause search condition to ensure no more than one row of data is selected.

NOTE: This is only a restriction in releases of DB2 prior to Version 2.

sqlcode: -535

sqlstate: 21504

SQL0536N

The DELETE statement is not valid because table "<name>" can be affected by the operation.

Cause: A DELETE operation was attempted with the indicated table referenced in a subquery.

The indicated table referenced in a subquery of the DELETE statement can be affected because it is either:

- A dependent of the DELETE object table in a relationship with a CASCADE or SET NULL delete rule.
- A dependent of another table in a relationship with a CASCADE or SET NULL delete rule and deletions from the DELETE object table can cascade to that table.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

Some DataJoiner data sources do not provide the appropriate values for "<name>". In these cases the appropriate fields contain values (such as "unknown") that indicate that the actual value is unknown.

The statement could not be processed.

Action: Do not reference a table in a DELETE statement subquery when the table can be affected by the DELETE statement.

NOTE: This error is only applicable to releases of DB2 prior to Version 2 and hosts accessed through DDCS.

sqlcode: -536

sqlstate: 42914

SQL0537N

The PRIMARY KEY or FOREIGN KEY clause identifies column "<name>" more than once.

Cause: The column "<name>" appears more than once in a PRIMARY KEY or FOREIGN KEY clause of a CREATE or ALTER statement.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

Some DataJoiner data sources do not provide the appropriate values for "<name>". In these cases the appropriate fields contain values (such as "unknown") that indicate that the actual value is unknown.

SQL0538N   SQL0541W

The statement cannot be processed.

Action: Specify unique names for each column.

sqlcode: -537

sqlstate: 42709

SQL0538N

FOREIGN KEY "<constraint-name>" does not conform to the description of the primary key of table "<name>".

Cause: The definition of the indicated foreign key does not conform to the primary key description of the table "<name>". Either the keys do not have the same number of columns or the descriptions are not compatible. Column descriptions are compatible if corresponding columns have compatible data types (both columns are numeric, character strings, graphic, date/time, or have the same distinct type).

"<constraint-name>" is the constraint name specified in the FOREIGN KEY clause or, if a constraint name was not specified, the first column name specified in the clause.

DataJoiner users: some DataJoiner data sources do not provide the appropriate values for "<name>" and "<table-name>". In these cases the appropriate fields contain values (such as "unknown") that indicate that the actual value is unknown.

The constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

The statement cannot be processed.

Action: Correct the statement so the foreign key description conforms to the primary key description.

sqlcode: -538

sqlstate: 42830

SQL0539N

Table "<name>" does not have a "<key-type>" key.

Cause: One of the following occurred:

- The table "<name>" was specified as a parent in a FOREIGN KEY clause but the table is not defined as a parent because it does not have a primary key.
- The ALTER TABLE statement attempted to drop the primary key for the table "<name>" but the table does not have a primary key.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

Some DataJoiner data sources do not provide the appropriate values for "<name>" and "<key-type>". In these cases the appropriate fields contain values (such as "unknown") that indicate that the actual value is unknown.

The statement cannot be processed. A system catalog cannot be defined as a parent in a referential constraint.

Action: When creating a referential constraint, specify the primary key before specifying any foreign keys (constraints).

sqlcode: -539

sqlstate: 42888

SQL0541W

The referential constraint defined by FOREIGN KEY "<name>" is a duplicate referential constraint.

Cause: A referential constraint was not defined because the indicated foreign key clause defines the same foreign key and parent table as another referential constraint.

"<name>" is the constraint name specified in the FOREIGN KEY clause or, if a constraint name was not specified, the first column name specified in the clause.

DataJoiner users: the constraint can exist on DataJoiner (if the child and parent tables exist as tables on DataJoiner), or it can exist on the data source (if the child and parent tables exist on the data source).

SQL0542N

"<name>" cannot be a column of a primary key because it can contain null values.

Cause: The column "<name>" identified in a PRIMARY KEY clause is defined to allow null values.

DataJoiner users: some DataJoiner data sources do not provide the appropriate values for "<name>". In these cases the appropriate fields contain values (such as "unknown") that indicate that the actual value is unknown.

The statement cannot be processed.

Action: Correct the column or primary key definition.

sqlcode: -542

sqlstate: 42831

SQL0543N

A row in a parent table cannot be deleted because the check constraint "<constraint-name>" restricts the deletion.

Cause: The delete operation cannot be executed because the target table is a parent table and is connected with a referential constraint to a dependent table with a delete rule of SET NULL. However, a check constraint defined on the dependent table restricts the column from containing a null value.

The statement cannot be processed.

Action: Examine the foreign key and its delete rule in the dependent table and the conflicting check constraint. Change either the delete rule or the check constraint so that they do not conflict with each other.

sqlcode: -543

sqlstate: 23511

SQL0544N

The check constraint "<constraint-name>" cannot be added because the table contains a row that violates the constraint.

Cause: At least one existing row in the table violates the check constraint that is be added in the ALTER TABLE statement.

The statement cannot be processed.

Action: Examine the check constraint definition that was specified in the ALTER TABLE statement and the data in the table to determine why there is a violation of

the constraint. Change either the check constraint or the data so that the constraint is not violated.

sqlcode: -544

sqlstate: 23512

SQL0545N

The requested operation is not allowed because a row does not satisfy the check constraint "<constraint-name>".

Cause: Check constraint violations can occur on either INSERT or UPDATE operations. The resulting row violated the check constraint definition on that table.

The statement cannot be processed.

Action: Examine the data and the check constraint definition in the catalog view SYSCAT.CHECKS to determine why the INSERT or UPDATE statement failed. Change the data so that the constraint is not violated.

sqlcode: -545

sqlstate: 23513

SQL0546N

The check constraint "<constraint-name>" is invalid.

Cause: A check constraint in the CREATE TABLE or ALTER TABLE statement is invalid for one or more of the following reasons:

- the constraint definition contains a subquery
- the constraint definition contains a column function
- the constraint definition contains a host variable
- the constraint definition contains a parameter marker
- the constraint definition contains a special register
- the constraint definition contains a variant user defined function
- the constraint definition contains a user defined function with external action
- the constraint definition contains a user defined function with the scratchpad option
- the check constraint is part of a column definition, and its check-condition contains a reference to a column name other than the column being defined.

The statement cannot be processed.

Action: The user response is one of the following, depending on the cause of the error:

- change the check constraint so that it does not include the listed item.

SQL0548N SQL0552N

- move the check constraint definition outside the
  column definition, so that it becomes a table level
  constraint definition.

sqlcode: -546

sqlstate: 42621

SQL0548N

A check constraint that is defined with
"<check-condition-element>" is invalid.

Cause: A check constraint in the CREATE TABLE or
ALTER TABLE statement is invalid for one or more of
the following reasons:

- the constraint definition contains a subquery
- the constraint definition contains a column function
- the constraint definition contains a host variable
- the constraint definition contains a parameter
  marker
- the constraint definition contains a special register
- the constraint definition contains a variant user
  defined function
- the constraint definition contains a user defined
  function with external action
- the constraint definition contains a user defined
  function with the scratchpad option
- the check constraint is part of a column definition,
  and its check-condition contains a reference to a
  column name other than the column being defined.

The token within the text of the error message lists the
invalid item.

The statement cannot be processed.

Action: The user response is one of the following,
depending on the cause of the error:

- change the check constraint so that it does not
  include the listed item.
- move the check constraint definition outside the
  column definition, so that it becomes a table level
  constraint definition.

sqlcode: -548

sqlstate: 42621.

SQL0551N

"<authorization-ID>" does not have the privilege
to perform operation "<operation>" on object
"<name>".

Cause: Authorization ID "<authorization-ID>" attempted
to perform the specified "<operation>" on "<name>"
without the proper authorization.

If creating or altering a table with referential constraints,
this message (SQLCODE) can indicate that the user
does not have the REFERENCES privilege to create or
drop a FOREIGN KEY. In this case the "<operation>" is
"REFERENCES" and the "<name>" is the object the
constraint references.

DataJoiner users: if this message is returned when the
user is changing the remote_pw column of the
SYSCAT.REMOTEUSERS view, the user is not
authorized to change a password for another user. The
user performing the alter operation must have either
SYSADM authority or an authorization ID (the value in
the USER special register) that matches the value of the
authid column in the rows being updated. Some data
sources do not provide the appropriate values for
"<authid>", <operation>, and <name>. In these cases
the message tokens will have the following format:
"<data source> AUTHID:UNKNOWN", "UNKNOWN", and
"<data source>:TABLE/VIEW", indicating that the actual
values for the authid, operation, and name at the
specified data source are not known.

The statement cannot be processed.

Action: Ensure that "<authorization-ID>" has the
authorization necessary to perform the operation.

DataJoiner users: this authorization can be at
DataJoiner, the data source, or both.

sqlcode: -551

sqlstate: 42501

SQL0552N

"<authorization-ID>" does not have the privilege
to perform operation "<operation>".

Cause: Authorization ID "<authorization-ID>" attempted
to perform the specified "<operation>" without the proper
authorization.

DataJoiner users: some DataJoiner data sources do not
provide the appropriate values for "<authorization-ID>"
and <operation>. In these cases the message tokens
will have the following format: "<data source>
AUTHID:UNKNOWN", and "UNKNOWN" indicating that

the actual values for authid and operation at the specified data source are not known.

The statement cannot be processed.

Action: Ensure that "<authorization-ID>" has the authorization to perform the operation.

DataJoiner users: this authorization can be at DataJoiner, the data source, or both.

sqlcode: -552

sqlstate: 42502

SQL0553N

An object cannot be created with the schema name "<schema-name>".

Cause: The reason the schema name "<schema-name>" is invalid depends on the type of object that is being created.

- Objects of types that existed in releases prior to DB2 Version 2 (tables, views, indexes and packages) cannot be created with the schema name SYSCAT, SYSSTAT or SYSIBM. It is strongly advised that schema names should not start with SYS since additional schemas starting with these letters may be reserved for exclusive use of DB2 in the future.
- Objects of types that are being introduced starting with DB2 Version 2 (user defined functions, distinct types, triggers, and aliases) cannot be created with any schema name that starts with the letters SYS.

The statement cannot be processed.

Action: Use a valid schema name or remove the explicit schema name and rerun the statement.

sqlcode: -553

sqlstate: 42939

SQL0554N

An authorization ID cannot grant a privilege to itself.

Cause: An authorization ID attempted to execute a GRANT statement where the authorization ID itself appears as one of the entries in the authorization ID list to which privileges are to be granted.

The statement cannot be processed.

Action: Remove the authorization ID from the list.

sqlcode: -554

sqlstate: 42502

SQL0555N

An authorization ID cannot revoke a privilege from itself.

Cause: An authorization ID attempted to execute a REVOKE statement where the authorization ID itself appears as one of the entries in the authorization ID list from which privileges are to be revoked.

The statement cannot be processed.

Action: Remove the authorization ID from the list.

sqlcode: -555

sqlstate: 42502

SQL0556N

An attempt to revoke a privilege from "<authorization-ID>" was denied because "<authorization-ID>" does not hold this privilege.

Cause: The privilege cannot be revoked because "<authorization-ID>" does not hold the privilege.

The statement cannot be processed.

Action: Change the REVOKE statement to conform to the REVOKE rules and resubmit the statement. Ensure that when a REVOKE statement lists several privileges to be revoked and a list of authorization IDs that each authorization ID possesses at least one of the specified privileges.

sqlcode: -556

sqlstate: 42504

SQL0557N

The specified combination of privileges cannot be granted or revoked.

Cause: One of the following occurred:

- The GRANT or REVOKE statement contains a combination of privileges that are of different classes. The privileges must all be of one class. Examples are DATABASE, PLAN, or TABLE.
- The GRANT statement attempted to grant a privilege for a view that is not allowed. ALTER, INDEX and REFERENCES cannot be granted for a view.

The statement cannot be processed.

Action: Correct and resubmit the statement.

SQL0558N SQL0573N

sqlcode: -557

sqlstate: 42852

SQL0558N

An attempt to revoke a privilege from "<authorization-ID>" was denied because "<authorization-ID>" would still hold "<control>" authority.

Cause: "<authorization-ID>" holds the "<control>" privilege. The privilege to be revoked is implicit in the "<control>" privilege, so it cannot be revoked unless the "<control>" privilege is also revoked.

The valid values for "<control>" are the following:

- DBADM
- CONTROL

The statement cannot be executed. No privileges are revoked.

Action: Revoke the "<control>" privilege, if desired.

sqlcode: -558

sqlstate: 42504

SQL0562N

The specified database privileges cannot be granted to PUBLIC (public).

Cause: The GRANT statement attempted to grant a database privilege to the reserved authorization ID PUBLIC (public). DBADM authority cannot be granted to PUBLIC (public).

The statement cannot be processed.

Action: The implied function is not supported.

sqlcode: -562

sqlstate: 42508

SQL0567N

"<authorization-ID>" is not a valid authorization ID.

Cause: The authorization ID specified by "<authorization-ID>" is not valid for one of the following reasons:

- It begins with "SYS", "sys", "IBM", "ibm", "SQL", "sql", or 0 through 9
- It contains characters other than a through z, A through Z, 0 through 9, and the three special characters (#,@.$).

- It is delimited and contains lower case letters.
- It is GUESTS, ADMINS, USERS, or LOCAL.
- It is PUBLIC preceded by the keyword USER or GROUP in a GRANT or REVOKE statement.

The statement cannot be processed.

Action: Correct the invalid authorization ID.

sqlcode: -567

sqlstate: 42602

SQL0569N

The GRANT/REVOKE statement failed because "<authorization-name>" identifies both a user and a group in the system.

Cause: The GRANT or REVOKE statement specified an authorization name that identifies both a user and a group in the system, and did not explicitly specify the USER or GROUP keyword in the statement. The statement is therefore ambiguous.

Action: Change the statement to explicitly specify the USER or GROUP keywords to uniquely identify the required action.

sqlcode: -569

sqlstate: 56092

SQL0572N

Package "<pkgname>" is inoperative.

Cause: The package "<pkgname>" is marked as inoperative and must be explicitly rebound before it can be used. This package cannot be used because one or more user-defined functions that it depends upon have been dropped.

Action: Explicitly rebind the named package using either the REBIND or the BIND command.

sqlcode: -572

sqlstate: 51028

SQL0573N

A column list specified in the references clause of constraint "<constraint-name>" does not identify the primary key of the parent table "<table-name>".

Cause: A list of column names was specified in the references clause for the constraint "<constraint-name>" that does not match the column names of the primary key of the referenced table "<table-name>".

The statement cannot be processed.

Action: Correct the column list in the references clause or add the primary key to the referenced table.

sqlcode: -573

sqlstate: 42890

SQL0574N

DEFAULT value is not valid for column "<column-name>" in table "<table-name>". Reason code: "<reason-code>".

Cause: The DEFAULT value for column "<column-name>" in table "<table-name>" is not valid. The possible reason codes are as follows:

1 The value is not assignable to the column because the constant does not conform to the format for a constant of that data type, the value has an incorrect length or precision, or the function returns an incorrect data type.

2 A floating-point constant is specified and the column is not a floating point data type

3 A decimal constant is specified and non-zero digits would be truncated when assigned to the column

4 The value is more than 254 bytes, including quotes for strings, introducer characters such as the X for a hex constant, fully qualified function names, and parentheses. Non-significant blanks in the value are ignored. In an unequal code page environment, the value may be more than 254 bytes due to expansion of the string in the database code page.

5 The USER special register is specified and the length attribute of the character string data type is less than 8.

6 A datetime special register (CURRENT DATE, CURRENT TIME, or CURRENT TIMESTAMP) is specified and does not match the data type of the column.

7 A function was specified that is not supported. The specified function must be system-generated cast function or one of the built-in functions BLOB, DATE, TIME, or TIMESTAMP.

8 The argument to a datetime function was not a character string constant or the corresponding datetime special register.

9 A system-generated cast function was specified and the column is not defined with a user-defined distinct type.

<0 Any reason code that is less than zero is an SQLCODE. The error in the DEFAULT value specification can be determined by checking the error message corresponding to this SQLCODE.

The statement cannot be processed.

Action: Correct the DEFAULT value based on the reason code that was returned.

sqlcode: -574

sqlstate: 42894

SQL0575N

View "<name>" cannot be used because it has been marked inoperative.

Cause: The view "<name>" has been marked inoperative because a table, view, alias, or privilege upon which it is dependent has been removed. The view cannot be used by any SQL statement other than one of the following:

- COMMENT ON
- DROP VIEW
- CREATE ALIAS
- CREATE VIEW

The statement cannot be processed.

Action: Recreate the view by issuing a CREATE VIEW statement using the same view definition as the inoperative view.

sqlcode: -575

sqlstate: 51024

SQL0576N

Alias "<name>" cannot be created for "<name2>" as it would result in a repetitive alias chain.

Cause: The alias definition of "<name>" on "<name2>" would have resulted in a repetitive alias chain which could never be resolved. For example, "alias A refers to alias B which refers to alias A" is a repetitive alias chain which could never be resolved.

The statement cannot be processed.

Action: Change the alias definition for "<name>" or revise the definition of one of the other alias definitions in the alias chain to avoid a repetitive chain.

sqlcode: -576

sqlstate: 42916

SQL0580N SQL0583N

SQL0580N

The result-expressions of a CASE expression cannot all be NULL.

Cause: There is a CASE expression in the statement that has all the result-expressions (expressions following THEN and ELSE keywords) coded with the keyword NULL.

The statement cannot be processed.

Action: Change the CASE expression to include at least one result-expression that is other than the keyword NULL.

sqlcode: -580

sqlstate: 42625

SQL0581N

The data types of the result-expressions of a CASE expression are not compatible.

Cause: There is a CASE expression in the statement that has result-expressions (expressions following THEN and ELSE keywords) that are not compatible.

The data type of a CASE expression is determined using the "Rules for Result Data Types" on the result-expressions. The data types of the result-expressions may be incompatible for one of the following reasons:

- they are not all character data types
- they are not all numeric data types
- They are not all date data types
- They are not all time data types
- They are not all timestamp data types
- They are not all the same user-defined distinct data types

The statement cannot be processed.

Action: Correct the result-expressions so that they are compatible.

sqlcode: -581

sqlstate: 42804

SQL0582N

A CASE expression in a select list, VALUES clause, IN predicate, or ORDER BY clause cannot include a quantified predicate, an IN predicate using a fullselect or an EXISTS predicate.

Cause: A search condition of a CASE expression is:

- a quantified predicate (one using SOME, ANY, or ALL),
- an IN predicate using a fullselect or
- an EXISTS predicate

and the CASE expression is part of:

- a select list
- a VALUES clause
- an IN predicate
- a GROUP BY clause or
- an ORDER BY clause.

Such CASE expressions are not supported.

The statement cannot be processed.

Action: Remove the use of the quantified predicate, IN predicate, or EXISTS predicate from the CASE expression. In some cases, it may be possible to write the query so that the CASE expression is not part of the select list or an IN predicate.

sqlcode: -582

sqlstate: 42625

SQL0583N

The use of function "<function.name>" is invalid because it is variant or has an external action.

Cause: The function "<function-name>" is defined as a variant function or a function with an external action. This type of function is not supported in the context in which it is used. The contexts in which these are not valid are:

- as the first operand of a BETWEEN predicate

- in the expression prior to the first WHEN keyword in a simple-case-expression.

- in an expression of a GROUP BY clause

- in an expression of an ORDER BY clause (external action only)

The statement cannot be processed.

Action: If the use of a variant or external action function was not intended, substitute a function without

these characteristics. If the behaviour associated with the variant or external action function is intentional, use the alternate form of the statements that make that intent explicit.

- Instead of the BETWEEN predicate, use the corresponding statement using the equivalent combination of comparison predicates (a>=b and a<=c instead of a between b and c).

- Instead of a simple-when-clause, use the corresponding searched-when-clause where the function would get specified in each search-condition.

- Remove the variant or external action function from the GROUP BY clause. If grouping is desired on a column of the result that is based on a variant or external action function use a nested table expression or a common table expression to first provide a result table with the expression as a column of the result.

- Remove the external action function from the ORDER BY clause. If the column is part of the result set of the query, change the expression in the ORDER BY clause to the simple-integer or simple-column-name form of the sort key.

sqlcode: -583

sqlstate: 42845

SQL0584N

Invalid use of NULL or DEFAULT.

Cause: DEFAULT can only be used in a VALUES clause that is part of an INSERT statement.

A VALUES clause that is not part of an INSERT statement must have a value other than NULL in at least one row for each column.

If DEFAULT is used as a column-name in a WHERE or HAVING clause, it must be capitalized and enclosed in double quotes.

DataJoiner users: DEFAULT can not be used in a VALUES clause of an INSERT statement where the object is a nickname.

Action: Substitute a value other than NULL or DEFAULT in the VALUES clause. If DEFAULT is used

as a column-name, capitalize it and enclose it in double quotes.

sqlcode: -584

sqlstate: 42608

SQL0585N

The schema name "<schema-name>" can not appear more than once in the function path.

Cause: The function path includes "<schema-name>" more than once. The function path can only include one occurrence of each schema name.

The statement cannot be processed.

Action: Remove duplicate occurrences of "<schema-name>" from the function path.

sqlcode: -585

sqlstate: 42732

SQL0586N

The total length of the CURRENT FUNCTION PATH special register cannot exceed 254 characters.

Cause: The CURRENT FUNCTION PATH special register is defined as a VARCHAR(254). The content of the string includes each schema name delimited with double quotes and separated from the next schema name by a comma. The total length of the string of all schema names in the CURRENT FUNCTION PATH cannot exceed 254 characters. The SET CURRENT FUNCTION PATH statement or the FUNCPATH option of the PREP or BIND command causing this message would exceed this limit.

The statement or command cannot be processed.

Action: Remove schema names to reduce the total length to fit the 254 character maximum length. If all the schema names are required, it may be necessary to consolidate some user-defined functions so that less schema names are required on the CURRENT FUNCTION PATH.

sqlcode: -586

sqlstate: 42907

SQL0595W SQL0601N

SQL0595W

Isolation level "<requested-level>" has been escalated to "<escalated-level>".

Cause: The isolation level specified is not supported by DB2. It has been escalated to the next higher level of isolation supported by DB2.

Action: To avoid this warning, specify an isolation level which is supported by DB2. DB2 supports isolation levels Repeatable Read (RR), Read Stability (RS), Cursor Stability (CS), and Uncommitted Read (UR).

sqlcode: +595

sqlstate: 01526

SQL0598W

Existing index "<name>" is used as the index for the primary key.

Cause: An index was required for a CREATE TABLE or ALTER TABLE operation that defined a primary key, and the indicated index matches the required index.

When creating a primary index, an index description matches if it identifies the same columns in the same order as the primary index without regard to ascending or descending specifications, and is specified as unique.

The statement is processed successfully.

Action: No action is required.

sqlcode: +598

sqlstate: 01550

SQL0599W

Comparison functions are not created for a distinct type based on a long string data type.

Cause: Comparison functions are not created for a distinct type based on a long string data type (BLOB, CLOB, DBCLOB, LONG VARCHAR, or LONG VARGRAPHIC) since the corresponding functions are not available for these built-in data types.

This is a warning situation. The statement is processed successfully.

Action: No action is required.

sqlcode: +599

sqlstate: 01596

SQL0600N

A function could not be generated because of a duplicate signature.

Cause: While attempting to create a distinct type, the generated cast function by the same name as the distinct type name could not be created because one with the same signature already exists in the schema.

The statement could not be processed.

Action: Choose another name for the distinct type or drop the function with the same name as the distinct type name.

sqlcode: -600

sqlstate: 42710

SQL0601N

The name of the object to be created is identical to the existing name "<name>" of type "<type>".

Cause: The CREATE or ALTER statement tried to create or add an object "<name>" when an object of type "<type>" already exists with that name on the application server or in the same statement.

If "<type>" is FOREIGN KEY, PRIMARY KEY, or CHECK CONSTRAINT, the "<name>" is the constraint name specified in the CREATE or ALTER TABLE statement or generated by the system.

DataJoiner users: some DataJoiner data sources do not provide the appropriate values for the "<name>" and "<type>" message tokens. In these cases, "<name>" and "<type>" will have the following format: "OBJECT:<data source> TABLE/VIEW", and "UNKNOWN", indicating that the actual values at the specified data source are not known.

The statement cannot be processed. No new object is created, and the existing object is not altered or modified.

Action: Either drop the existing object or choose another name for the new object.

DataJoiner users: if the statement is a CREATE FUNCTION MAPPING or a CREATE TYPE MAPPING statement, the user can choose to either specify or not specify a new mapping name. If the user doesn't specify a new name, the system automatically creates a unique one.

sqlcode: -601

sqlstate: 42710

SQL0602N

The CREATE INDEX statement specifies too many columns (16 is the maximum).

Cause: The number of columns specified in the CREATE INDEX statement exceeds the maximum of 16 for the database manager.

DataJoiner users: the limit for other data sources can be different. That limit could have been exceeded. This problem may be detected on DataJoiner, or it may be detected on the data source.

The statement cannot be processed.

Action: Change the index definition to conform to the column limit of 16.

DataJoiner users: change the index definition to conform to the column limit for the data source.

sqlcode: -602

sqlstate: 54008

SQL0603N

A unique index cannot be created because the table contains rows which are duplicates with respect to the values of the identified columns.

Cause: The index defined in the CREATE INDEX statement could not be created as unique, because the specified table already contains rows that duplicate the values of the identified columns.

DataJoiner users: this situation may be detected by DataJoiner or it may be detected by the data source.

The statement cannot be processed. The specified index is not created.

Action: Examine the data to determine if the duplicate data is permitted. Alternatively, consider creating a non-UNIQUE index.

sqlcode: -603

sqlstate: 23515

SQL0604N

The length, precision, or scale attribute for column, distinct type or function "<data-item>" is not valid.

Cause: There is an error in the data type specification in a CREATE or ALTER statement, or a CAST specification. An invalid length, precision or scale attribute may have been specified, or it may be that the data type itself is incorrect or not permitted in this context. The location of the error is given by "<data-item>" as follows:

*   for a CREATE or ALTER TABLE statement, "<data-item>" gives the name of the column containing the error.
*   for a CREATE FUNCTION statement, "<data-item>" is a token that identifies the area of the problem in the statement. For example, "PARAMETER 2" or "RETURNS" or "CAST FROM".
*   for a CREATE DISTINCT TYPE statement, "<data-item>" gives the name of the type being defined.
*   for CAST(expression AS data-type), "<data-item>" is "CAST".

DataJoiner users: if the statement is a CREATE TYPE MAPPING statement, an attempt was made to create a type mapping where a type attribute for either the local data type or the remote data type is not valid. Possible reasons include:

*   The local length/precision is set to 0 or a negative value.
*   The length/precision attribute is specified for data types such as date/time/timestamp, float, or integer.
*   The scale attribute is specified for data types such as character, date/time/timestamp, float, or integer.
*   The FOR BIT DATA clause is specified for a non-character type.
*   The remote precision is set to 0 for remote types other than Informix datetime.
*   An invalid field qualifier is being used in a type mapping for an Informix datetime type.
*   An ending value is lower than the starting value in a precision/scale range.

The statement cannot be processed.

Action: Correct the syntax and try again.

sqlcode: -604

sqlstate: 42611

SQL0605W SQL0613N

SQL0605W

The index was not created because the index "<name>" already exists with the required description.

Cause: A CREATE INDEX operation attempted to create a new index and the indicated index matches the required index.

 For CREATE INDEX, two index descriptions match if they identify the same columns in the same order with the same ascending or descending specifications, and are both specified as unique or the new index is specified as non-unique.

The new index was not created.

Action: No action is required unless the existing index "<name>" is not a suitable index.

sqlcode: +605

sqlstate: 01550

SQL0607N

"<operation>" is not defined for system objects.

Cause: The "<operation>" specified in the SQL statement cannot be performed on system objects. One of the following was attempted:

- DROP or ALTER a system owned object such as a system catalog table, built-in function or built-in data type
- COMMENT ON a system owned built-in function.
- INSERT or DELETE on a system catalog table.
- UPDATE directly on a system catalog table. Some columns of a subset of system catalog tables are updateable. For UPDATE operation on these catalog tables, updateable views in SYSSTAT schema should be used. For a description of updateable catalog views (SYSSTAT views), see the *SQL Reference*.
- CREATE or DROP of an index on a system table
- CREATE of a trigger on a system table
- A non-updateable system table was identified in the FROM clause of a SELECT statement containing a FOR UPDATE clause. For a list of updateable system catalogs see the *SQL Reference*.
- DROP or ALTER of a system table space.

The statement cannot be processed.

Action: Do not attempt to modify any system objects except for those columns of the system catalog tables which are updateable through updateable SYSSTAT views. See the *SQL Reference* for more information.

sqlcode: -607

sqlstate: 42832

SQL0612N

"<name>" is a duplicate column name.

Cause: A statement was issued with the same column name appearing more than once where duplicates are not allowed. Where these column names appears varies depending on the type of statement.

- CREATE TABLE statements cannot have the same column name defined for two columns.
- CREATE VIEW statements or commont table expression definitions cannot have the same column name in the column name list. If no column name list is specified, then the column names of the columns in the select list of the view must be unique.
- ALTER TABLE statement cannot add a column to a table using the name of a column that already exists or is the same as another column being added.
- CREATE INDEX cannot have a column name specified more than once as part of the index key.
- CREATE TRIGGER cannot have a column name specified more than once in the list of columns that cause an update trigger to be activated.

The statement cannot be processed.

Action: Specify unique names for each column as appropriate for the type of statement.

sqlcode: -612

sqlstate: 42711

SQL0613N

The primary key is too long or has too many columns.

Cause: The sum of the internal lengths of the PRIMARY KEY clause columns exceeds 255 or the number of columns exceeds the maximum of 16. Also, a primary key cannot be defined using a LONG VARCHAR column.

The statement cannot be processed.

Action: Modify the primary key definition by eliminating one or more key columns to conform to the column limit of 16 and the key length limit.

sqlcode: -613

sqlstate: 54008

SQL0614N

The index cannot be created because the combined length of the specified columns is more than 255 bytes.

Cause: The index could not be created because the sum of the key column internal lengths would exceed 255. Also, an index cannot be created using a LONG VARCHAR, LONG VARGRAPHIC, or LOB column.

The statement cannot be processed. The specified index was not created.

Action: To modify the index definition, eliminate one or more key columns and reduce the key length to the permitted maximum.

sqlcode: -614

sqlstate: 54008

SQL0624N

Table "<name>" already has a primary key.

Cause: A primary key cannot be defined in an ALTER TABLE statement because the indicated table already has a primary key.

The statement cannot be executed.

Action: A table cannot have more than one primary key.

sqlcode: -624

sqlstate: 42889

SQL0628N

Multiple or conflicting keywords involving the "<clause-type>" clause are present.

Cause: There are several possible reasons why this condition might have been diagnosed for the statement. Exactly which one happened is indicated by the value of "<clause-type>". The possibilities to consider are:

- The keyword may not be specified in the same statement as some other keyword.

- The keyword may be part of a sequence of keywords where the order in which they are specified is not enforced. A keyword in such a sequence may have been specified with the contradicting keyword also specified.

- The keyword may appear more than once with different associated values.

Action: Check that the statement conforms to the syntax and rules defined for the statement. Correct any invalid occurrences of duplicate or conflicting keywords.

sqlcode: -628

sqlstate: 42613

SQL0629N

SET NULL cannot be specified because FOREIGN KEY "<name>" cannot contain null values.

Cause: The SET NULL option of the indicated FOREIGN KEY clause is not valid because no column of the key allows null values.

"<name>" is the constraint name specified in FOREIGN KEY clause or, if a constraint name was not specified, is the first column name specified in the clause.

The statement cannot be processed.

Action: Change either a column of the key to allow null values or change the delete rule.

sqlcode: -629

sqlstate: 42834

SQL0631N

FOREIGN KEY "<name>" is too long or has too many columns.

Cause: The sum of the column internal lengths identified in the FOREIGN KEY clause in a CREATE TABLE statement exceeds 255, or the number of columns identified exceeds 16. Also, a foreign key cannot be defined using a LONG VARCHAR column.

"<name>" is the constraint name specified in the FOREIGN KEY clause or, if a constraint name was not specified, the first column name specified in the clause.

The statement cannot be processed.

Action: To modify the foreign key definition, eliminate one or more key columns and conform to the 16 column limit and the key length limit.

sqlcode: -631

sqlstate: 54008

SQL0632N   SQL0634N

FOREIGN KEY "<name>" is not valid because the table cannot be defined as a dependent of table "<table-name>" because of delete rule restrictions (reason code = "<reason-code>").

Cause: A referential constraint cannot be defined because the object table of the CREATE TABLE or ALTER TABLE statement cannot be defined as a dependent of table "<table-name>" for one of the following reason codes:

- (01) The relationship is self-referencing and a self-referencing relationship already exists with the SET NULL delete rule.
- (02) The relationship forms a cycle of two or more tables that cause the table to be delete connected to itself (all other delete rules in the cycle would be CASCADE).
- (03) The relationship causes the table to be delete connected to the indicated table through multiple relationships and the delete rule of the existing relationship is SET NULL.

The delete rules of the existing relationships cause an error, not the delete rule specified in the FOREIGN KEY clause of the CREATE TABLE or ALTER TABLE statement.

"<name>" is the constraint name specified in the FOREIGN KEY clause or, if a constraint name is not specified, the first column name specified in the clause.

The statement cannot be processed.

Action: If possible, eliminate the particular FOREIGN KEY clause from the CREATE TABLE or ALTER TABLE statement.

sqlcode: -632

sqlstate: 42915

The delete rule of FOREIGN KEY "<name>" must be "<delete-rule>" (reason code = "<reason-code>").

Cause: The delete rule specified in a FOREIGN KEY clause of the CREATE TABLE or ALTER TABLE statement is not valid. The indicated delete rule is require for one of the following reason codes:

- (01) The referential constraint is self-referencing and an existing self-referencing constraint has the

indicated delete rule (NO ACTION, RESTRICT or CASCADE).

- (02) The referential constraint is self-referencing and the table is dependent in a relationship with a delete rule of CASCADE.
- (03) The relationship would cause the table to be delete-connected to the same table through multiple relationships and such relationships must have the same delete rule (NO ACTION, RESTRICT or CASCADE).

"<name>" is the constraint name specified in the FOREIGN KEY clause or, if a constraint name was not specified, the first column name specified in the clause.

The statement cannot be processed.

Action: If possible, change the delete rule.

sqlcode: -633

sqlstate: 42915

The delete rule of FOREIGN KEY "<name>" must not be CASCADE (reason code = "<reason-code>").

Cause: The CASCADE delete rule specified in the FOREIGN KEY clause of the CREATE TABLE or ALTER TABLE statement is not valid for one of the following reason codes:

- (01) A self-referencing constraint exists with a delete rule of SET NULL, NO ACTION or RESTRICT.
- (02) The relationship would form a cycle that would cause a table to be delete connected to itself. One of the existing delete rules in the cycle is not CASCADE, so this relationship may be definable if the delete rule is not CASCADE.
- (03) The relationship would cause another table to be delete-connected to the same table through multiple paths with different delete rules or with delete rule equal to SET NULL.

  "<name>" is the constraint name specified in the FOREIGN KEY clause or, if a constraint name was not specified, the first column name specified in the clause.

The statement cannot be processed.

Action: If possible, change the delete rule.

sqlcode: -634

sqlstate: 42915

SQL0637N

More than one PRIMARY KEY clause or more than one DROP PRIMARY KEY clause was specified.

Cause: The CREATE TABLE statement contains two or more PRIMARY KEY clauses, or the ALTER TABLE statement contains two or more PRIMARY KEY or DROP PRIMARY KEY clauses.

The statement cannot be processed.

Action: Correct the statement.

sqlcode: -637

sqlstate: 42614

SQL0638N

Table "<name>" cannot be created because no column definitions were specified.

Cause: The CREATE TABLE statement does not contain any column definitions.

The statement cannot be processed.

Action: Add one or more column definitions to the statement.

sqlcode: -638

sqlstate: 42601

SQL0644N

Invalid value specified for keyword "<keyword>" in statement "<statement-type>".

Cause: The keyword "<keyword>" is followed by an invalid value as allowed by the description of "<statement-type>". For numeric values, the value may be outside a defined range. For other types, the value is not in the defined set of valid values.

Action: Determine from reference documentation for the "<statement-type>" what are valid values and make the appropriate change.

sqlcode: -644

sqlstate: 42615

SQL0659N

Maximum size of a table object has been exceeded.

Cause: One or more of the objects which make up a table has reached its maximum size. The storage objects that make up a table are:

- Data: This is where basic column data is stored
- Index: This is where all indexes on the table are stored
- Long Data: This is where LONG VARCHAR and LONG VARGRAPHIC column data is stored
- Lob/Lob Allocation: This is where BLOB, CLOB, and DBCLOB column data and control information are stored

Once a storage object has grown to its maximum size, it cannot be extended further.

Action: To make existing space within the object available to store new data, the following actions may be taken:

- Reorganize the table
- Delete unnecessary rows from the table
- Drop an index from the table
- Update rows to reduce the amount of data (a reorg may be required after this action in order to reclaim unused storage)

sqlcode: -659

sqlstate: 54032

SQL0667N

The FOREIGN KEY "<name>" cannot be created because the table contains rows with foreign key values that cannot be found in the primary key of the parent.

Cause: The definition of the indicated foreign key failed because the table being altered contains at least one row where the foreign key does not match the primary key value in the parent table.

"<name>" is the constraint name in the FOREIGN KEY clause or, if a constraint name was not specified, the first column name specified in the clause.

The statement cannot be processed. The specified table is not altered.

Action: Remove the erroneous table rows and define the foreign key.

sqlcode: -667

sqlstate: 23520

SQL0668N

Operation not allowed when the underlying table is in Check Pending.

Cause: When a table is in a check pending state, there may be one or more rows that violate constraints defined on the data. This table (or object that depends on the table) cannot be used for the operation.

Action: Execute the SET CONSTRAINTS statement with the IMMEDIATE CHECKED option and ensure that the data satisfies all the constraints defined on the table.

sqlcode: -668

sqlstate: 57016

SQL0669N

A primary index cannot be dropped explicitly.

Cause: The DROP INDEX statement attempted to drop an index created to enforce the primary key of the table. The primary index cannot be dropped using the DROP INDEX statement.

The statement cannot be processed. The specified index is not dropped.

Action: If you do not want to keep the primary key, use the DROP PRIMARY KEY clause of the ALTER TABLE statement to remove the primary key and its primary index.

sqlcode: -669

sqlstate: 42917

SQL0670N

The row length of the table is too long (the database manager maximum row length is 4005 bytes).

Cause: The row length of a table in the database manager cannot exceed 4005 bytes. The length is calculated by adding the internal lengths of the columns. Details of internal column lengths can be found under CREATE TABLE in the *SQL Reference*.

One of two conditions can occur:

*   The row length for the table exceeds 4005 as defined in the CREATE TABLE or CREATE VIEW statement.
*   The row length would exceed 4005 bytes for a temporary table needed to process the SELECT statement.

The statement cannot be processed.

Action: In the case of CREATE TABLE or CREATE VIEW, to reduce the row length, eliminate one or more columns or reduce the lengths of one or more columns.

In the other case, eliminate one or more columns from the temporary table.

sqlcode: -670

sqlstate: 54010

SQL0673N

A primary index cannot be created because the table contains rows which are duplicates with respect to the values of the identified primary key columns.

Cause: The primary key definition failed because the table being altered already contains rows that duplicate the values of the PRIMARY KEY clause columns.

The statement cannot be processed. The specified table is not altered.

Action: Remove the erroneous rows from the table before attempting to define the primary key.

sqlcode: -673

sqlstate: 23515

SQL0680N

Too many columns are specified for a table (maximum is 255).

Cause: The maximum number of columns permitted for each table is 255.

DataJoiner users: other data sources may have a different maximum column limit. This limit has been exceeded.

The statement cannot be processed.

Action: Limit the table columns to 255. Create separate tables, as required, to hold additional information beyond 255 columns.

DataJoiner users: limit the number of table columns to the maximum supported by the data source. Create separate tables or views, as required, to hold additional information beyond the maximum number of columns supported by the data source.

sqlcode: -680

sqlstate: 54011

SQL0683N

The specification for column, distinct type or function "<data-item>" contains incompatible clauses.

Cause: There is an error in a data item specification in a CREATE or ALTER. Incompatible specifications are present, such as: "INTEGER and FOR BIT DATA". The location of the error is given by "<data-item>" as follows:

- for a CREATE or ALTER TABLE statement, "<data-item>" gives the name of the column containing the error.
- for a CREATE FUNCTION statement, "<data-item>" is a token that identifies the area of the problem in the statement. For example, "PARAMETER 3" or "RETURNS" or "CAST FROM".
- for a CREATE DISTINCT TYPE statement, "<data-item>" gives the name of the type being defined.

The statement cannot be processed.

Action: Remove the incompatibility and try the statement again.

sqlcode: -683

sqlstate: 42842

SQL0696N

The definition of trigger "<trigger-name>" includes an invalid use of correlation name or transition table name "<name>". Reason code="<reason-code>".

Cause: The trigger definition included an invalid use of "<name>". The value of "<reason-code>" indicates the specific problem as follows.

1 NEW correlation name and NEW_TABLE name are not allowed in a DELETE trigger.
2 OLD correlation name and OLD_TABLE name are not allowed in an INSERT trigger.
3 OLD_TABLE name and NEW_TABLE name are not allowed in a BEFORE trigger.

The statement cannot be processed.

Action: Remove the invalid correlation name or transition table name along with the preceding keyword.

sqlcode: -696

sqlstate: 42898

SQL0697N

OLD or NEW correlation names are not allowed in a trigger defined with the FOR EACH STATEMENT clause.

Cause: The trigger, as defined, includes a REFERENCING clause with OLD or NEW correlation names specified (or both) and the FOR EACH STATEMENT clause. These cannot be specified together.

The statement cannot be processed.

Action: Remove any OLD or NEW correlation names from the REFERENCING clause or replace FOR EACH STATEMENT with FOR EACH ROW.

sqlcode: -697

sqlstate: 42899

SQL0707N

The object "<name>" cannot be created because the first three characters are reserved for system objects.

Cause: The following is a list of reserved names:

- a table space name cannot start with 'SYS'

Action: Select a name that does not start with a reserved prefix.

sqlcode: -707

sqlstate: 42939

SQL0713N

The replacement value for "<special-register>" is invalid.

Cause: Either the value specified in the SET <special-register> statement is not a valid value of the indicated special register, or the value specified is NULL as a result of an indicator variable.

The statement cannot be executed.

Action: Correct the replacement value and/or any indicator variables. See the *SQL Reference* for an explanation of the valid values of each special register.

sqlcode: -713

sqlstate: 42815

SQL0723N SQL0751N

SQL0723N

An error occurred in a triggered SQL statement in trigger "<trigger-name>". Information returned for the error includes SQLCODE "<sqlcode>", SQLSTATE "<sqlstate>" and message tokens "<token-list>"

Cause: An SQL statement in the trigger "<trigger-name>" has failed during execution of the trigger. The sqlcode, sqlstate and message token list (each token is separated by the vertical bar character) are provided. The message tokens may be truncated. See the corresponding message for the "<sqlcode>" for further explanation of the error.

The trigger and the original SQL statement that caused the trigger to execute cannot be processed.

Action: Check the message associated with the SQLCODE of the SQL statement that failed. Follow the action suggested by that message.

sqlcode: -723

sqlstate: 09000

SQL0724N

The activation of trigger "<trigger-name>" would exceed the maximum level of cascading.

Cause: Cascading of triggers occurs when a triggered SQL statement in a trigger would result in another trigger being activated or when a referential constraint delete rule causes additional triggers to be activated. The depth of this cascading is limited to 16.

Note that recursive situations where a trigger includes a triggered SQL statement that directly or indirectly causes the same trigger to be activated is a form of cascading that is very likely to cause this error if there are no conditions to prevent cascading from exceeding the limit.

The "<trigger-name>" specified is one of the triggers that would have been activated at the seventeenth level of cascading.

Action: Start with the triggers that are activated by the UPDATE, INSERT or DELETE statement that received this error. If any of these triggers are recursive, ensure that there is some condition that prevents the trigger from being activated more than the limit allows. If this is not the cause of the problem, follow the chain of triggers that are activated to determine the chain that exceeds the cascading limit.

sqlcode: -724

sqlstate: 54038

SQL0750N

The source table cannot be renamed because it is referenced in a view, trigger, check constraint or referential constraint.

Cause: The source table in a RENAME statement cannot be renamed for one or more of the following reasons.

- The table is referenced in one or more existing views.
- The table is referenced in one or more existing triggers. This includes triggers on the table or references in triggered SQL statements.
- The table has check constraints defined.
- The table is involved in one or more referential constraints as a parent or dependent table.

The statement cannot be processed.

Action: Drop the views, triggers, check constraints or referential constraints on the table before issuing the RENAME statement. For views dependent on the table, query SYSCAT.VIEWDEP where the table matches BSCHEMA and BNAME columns. For triggers dependent on the table, query SYSCAT.TRIGDEP where the table matches BSCHEMA and BNAME columns. For check constraints on the table, query SYSCAT.CHECKS where the table matches TABSCHEMA and TABBNAME columns. For referential constraints dependent on the table, query SYSCAT.REFERENCES where the table matches TABSCHEMA and TABNAME columns or REFTABSCHEMA and REFTABNAME columns.

sqlcode: -750

sqlstate: 42986

SQL0751N

The trigger "<trigger-name>" is defined with an unsupported triggered SQL statement.

Cause: The trigger is defined with a triggered SQL statement that does match the following list.

- A BEFORE trigger can include the following triggered SQL statements:

  – a fullselect
  – a SET transition-variable statement (only when FOR EACH ROW is specified).

- An AFTER trigger can include the following triggered SQL statements:

  – in INSERT statement

– a searched UPDATE statement
– a searched DELETE statement
– a fullselect

In some cases the "<trigger-name>" is not available for use in the message.

Action: Check the triggered SQL statements in the trigger for any statement that does not match the above list and remove it.

sqlcode: -751

sqlstate: 42987

SQL0752N

Connecting to a database is not permitted within a logical unit of work when the CONNECT type 1 connection setting is in use.

Cause: An attempt was made to connect to either another database or the same database before issuing a COMMIT or ROLLBACK statement. The request cannot be processed within a CONNECT type 1 environment.

Action:

- Submit a COMMIT or ROLLBACK statement before requesting a connection to another database.

- If multiple databases need to be updated within a unit of work, change the connection settings to SYNCPOINT TWOPHASE and CONNECT 2 by re-precompiling or by issuing the SET CLIENT API from within the application.

sqlcode: -752

sqlstate: 0A001

SQL0801N

Division by zero was attempted.

Cause: The processing of a column function or arithmetic expression resulted in division by zero.

The statement cannot be processed. For the INSERT, UPDATE, or DELETE statements, no inserts or updates are performed.

Action: Examine the SQL statement to determine the cause of the problem. If the problem is data dependent, it is necessary to examine the data processed when the error occurred. Refer to the *SQL Reference* to see the valid ranges for the data types.

DataJoiner users: examine the SQL statement to determine the cause of the problem. If the problem is data dependent, examine the data being processed at the data sources when the error occurred.

sqlcode: -801

sqlstate: 22012

SQL0802N

Arithmetic overflow or other arithmetic exception occurred.

Cause: The processing of a column function or arithmetic expression has resulted in an arithmetic overflow.

The statement cannot be processed. For the INSERT, UPDATE, or DELETE statements, no inserts or updates are performed.

Action: Examine the SQL statement to determine the cause of the problem. If the problem is data dependent, it is necessary to examine the data processed when the error occurred. Refer to the *SQL Reference* to see the valid ranges for the data types.

DataJoiner users: examine the SQL statement to determine the cause of the problem. If the problem is data dependent, examine the data being processed at the data sources when the error occurred. Refer to the corresponding SQL Reference for the data source to determine the valid ranges for the data types.

sqlcode: -802

sqlstate: 22003

SQL0803N

One or more values in the INSERT or UPDATE statement are not valid because they would produce duplicate rows for a table with a unique index.

Cause: The INSERT or UPDATE object table is constrained by one or more UNIQUE indexes to have unique values in certain columns or groups of columns. Completing the requested insert or update results in duplicates of the column values.

Alternatively, if a view is the object of the INSERT or UPDATE statement, it is the table on which the view is defined that is constrained.

The statement cannot be processed. The table remains unchanged.

Action: Examine the definitions for all UNIQUE indexes defined on the object table to determine the uniqueness constraints those indexes impose.

For an UPDATE statement, ensure that the specified operation is not itself inconsistent with those uniqueness

SQL0804N   SQL0808N

constraints. If this does not show the error, examine the object table content to determine the cause of the problem.

For an INSERT statement, examine the object table content to determine which of the values in the specified value list violates the uniqueness constraint. Alternatively, if the INSERT statement contains a subquery, the object table contents addressed by that subquery must be matched against the object table contents to determine the cause of the problem.

DataJoiner users: isolate the problem to the data source failing the request (refer to the *Problem Determination Guide* to determine which data source is failing to process the SQL statement) and examine the index definitions and data for the conditions listed previously.

sqlcode: -803

sqlstate: 23505

SQL0804N

The application program input parameters for the SQL statement are not valid.

Cause: An error occurred while processing the SQLDA(s) or host variable(s) in the SQL statement.

The call parameter list, which is created by the precompiler, may be not correct if the application programmer has modified the output of the precompiler, used a variable name beginning with *SQL* in the application program, or over wrote the call parameter list in another way.

The SQLDA, which is created by the application program, may be invalid for one of the following reasons:

- the length of the input host variable (as provided in the length field of a VARCHAR, VARGRAPHIC, or LOB data type) is greater than the declared length, or is uninitialized
- the length of the data accessed via the SQLDATA pointer is greater than SQLLEN
- SQLN is less than SQLD
- the SQLDA contains a LOB type host variable but the SQLDOUBLED flag is not set
- the SQLLEN is not correct for the specified data type
- the SQLTYPE is invalid

The statement cannot be processed.

Action: Examine the application program for any errors noted. Note that the programmer should not attempt to modify the precompiler output.

sqlcode: -804

sqlstate: 07002

SQL0805N

Package "<package-name>" was not found.

Cause: Possible causes for this message (SQLCODE) are:

- The specified package or program is not defined in the database.
- The program was not bound or it was dropped.
- The application being run is not bound to the database.

The statement cannot be processed.

Action: Specify the correct package name or bind the program. If the application being run is not bound to the database, contact the database administrator to do the necessary binding.

DataJoiner users: ensure that the packages required for DataJoiner are bound at the applicable data sources. Refer to the *IBM DataJoiner Planning, Installation, and Configuration Guide* for more information on binding packages DataJoiner uses to the data sources.

sqlcode: -805

sqlstate: 51002

SQL0808N

The CONNECT statement semantics are not consistent with those of other existing connections.

Cause: The CONNECT statement originates from a source file which was precompiled having different connection options (SQLRULES, CONNECT type, SYNCPOINT, or RELEASE type) than that of a source file for which a connection exists.

Action: Ensure that all source files are precompiled using the same CONNECT options, or if this is not possible, call the SET CLIENT api to set the desired options for the application process before issuing the first CONNECT statement.

sqlcode: -808

sqlstate: 08001

Cause: One of the following caused the error:

- Execution of an embedded SELECT INTO or VALUES INTO statement resulted in a result table of more than one row.

- Execution of a scalar fullselect resulted in a result table of more than one row.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

The statement cannot be processed.

Action: Ensure that the statement contains the proper condition specifications. If it does, there may be a data problem that is causing more than one row to be returned when only one is expected.

DataJoiner users: isolate the problem to the data source failing the request (refer to the *Problem Determination Guide* to determine which data source is failing to process the SQL statement) and examine the selection criteria and data for that object.

sqlcode: -811

sqlstate: 21000

SQL0818N

A timestamp conflict occurred.

Cause: The timestamp generated by the precompiler at precompile time is not the same as the timestamp stored with the package at bind time.

This problem can be caused by the following:

- Precompile, compile, and link without doing an application bind.
- Precompile and bind without doing the program compile and link.
- Bind the application using a bind file that resulted from a different precompile of the program than the precompile that produced the object module linked into the application module.
- Bind an application with the same name as an existing plan and execute the existing (old) application.

DataJoiner users: in addition to the causes listed previously, the problem can also be due to not binding the DataJoiner packages at all applicable data sources.

The statement cannot be processed.

Action: Bind the application again, using the bind file for the program that matches the object module. Or, execute the program that corresponds to the package stored in the database.

If installing the sample database, record the number and text of this message and then contact your technical service representative.

DataJoiner users: in addition to the previously listed actions, ensure that the packages required for DataJoiner are bound at the applicable data sources. Refer to the *IBM DataJoiner Planning, Installation, and Configuration Guide* for more information on binding packages DataJoiner uses to the data sources.

sqlcode: -818

sqlstate: 51003

SQL0822N

The SQLDA contains a data address or indicator variable address which is not valid.

Cause: The application program placed an address that is not valid in the SQLDA.

The statement cannot be processed.

Action: Correct the application program so that valid addresses are placed in SQLDA.

sqlcode: -822

sqlstate: 51004

SQL0840N

Too many items were returned in a SELECT list (maximum is 255).

Cause: The number of items returned in the SELECT list exceeds the allowable maximum. The maximum for the database manager is 255.

The statement cannot be processed.

Action: Determine whether all the information is actually needed. (Note that the number of items returned by the SELECT list * in the SQL statement SELECT * from A, B, C is the sum of the number of columns in all three tables.) If possible, rewrite the SQL statement so only the necessary items of information are returned. If all the information is necessary, break the SQL statement into two or more statements.

sqlcode: -840

sqlstate: 54004

SQL0842N SQL0863W

SQL0842N

A connection to server "<server-name>" already exists.

Cause: SQLRULES(STD) is in effect and a CONNECT statement identifies an existing SQL connection.

Action: The correction depends on the error:

- If the server name is not the intended name, correct it.
- If SQLRULES(STD) is in effect and the CONNECT statement identifies an existing SQL connection, replace the CONNECT with SET CONNECTION or change the option to SQLRULES(DB2).

Correct the error in the application and try again.

sqlcode: -842

sqlstate: 08002

SQL0843N

The SET CONNECTION, DISCONNECT or RELEASE statement must specify an existing connection.

Cause: One of the following rules was violated:

- A SET CONNECTION statement must identify an existing SQL connection of the application process.
- A RELEASE statement must identify an existing SQL connection of the application process.
- A DISCONNECT statement must identify an existing SQL connection of the application process.

Action: The correction depends on the error:

- If the server name is not the intended name, correct it.
- Ensure that a connection to the server has been established and is in a current or dormant state before issuing SET CONNECTION, DISCONNECT or RELEASE for that connection.

Correct the error in the application and try again.

sqlcode: -843

sqlstate: 08003

SQL0859N

Access to the Transaction Manager Database failed with SQLCODE "<SQLCODE>".

Cause: The application was precompiled with SYNCPOINT(TWOPHASE) and requires a Transaction Manager Database to coordinate the two phase commit. Reasons why the Transaction Manager Database is not available might include the following:

- It was not created.
- The *tm_database* field of the database manager configuration file has not been updated and activated with the name of the database.
- The database exists, but communication to the database failed.

Action: Possible actions:

- Refer to the SQLCODE that was returned with this message, and follow the appropriate action for that SQLCODE.
- Verify that the *tm_database* exists; if not, create a new database or select a database that currently exists for use as the TM database. It is recommended to create a separate database if there are no serious constraints on disk storage.
- Update the database manager configuration for the TM database using the field *tm_database* if that has not already been done.
- Verify that a connection to the *tm_database* can be made. For example, attempt the connection using the Command Line Processor.
- Ensure that the *tm_database* chosen is not a database accessed through DDCS.

sqlcode: -859

sqlstate: 08502

SQL0863W

A successful connection was made, but only single byte characters should be used.

Cause: The server database and client application are using codepages for different language types and any characters outside the 7-bit ASCII range cannot be guaranteed (only the characters in the 7-bit ASCII range exist in all codepages). For example, the connection might be between a Japanese and a Latin-1 codepage, but none of the Japanese characters will be available in the Latin-1 codepage, so all of these characters should be avoided (English characters are fine though).

DataJoiner users: possible causes are:

- The DataJoiner database supports single-byte and double-byte characters but the database client system supports only single-byte characters.
- The data source supports single-byte and double-byte characters but the DataJoiner system supports only single-byte characters.

Action: Do not submit SQL statements or commands that use characters which are not common between the application and database codepages.

DataJoiner users: do not submit SQL statements or commands that use characters which are not common between the client system, the DataJoiner system, and the data source.

sqlcode: +863

sqlstate: 01539

SQL0865N

Invalid tm_database value.

Cause: The database chosen as the *tm_database* in the database manager configuration is not valid. The database must be at level DB2 V2.1 or later, and cannot be a database accessed through DRDA protocol (ie. through DDCS).

The statement cannot be executed.

Action:

1. Update the database manager configuration to specify a valid database for the *tm_database parameter*.

2. Issue db2stop and db2start to ensure that the change takes effect.

sqlcode: -865

sqlstate: 08001

SQL0866N

Connection redirection failed. Reason code: "<reason-code>"

Cause: Directory cataloging for a database caused a connection to be redirected in a manner which is not supported.

Possible reason codes are:

01   A database connection involved more than one redirection from a server to another server; only one connection redirection is supported.

02   A connection was attempted which involved both a current version of DB2 client or server and a version 1 client or server. This attempt failed because redirection is not supported with version 1 clients or servers.

Action: By the reason code, the actions are:

01   Recatalog the database so that there is no more than one server redirecting the connection to another server in the path of the connection.

02   Recatalog the database so that there is no intermediary server redirecting the connection.

sqlcode: -866

sqlstate: 08001

SQL0868N

A CONNECT using a USER/USING clause was attempted to a server for which a connection already exists.

Cause: A current or dormant connection to a server exists, and an attempt to CONNECT to this server using the USER/USING clause is invalid.

Action: Possible actions:

- Use the SET CONNECTION statement to make the connection to the DORMANT connection current.

- Use the CONNECT statement without USER/USING if SQLRULES(DB2) are being used by the application.

- Complete the existing unit of work, disconnect, and then reconnect using USER/USING.

sqlcode: -868

sqlstate: 51022

SQL0900N

The application state is in error. A database connection does not exist.

Cause: A connection to a database does not exist. This may be due to one of the following reasons:

- A serious error in the application state has caused the database connection to be lost.

- The application may have disconnected from a database and not established a new current connection before executing the next SQL statement.

Action: Reestablish a current connection be either switching to an existing dormant connection (using

SQL0901N   SQL0903N

CONNECT TO or SET CONNECTION) or by establishing a new connection (using CONNECT).

sqlcode: -900

sqlstate: 08003

SQL0901N

The SQL statement failed because of a non-severe system error. Subsequent SQL statements can be processed. (Reason "<reason>".)

Cause: An error has occurred which is due to a system error. Processing of the SQL statement has ended for reason "<reason>" (this is in English only and is useful only to IBM support personnel).

Action: Record the message number (SQLCODE) and reason "<reason>".

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Then contact your technical service representative with the following information:

- Problem description
- SQLCODE
- reason "<reason>"
- SQLCA contents if possible
- Trace file if possible.

DataJoiner users: isolate the problem to the data source failing the request (refer to the *Problem Determination Guide* to determine which data source is failing to process the SQL statement) and take the necessary diagnostic steps for that data source. The problem determination procedures for data sources vary, so refer to the applicable data source manuals.

sqlcode: -901

sqlstate: 58004

SQL0902C

A system error (reason code = "<reason-code>") occurred. Subsequent SQL statements cannot be processed.

Cause: A system error occurred.

Action: Record the message number (SQLCODE) and reason code in the message.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the

Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Then contact your technical service representative with the following information:

- Problem description
- SQLCODE and embedded reason code
- SQLCA contents if possible
- Trace file if possible.

DataJoiner users: isolate the problem to the data source failing the request (refer to the *Problem Determination Guide* to determine which data source is failing to process the SQL statement) and take the necessary diagnostic steps for that data source. The problem determination procedures for data sources vary, so refer to the applicable data source manuals.

sqlcode: -902

sqlstate: 58005

SQL0903N

COMMIT statement failed, transaction rolled back. Reason code: "<reason-code>"

Cause: One or more of the servers participating in the current unit of work was unable to prepare the database to be committed. The COMMIT statement has failed and the transaction has been rolled back.

Possible reason codes are:

01   A connection to one of the databases participating in the unit of work was lost.

   DataJoiner users: if the database you connected to is a DataJoiner database where nicknames are used, a connection to one of the data sources required for a nickname within the database was lost.

02   One of the databases participating in the unit of work was accessed, but unable to prepare to commit.

   DataJoiner users: if the database you connected to is a DataJoiner database where nicknames are used, one of the data sources required for a nickname is unable to prepare to commit.

Action: If a connection to a database was lost, reestablish the connection. If the failure was not connection related, reference the error diagnostic logs on the remote system to determine the nature of the failure and what action might be required. Rerun the application.

sqlcode: -903

sqlstate: 40504

SQL0910N

The SQL statement cannot access an object on which a modification is pending.

Cause: The application program attempted to access an object within the same unit of work in which either:

-

SQL0913N   SQL0925N

SQL0913N

Unsuccessful execution caused by deadlock or timeout. Reason code "<reason-code>".

Cause: The request issued was involved in an unresolved contention for use of an object and the execution failed.

The reason codes are as follows:

2   transaction failed due to deadlock.
68   transaction failed due to lock timeout.
80   statement failed due to timeout.

Action:

- For reason code 80, you can retry the failed statement without terminating the application. If the application accesses multiple remote databases, it may be better to rollback the transaction to avoid the possibility of a global deadlock.

- For other reason codes, issue a request to rollback the transaction. The transaction cannot be commited due to the failure of the current transaction branch.

- To help avoid deadlock or lock timeout, issue frequent COMMIT operations, if possible, within long-running applications or applications requiring data with high concurrent access.

sqlcode: -913

sqlstate: 57033

SQL0918N

Application must execute a rollback.

Cause: The unit of work has already been rolled back in the database but other resource managers involved in this unit of work might not. To ensure integrity of this application, all SQL requests will be rejected until the application issues a rollback.

Action: All SQL requests will be rejected until the application issues a rollback. For example, in a CICS environment, this would be a CICS SYNCPOINT ROLLBACK command.

sqlcode: -918

sqlstate: 51021

SQL0920N

Data on a database client system cannot be accessed from other database client systems.

Cause: The workstation has been configured as a client or a server with local clients. Any databases created on this system cannot be shared by other workstations.

The function cannot be processed.

Action: Request data only from server workstations.

sqlcode: -920

sqlstate: 57019

SQL0925N

SQL COMMIT invalid for application execution environment.

Cause: COMMIT is disallowed in the following cases:

- In a Distributed Transaction Processing environment such as CICS, a static SQL COMMIT statement was attempted, but a commit statement specific to the environment is required. For example, in a CICS environment this would be the CICS SYNCPOINT command.

- A DB2 application precompiled or set to use CONNECT 2 in a non-TP Monitor environment has issued a dynamic SQL COMMIT statement, whereas only static SQL COMMITs are allowed.

- When issued from a stored procedure, SQL COMMIT is also not allowed if the calling program is executing in a distributed unit of work or Distributed Transaction Processing environment.

Action: Correct the problem by doing one of the following:

- Remove the statement issuing the COMMIT and replace it with a statement which does the valid equivalent for the environment
- In the case of a connect type 2 in a non-TP Monitor environment, use only static COMMIT.
- In the case of a stored procedure, remove the COMMIT

sqlcode: -925

sqlstate: 2D521

SQL0926N

SQL ROLLBACK invalid for application execution environment.

Cause: ROLLBACK is disallowed in the following cases:

1. In a Distributed Transaction Processing environment such as CICS, a static SQL ROLLBACK statement was attempted, but a rollback statement specific to the environment is required. For example, in a CICS environment this would be the CICS SYNCPOINT ROLLBACK command.

2. A DB2 application precompiled or set to use CONNECT 2 has issued a dynamic SQL ROLLBACK statement, whereas only static SQL ROLLBACKs are allowed.

3. When issued from a stored procedure, SQL ROLLBACK is also restricted if the calling program is executing in a distributed unit of work (CONNECT type 2) or Distributed Transaction Processing environment.

Action:

1. Remove the statement issuing the ROLLBACK and replace it with a statement which does the valid equivalent for the environment.

2. In the case of a connect type 2, use only static COMMIT.

3. In the case of a stored procedure, remove it entirely.

sqlcode: -926

sqlstate: 2D521

SQL0930N

There is not enough storage available to process the statement.

Cause: A request was made to the database that required another memory page but no more pages are available to the database manager.

The statement cannot be processed.

Action: Possible solutions include:

- Verify that your system has sufficient real and virtual memory.
- Remove background processes.
- If the error occurs during DUOW resynchronization, increase the *maxagents* and reduce the

resync_interval database manager configuration parameter values.

sqlcode: -930

sqlstate: 57011

SQL0931C

Operating system file table overflow occurred. Subsequent SQL statements cannot be processed.

Cause: An Operating system limit has been reached. The application program is not permitted to issue additional SQL statements. The database is marked as needing recovery and all applications using the database are prevented from accessing the database.

Action: Terminate all the applications using the database. Restart the database.

To help prevent the problem from recurring:

- Change the MAXFILOP database configuration parameter to a smaller value (this will reduce DB2's use of the operating system file table), and/or

- Terminate other applications that are using files, if appropriate, and/or,

- Refer to the operating system documentation for increasing the operating system file table limit. In most UNIX environments this can be accomplished by updating the kernel configuration with a larger value. (On AIX, this may only be possible by increasing the amount of memory in your machine).

sqlcode: -931

sqlstate: 58005

SQL0950N

The table or index cannot be dropped because it is currently in use.

Cause: A DROP TABLE or DROP INDEX statement cannot be issued when an open cursor is currently using the table or index.

The statement cannot be processed. The table or index is not dropped.

Action: Close any cursors required and resubmit the statement.

sqlcode: -950

sqlstate: 55006

SQL0951N SQL0956C

SQL0951N

The table cannot be referenced in a SET CONSTRAINTS statement because it is currently in use.

Cause: A SET CONSTRAINTS statement cannot be issued for a table that has an open cursor.

The statement cannot be processed.

Action: Ensure that the SET CONSTRAINTS statement does not reference tables that have open cursors.

sqlcode: -951

sqlstate: 55007

SQL0952N

Processing was cancelled due to an interrupt.

Cause: The user may have pressed the interrupt key sequence.

The statement processing is terminated. Some changes may have been applied to the database, but not committed, before termination occurred.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Continue the application.

If installing the sample database, drop it and install the sample database again.

sqlcode: -952

sqlstate: 57014

SQL0953C

Not enough storage is available in the agent heap to process the statement.

Cause: All available memory in the server agent has been used.

The command or statement cannot be processed.

Action: Terminate the application on receipt of this message. Increase the size of the agent heap parameter (*agentheap*) in the corresponding database configuration file.

sqlcode: -953

sqlstate: 57011

SQL0954C

Not enough storage is available in the application heap to process the statement.

Cause: All available memory for the application has been used.

The statement cannot be processed.

Action: Terminate the application on receipt of this message. Increase the database configuration parameter (*applheapsz*) to allow a larger application heap.

sqlcode: -954

sqlstate: 57011

SQL0955C

Sort memory cannot be allocated to process the statement.

Cause: Insufficient virtual memory is available to the database agent for sort processing.

The statement cannot be processed but other SQL statements may be processed.

Action: Terminate the application on receipt of this message. Decrease the size of the sort heap parameter (*sortheap*) in the corresponding database configuration file.

sqlcode: -955

sqlstate: 57011

SQL0956C

Not enough storage is available in the database heap to process the statement.

Cause: All available memory for the database has been used.

The statement cannot be processed.

Action: Terminate the application on receipt of this message. Increase the database configuration parameter (*dbheap*) to allow a larger database heap. If the number of I/O servers is near the high limit, reducing this number may also help.

sqlcode: -956

sqlstate: 57011

SQL0957C

Not enough storage is available in the communication heap of the database client, DDCS, or DataJoiner machine to process the statement.

Cause: All available memory in the communication heap of the database client, DDCS, or DataJoiner machine has been used.

The command or statement cannot be processed.

Action: Terminate the application on receipt of this message (SQLCODE). Increase the size of the communication heap (*comheapsz*) parameter in the client, DDCS, or DataJoiner workstation database manager configuration file.

NOTE: This message is applicable only for releases of DB2 prior to Version 2.

sqlcode: -957

sqlstate: 57011

SQL0958C

The maximum number of open files has been reached.

Cause: The maximum number of file handles available to the database has been reached.

The statement cannot be processed.

Action: Increase any parameter that affects the maximum number of open files allowed at the location of the database. This includes increasing the configuration parameter (*maxfilop*) to allow more file handles for the instance and terminating other sessions to reduce the number of file handles being used.

sqlcode: -958

sqlstate: 57009

SQL0959C

Not enough storage is available in the communication heap of the server to process the statement.

Cause: All available memory in the server communication heap has been used.

The command or statement cannot be processed.

Action: Terminate the application on receipt of this

message. Increase the size of the communication heap (*comheapsz*) parameter in the server workstation database manager configuration file.

NOTE: This message is applicable only for releases of DB2 prior to Version 2.

sqlcode: -959

sqlstate: 57011

SQL0960C

The maximum number of files has been reached in the database.

Cause: The maximum number of database files has been reached.

The statement cannot be processed.

Action: Terminate the application on receipt of this message. Have all active applications disconnect from the database and then have them reconnect to the database. If the error continues, drop either tables, indexes, or both from the database, or split the database.

If installing the sample database, drop it and install the sample database again.

sqlcode: -960

sqlstate: 57011

SQL0961C

Not enough storage is available in the remote services heap of the database client, DDCS, or DataJoiner machine to process the statement.

Cause: All available memory in the remote services heap of the database client, DDCS, or DataJoiner machine has been used.

The command or statement cannot be processed.

Action: Terminate the application on receipt of this message. Increase the size of the remote services heap (*rsheapsz*) parameter in the client, DDCS, or DataJoiner workstation database manager configuration database manager configuration file.

NOTE: This message is applicable only for releases of DB2 prior to Version 2.

sqlcode: -961

sqlstate: 57011

SQL0962C • SQL0966N

SQL0962C

Not enough storage is available in the remote services heap of the server to process the statement.

Cause: All available memory in the server remote services heap has been used.

The command or statement cannot be processed.

Action: Terminate the application on receipt of this message (SQLCODE). Increase the size of the remote services heap (*rsheapsz*) parameter in the server workstation database manager configuration file.

NOTE: This message is applicable only for releases of DB2 prior to Version 2.

sqlcode: -962

sqlstate: 57011

SQL0964C

The transaction log for the database is full.

Cause: All space in the transaction log is being used.

If a circular log with secondary log files is being used, an attempt has been made to allocate and use them. When the file system has no more space, secondary logs cannot be used.

If an archive log is used, then the file system has not provided space to contain a new log file.

The statement cannot be processed.

Action: Execute a COMMIT or ROLLBACK on receipt of this message (SQLCODE) or retry the operation.

If the database is being updated by concurrent applications, retry the operation. Log space may be freed up when another application finishes a transaction.

Issue more frequent commit operations. If your transactions are not committed, log space may be freed up when the transactions are committed. When designing an application, consider when to commit the update transactions to prevent a log full condition.

If deadlocks are occurring, check for them more frequently. This can be done by decreasing the database configuration parameter DLCHKTIME. This will cause deadlocks to be detected and resolved sooner (by ROLLBACK) which will then free log space.

If the condition occurs often, increase the database configuration parameter to allow a larger log file. A

larger log file requires more space but reduces the need for applications to retry the operation.

If installing the sample database, drop it and install the sample database again.

sqlcode: -964

sqlstate: 57011

SQL0965W

There is no message text corresponding to SQL warning "<error>" in the message file on this workstation. The warning was returned from module "<name>" with original tokens "<token list>".

Cause: A warning code (SQLCODE) was returned to the application. The warning code does not correspond to a message in the database manager message file at this workstation. The warning code either could not be mapped from another product to a database manager warning or could not be found in the database manager message file.

Action: Refer to the location where the database resides to find a corresponding message for the specified SQLCODE. Then examine the application to determine the cause of the problem. If the problem may be data dependent, it is necessary to examine the data being processed when the error occurred.

SQL0966N

The error mapping file "<name>" specified in the Database Connection Services directory cannot be found or cannot be opened.

Cause: One of the following conditions is true:

- The error mapping file does not exist.
- The error mapping file is currently opened by another application.
- The error mapping file is not located in the expected path.
- The error mapping file is damaged.

The error mapping data could not be retrieved.

Action: Release the file from the application in which it is open or reinstall or restore the original file.

sqlcode: -966

sqlstate: 57013

SQL0967N

The format of the error mapping file "<name>" specified in the Database Connection Services directory is not valid.

Cause: An error occurred while the program was reading the error mapping file.

The error mapping data could not be retrieved.

Action: Correct all syntactic errors in the error mapping file.

sqlcode: -967

sqlstate: 55031

SQL0968C

The file system is full.

Cause: One of the file systems containing the database is full. This file system may contain the database directory, the database log files, or a tablespace container.

The statement cannot be processed.

Action: Refer to the diagnostic log to determine which file system is full. Free system space by erasing unwanted files. Do not erase database files. If additional space is required, it may be necessary to drop tables and indexes identified as not required.

On unix-based systems, this disk full condition may be due to exceeding the maximum file size allowed for the current userid. Use the chuser command to update fsize. A reboot may be necessary.

sqlcode: -968

sqlstate: 57011

SQL0969N

There is no message text corresponding to SQL error "<error>" in the message file on this workstation. The error was returned from module "<name>" with original tokens "<token list>".

Cause: An error code (SQLCODE) was returned to the application. The error code does not correspond to a message in the database manager message file at this workstation. The error code either could not be mapped

from another product to a database manager error or could not be found in the database manager message file.

Action: Refer to the location where the database resides to find a corresponding message for the specified SQLCODE. Then examine the application to determine the cause of the problem. If the problem may be data dependent, it is necessary to examine the data being processed when the error occurred.

DataJoiner users: isolate the problem to the data source failing the request (refer to the *Problem Determination Guide* to determine which data source is failing to process the SQL statement). Find the "<error>" in the applicable manual at the data source. Examine the application to determine the cause of the problem. If the probem is data dependent, it may be necessary to examine the data being processed at the data source at the time the error occurred.

SQL0970N

The system attempted to write to a read-only file.

Cause: A file used by the database is marked read-only but requires write access.

The statement cannot be processed.

Action: Terminate the application on receipt of this message (SQLCODE). Ensure that all database files are allowed both read and write access.

sqlcode: -970

sqlstate: 55009

SQL0972N

The database drive does not contain the correct diskette.

Cause: The diskette in the drive is not the database diskette.

The statement cannot be processed.

Action: Place the required diskette into the drive. Do not remove a diskette if the application starts using a database that resides in it.

sqlcode: -972

sqlstate: 57019

SQL0973N   SQL0978N

SQL0973N

Not enough storage is available in the "<heap-name>" heap to process the statement.

Cause: All available memory for this heap has been used. The statement cannot be processed.

Action: Terminate the application on receipt of this message (SQLCODE). Increase the configuration parameter for "<heap-name>" to increase the heap size.

sqlcode: -973

sqlstate: 57011

SQL0974N

The drive containing the database is locked.

Cause: The system reported that the drive containing the database is locked.

The statement cannot be processed.

Action: Ensure no other processing (for example, CHKDSK) that can lock the drive is occurring on the system. Retry the operation.

If installing the sample database, drop it and install the sample database again.

sqlcode: -974

sqlstate: 57020

SQL0975N

A new transaction could not be started because database or instance "<name>" is quiesced by user "<username>". Quiesce type: "<type>".

Cause: Another user has quiesced the instance or database that you are attempting to use, and no new transactions are allowed until the instance or database is no longer in the quiesced state.

Quiesce type "<type>" refers to the instance or database already quiesced and is a '1' for an instance and a '2' for a database.

Action: Contact the user who currently has the instance or database quiesced to determine when DB2 will no longer be quiesced, and retry the request at that time.

sqlcode: -975

sqlstate: 57046

SQL0976N

The diskette drive door is open.

Cause: The drive door for the database is open.

The statement cannot be processed.

Action: Close the drive door and retry the operation.

sqlcode: -976

sqlstate: 57021

SQL0977N

COMMIT state unknown.

Cause: The *tm_database* has become unavailable during COMMIT processing, and this has caused the result of the COMMIT to be unknown. Resynchronization of the databases will occur when the *tm_database* becomes available. Note that the transaction may be rolled back during resynchronization. Further execution of SQL statements can safely occur, however locks might be held until the resynchronization process is complete.

Action: Verify that the connection to the *tm_database* can be made, for example by using CLP. If it cannot be made, follow the necessary actions for the SQLCODE returned to ensure that a connection can be established.

sqlcode: -977

sqlstate: 40003

SQL0978N

The diskette is write-protected.

Cause: A write operation was attempted to the database but the database diskette is write protected.

The statement cannot be processed.

Action: Verify that you are using the correct diskette. Remove protection from the diskette, if necessary.

sqlcode: -978

sqlstate: 55009

SQL0979N

COMMIT has failed against "<num>" databases for an application process running with SYNCPOINT of NONE. The failures include the following database alias and SQLSTATE pairs (a maximum of four can be returned): "<alias/SQLSTATE1>", "<alias/SQLSTATE2>", "<alias/SQLSTATE3>", "<alias/SQLSTATE4>".

Cause: An application was connected to multiple databases, and a COMMIT was issued which failed for one or more of these connections.

DataJoiner users: if one of the failed connections is a DataJoiner database where nicknames are used, then a commit against one of the data sources required for a nickname has failed.

Action: Depending upon the nature of the application and the data being updated, the user might wish to discontinue the processing being done, log the failure, and issue the appropriate SQL to ensure that the changes intended by the application are consistently reflected across all databases involved.

If a full list of databases affected by COMMIT errors could not be returned please refer to the diagnostic log for a full list.

sqlcode: -979

sqlstate: 40003

SQL0980C

A disk error occurred. Subsequent SQL statements cannot be processed.

Cause: A disk error occurred that prevented successful execution of the current and subsequent SQL statements. The application program is not permitted to issue additional SQL statements. For example, a recovery routine associated with the application program cannot issue additional SQL statements. The database is marked as needing recovery and all applications using the database are prevented from accessing the database.

The statement cannot be processed.

Action: Record all error information from the SQLCA, if possible. Terminate all applications using the database. Determine if the error is a hardware error and take the appropriate action as specified in the *Problem Determination Guide* for hardware problems. Restart the database. If recovery is not possible, restore the database from a backup copy.

If installing the sample database, drop it and install the sample database again.

sqlcode: -980

sqlstate: 58005

SQL0982N

A disk error occurred. However, subsequent SQL statements can be processed.

Cause: A disk error occurred during processing of a temporary system file that prevented successful execution of the current SQL statement. However, subsequent SQL statements can be processed.

The statement cannot be processed.

Action: Terminate the application on receipt of this message (SQLCODE).

sqlcode: -982

sqlstate: 58004

SQL0983N

The transaction log does not belong to the current database.

Cause: The signature stored in the log file does not match the database dependent signature. This error usually occurs when the user specified that the log file be stored in a directory different from where the database is stored. File redirection can be involved.

The statement cannot be processed.

Action: Resubmit the command with the proper access to the log file.

sqlcode: -983

sqlstate: 57036

SQL0984C

COMMIT or ROLLBACK was not successful. Subsequent SQL statements cannot be processed.

Cause: A commit or rollback operation could not be processed successfully because of a system error. The application program is not permitted to issue additional SQL statements. For example, a recovery routine associated with the application program may not issue additional SQL statements. The database is marked as needing recovery and all applications using the database are prevented from accessing the database.

SQL0985C SQL0992C

The statement cannot be processed.

Action: Record the message number (SQLCODE) and all SQLCA error information, if possible. Terminate all applications using the database. Restart the database. If installing the sample database, drop it and install the sample database again.

If recovery is not possible, restore the database from a backup copy.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Contact your technical service representative with the following information:

Information required:

- Problem description
- SQLCODE
- SQLCA contents if possible
- Trace file if possible.

DataJoiner users: isolate the problem to the data source failing the request (refer to the *Problem Determination Guide* to determine which data source is failing to process the SQL statement) and take the necessary diagnostic steps and database recovery procedures for that data source. The problem determination procedures and database recovery procedures for data sources vary, so refer to the applicable data source manuals.

sqlcode: -984

sqlstate: 58005

SQL0985C

A file error occurred while processing the database catalogs. The database is not usable.

Cause: The system is unable to recover from an I/O error on a catalog file.

The system cannot process any statements using the database.

Action: Restore the database from a backup copy.

If installing the sample database, drop it and install the sample database again.

sqlcode: -985

sqlstate: 58005

SQL0986N

A file error occurred while processing a user table. The table is not usable.

Cause: The data in the table is no longer valid.

The system cannot process any statements using the table.

Action: Restore the database from a backup version if the database is inconsistent.

If installing the sample database, drop it and install the sample database again.

sqlcode: -986

sqlstate: 58004

SQL0990C

An index error occurred. Reorganize the table.

Cause: An index has had considerable activity that used all the free space for indexes.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

The statement cannot be processed.

Action: Commit your work and retry the command. If the error continues, roll back your work. If errors still persist, reorganize the table, if possible.

DataJoiner users: isolate the problem to the data source failing the request (refer to the *Problem Determination Guide* to determine which data source is failing to process the SQL statement) and follow the index re-creation procedures for that data source.

SQL0992C

The release number of the precompiled program is not valid.

Cause: The release number of the precompiled program (package) is not compatible with the release number of the installed version of the database manager.

The precompiled program (package) cannot be used with the current version of the database manager. The command cannot be processed.

Action: Use only programs that are precompiled with a compatible release level of the database manager.

sqlcode: -986

sqlstate: 51008

SQL0993W

The new path to the log (newlogpath) in the database configuration file is not valid.

Cause: The path to the log file is not valid for one of the following reasons:

- The path does not exist.
- A file with the correct name was found in the specified path, but it is not a log file for this database.
- The database manager instance id does not have permission to access the path or a log file.

The requested change is not made.

Action: To change the path to the log file, submit a database configuration command with a valid value.

sqlcode: +993

sqlstate: 01562

SQL0994N

Invalid usage of the application savepoint.

Cause: The application savepoint function has been used inconsistently. The program attempted to perform one of the following:

- Request more than one active savepoint.
- Issue an end savepoint call without an active savepoint.
- Issue a rollback savepoint call without an active savepoint.

The function cannot be processed.

Action: Correct the savepoint usage in the program.

SQL0995W

The current path to the log file (logpath) is not valid. The log file path is reset to the default.

Cause: The path to the log file is not valid for one of the following reasons:

- The path does not exist.
- A file with the correct name was found in the specified path, but it is not a log file for this database.

- The database manager instance id does not have permission to access the path or log file.

For circular logging, the log file is created in the default log path. For archive logging, the next log file is created in the default log path. The requested change is not made.

Action: To change the path to the log file, submit a configuration command with a valid value.

sqlcode: +995

sqlstate: 01563

SQL0996N

Cannot free pages for an object in a table space.

Cause: There are corrupted internal database pages or internal logic errors in a table space. Details can be found in the system error log and/or the database manager error log.

Action: Discontinue use of the object or table space. Contact IBM service to inspect the object and the table space.

sqlcode: -996

sqlstate: 58035

SQL0997W

General informational message for transaction processing. Reason Code = "<XA-reason-code>".

Cause: The SQLCODE 997 is only passed between components of the database manager and will not be returned to an application. It is used to carry XA return codes for non-error situations. The possible reason codes are:

- XA_RDONLY (3) - the transaction branch was read-only and has been committed.

- 64 - TM database indicates transaction to be committed on DUOW resynchronization

- 65 - TM database indicates transaction to be rolled back on DUOW resynchronization

Action: No action required.

SQL0998N SQL0998N

SQL0998N

Error occurs during transaction or heuristic processing. Reason Code = "<reason-code>" Subcode = "<subcode>".

Cause: An error is detected when processing a distributed transaction. The transaction is:

- Running under a Distributed Transaction Processing environment (such as those from CICS or other transaction managers).
- Performing heuristic operations.
- Updating multiple nicknames within a DataJoiner database, where each updated nickname represents a different data source. In this case, one of the data sources has failed during transaction processing. The reason code returned in this case is the reason for the failure at the data source, not the DataJoiner database.

The possible reason codes (corresponding X/Open XA reason codes are shown in parenthesis) are:

- 01 - (XAER_ASYNC) asynchronous operation already outstanding.
- 02 - (XAER_RMERR) a resource manager error occurred in the transaction branch.
- 03 - (XAER_NOTA) the XID is not valid.
- 04 - (XAER_INVAL) invalid arguments were given. Possible subcodes are:
    - 01 - xa_info pointer is invalid, eg. the XAOpen string is null.
    - 02 - database name exceeds maximum length.
    - 03 - username exceeds maximum length.
    - 04 - password exceeds maximum length.
    - 05 - username specified but not a password.
    - 06 - password specified but not a username.
    - 07 - too many parameters in the xa_info string.
    - 08 - multiple xa_opens generate different RM ids for the same database name.
    - 09 - database name not specified.
- 05 - (XAER_PROTO) routine invoked in an improper context.
- 06 - (XAER_RMFAIL) resource manager unavailable.
- 07 - (XAER_DUPID) the XID already exists.
- 08 - (XAER_OUTSIDE) RM doing work outside global transaction.

- 09 - registration (ax_reg) with the transaction manager failed. Possible subcodes are:
    - 01 - Joining XID not found
    - 02 - dynamic library specified in the tp_mon_name configuration parameter could not be loaded.
- 10 - attempted to start a different transaction while suspended.
- 12 - unregistering (ax_unreg) with the transaction manager failed.
- 13 - ax interface failure: ax_reg() and ax_unreg() not found.
- 35 - Heuristic operations are invalid for non-XA database.
- 36 - The XID is not known by the database manager.
- 37 - The transaction has already been heuristically committed.
- 38 - The transaction has already been heuristically rolled back.
- 39 - The transaction is not an indoubt transaction.
- 40 - Only rollbacks are allowed for this transaction.
- 69 - Database log ID mismatch during DUOW resynchronization.
- 85 - As a result of heuristic processing, the transaction has partially committed and rolled back.

Action: For reason codes 1 through 8, an entry will be made in the system log because sometimes the SQLCA cannot be returned to the caller.

If the error is due to a failed data source associated with a DataJoiner nickname, the location of the failed data source will always appear in DataJoiner's system log.

For reason code 4, verify the content of the xa open string and make necessary corrections.

For reason code 9, subcode 02, ensure that the tp_mon_name configuration parameter contains the name of the dynamic library in the external product which has the ax_reg() function used for dynamic registration of transactions.

For reason code 35, you attempted to perform a heuristic operation against a database that only participates as a read-only resource manager in a global transaction. Examples are any DRDA databases such as DB2 on MVS. These type of non-XA databases would not have any XA indoubt transactions.

For reason codes 36, 37, and 38, you attempted to perform an invalid heuristic operation on an indoubt transaction. Either you specified the wrong XID, or a heuristic or resync operation had taken place since you recorded this XID. Perform a Heuristic Query request to get the current list of indoubt transactions to verify if you still need to perform your heuristic operation.

For reason code 39, the XID specified is for a transaction that has ended and is waiting for the two-phase commit process to begin. You can only perform heuristic operations on transactions that have entered the two-phase commit process and become indoubt transactions.

For reason code 40, an SQL statement was attempted under a transaction which has failed. An example of this is to attempt an SQL statement in a transaction thread that has successfully registered after a tightly coupled thread participating in the same transaction has abnormally terminated.

For reason code 69, either the transaction manager (TM) database or the resource manager (RM) database or both are different from the ones when the indoubt transaction was generated. In other words, the TM database or RM database names could be referencing different database instances. The log ID mismatch may be caused by the following reasons:

- The database directory for the TM database at the RM instance is incorrect.

- The configuration may have been changed after the indoubt transaction was generated.

- The database may have been dropped and recreated. In that case, you can only commit or rollback the indoubt transaction heuristically.

For reason code 85, you are updating multiple data sources, and some of the data sources have been heuristically rolled back or committed, resulting in the transaction having partially committed and rolled back. With this reason code data is now in an inconsistent state, and you must manually check every data source updated by the transaction to correct the data.

General information collection procedure:

If you cannot resolve the problem identified by the reason codes, record the message number (SQLCODE), reason code, and the optional subcode in the message or the SQLCA in the system log.

If the failure is from a DataJoiner database, you should also record the location of the failed data source found in DataJoiner's system log.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Then, contact your technical service representative with the following information:

- Problem description

- SQLCODE, embedded reason code, and subcode

- SQLCA contents (if possible)

- Trace file (if possible)

- Location of the failed data source if the failure is from DataJoiner

There might be additional information in the console or message logs of the transaction manager and the database manager.

sqlcode: -998

sqlstate: 58005

SQL1000N

"<alias>" is not a valid database alias name.

Cause: The alias specified in the command or api is not valid. The alias must contain 1 to 8 characters (bytes in MBCS countries) and all the characters must be from the database manager base character set.

The command cannot be processed.

Action: Resubmit the command with the correct alias.

SQL1001N

"<name>" is not a valid database name.

Cause: The syntax of the database name specified in the command is not valid. The database name must contain 1 to 8 characters and all the characters must be from the database manager base character set.

The command cannot be processed.

Action: Resubmit the command with the correct database name.

sqlcode: -1001

sqlstate: 2E000

SQL1002N SQL1007N

SQL1002N

"<drive>" is not a valid drive.

Cause: The drive specified in the command is not valid. The drive is a single character (A to Z) indicating the diskette drive or fixed-disk partition where the database or database directory is located.

The command cannot be processed.

Action: Resubmit the command with the correct drive.

SQL1003N

The password is not valid because the syntax is not correct or the password does not match the password of the specified database.

Cause: The length of the password is expected to be 18 characters or less. However, if the password is to be verified for an APPC conversation, it must be 8 characters or less.

Action: Unsure that the password is not longer than the limit allowed.

sqlcode: -1003

sqlstate: 28000

SQL1004C

There is not enough storage on the file system to process the command.

Cause: There is not enough storage on the specified file system to process the command.

The command cannot be processed.

Action: Choose a different file system or erase some non-database files from the specified file system to make space for the database manager functions.

SQL1005N

The database alias "<name>" already exists in either the local database directory or the system database directory.

Cause: The alias name specified has already been used. If no alias is specified in the catalog database command, the database name is used as the alias. When a database is created, the alias name is the same as the database name.

This error may occur on the catalog database command when the alias already exists in the system database directory.

On the create database command this error may occur in one of the following situations:

- The alias already exists in the system database directory and local database directory.

- The alias already exists in the system database directory but not in the local database directory.

- The alias already exists in the local database directory but not in the system database directory.

Action: For the catalog database command, uncatalog the alias from the system database directory and resubmit your original command or catalog the database with a different alias name.

For the create database command, do the following operations with respect to the above 3 situations:

- Drop the database using the alias name. Resubmit your original command.

- Uncatalog the alias. Resubmit your original command.

- Catalog the alias into the system database directory. Drop the database using the same alias. Resubmit your original command.

SQL1006N

The code page "<code page>" of the application does not match the code page "<code page>" of the database.

Cause: The application could not connect to the database because the active codepage is different from the one that was active when the database was created.

The command cannot be processed.

Action: Exit the current application program and return to the operating system. Change the code page for the process and restart the application program.

SQL1007N

Error in finding pages for an object in a table space.

Cause: There are corrupted internal database pages or internal logic error for a table space. Details can be found in the system error log and/or the database manager error log.

Action: Discontinue use of the object or table space. Contact IBM service to inspect the object and the table space.

sqlcode: -1007

sqlstate: 58034

SQL1008N

Invalid table space ID.

Cause: The table space ID specified does not exist. It is either larger than the current maximum table space ID, or the table space has been dropped.

Action: Discontinue use of the database. Save the diagnostic information in the error log and contact IBM service.

sqlcode: -1008

sqlstate: 58036

SQL1009N

The command is not valid.

Cause: The command is not supported when issued on a client-only workstation or against remote databases. An example of such a command is cataloging a local database.

The command cannot be processed.

Action: Possible solutions include:

- Issue the specified command from a non client-only workstation or from the workstation where the database resides.
- Verify that the database is cataloged correctly.
- Issue a different command.

SQL1010N

"<type>" is not a valid type parameter.

Cause: The type specified in the Database Environment command is not valid. It must be '0' for an indirect database or '1' for a remote database.

In addition, on the AIX platform, the type can be '3' for a database that has a DCE global name.

The command cannot be processed.

Action: Resubmit the command with a valid type.

SQL1011N

No path was specified in the CATALOG DATABASE command for an indirect entry.

Cause: A CATALOG DATABASE command was issued for an indirect entry but no path was specified. Indirect entries must specify the path where the database resides.

The command cannot be processed.

Action: Resubmit the command with the required path or change the type.

SQL1012N

No node name was specified in the CATALOG DATABASE command for a remote entry.

Cause: There was no *nodename* parameter specified in the CATALOG DATABASE command for a remote entry. Remote entries must specify the node name of the database.

The command cannot be processed.

Action: Resubmit the command with the *nodename* parameter or a different type.

SQL1013N

The database alias name or database name "<name>" could not be found.

Cause: The database name or alias specified in the command is not an existing database or the database could not be found in the (client or server) database directories.

Action: Ensure that the specified database name exists in the system database directory. If the database name does not exist in the system database directory, then the database either does not exist or the database name has not been cataloged.

If the database name appears in the system database directory and the entry type is INDIRECT, ensure that the database exists in the specified local database directory. If the entry type is REMOTE, then ensure that the database exists and is cataloged on the database directories of the server node.

DataJoiner users: in addition to the above, verify that the database names specified in SYSCAT.SERVERS are all valid. Correct any SYSCAT.SERVERS entry for which the database specified in that entry does not exist.

sqlcode: -1013

SQL1014W   SQL1020C

sqlstate: 42705

SQL1014W

There are no more entries in the directory, file, or list being scanned.

Cause: The scan of the directory, file, or list is finished.

Action: No action is required.

SQL1015N

The database must be restarted because the previous session did not conclude normally.

Cause: The database must be restarted because of an abnormal termination of the previous session (for example, a power failure).

The command cannot be processed.

Action: On receipt of this message (SQLCODE), an application can post a message warning the user that it takes time to recover the database. To restart the database, issue the RESTART DATABASE command.

sqlcode: -1015

sqlstate: 55025

SQL1016N

The local_lu alias "<name>" specified in the CATALOG NODE command is not valid.

Cause: The local logical unit (*local_lu*) alias specified in the CATALOG NODE command is not permitted. The local logical unit alias is the local SNA logical unit alias and must be 1 to 8 characters and cannot contain blank characters.

The command cannot be processed.

Action: Verify that the alias is a permitted logical unit name. Verify the characters used in the name. Resubmit the command with a valid logical unit name.

SQL1017N

The mode parameter "<mode>" specified in the CATALOG NODE command is not valid.

Cause: The *mode* specified in the CATALOG NODE command is not permitted.

The *mode* parameter identifies the communications profile the Communications Manager uses to set up a session. The mode must be 1 to 8 characters. Valid characters are uppercase or lowercase A through Z, 0 through 9, #, @, and $. The first character must be alphabetic. The system changes lowercase characters to uppercase.

The command cannot be processed.

Action: Verify that the name is a permitted mode name. Verify the characters used in the name. Resubmit the command with a correct mode.

SQL1018N

The node name "<name>" specified in the CATALOG NODE command already exists.

Cause: The node name specified in the *nodename* parameter of the CATALOG NODE command is already cataloged in the node directory on this file system.

The command cannot be processed.

Action: If the *nodename* parameter is typed correctly, continue processing.

Uncatalog the cataloged node in the node directory if the node cataloged information is no longer valid and resubmit the command. If the node cataloged information is valid, define a new node name and resubmit the command using the new node name.

SQL1019N

The node name "<name>" specified in the command is not valid.

Cause: The node name specified in the command is not valid. The node name may be 1 to 8 characters and all of the characters must be from the database manager base character set.

The command cannot be processed.

Action: Resubmit the command with the correct node name.

SQL1020C

The node directory is full.

Cause: The node directory cannot hold any more entries.

The command cannot be processed.

Action: Uncatalog any unnecessary entries in the node directory.

SQL1021N

**The node name "\<name\>" specified in the
UNCATALOG NODE command was not found.**

Cause: The *nodename* specified in the command could
not be found in the node directory.

The command cannot be processed.

Action: If the *nodename* parameter is correct, the node
may already have been uncataloged, and processing
can continue. Otherwise, resubmit the command with
the correct node name.

SQL1022C

**There is not enough memory available to
process the command.**

Cause: Not enough random access memory (RAM) is
available to process the command.

If a remote procedure was invoked, the remote
procedure may use a local variable space that is larger
than the allowable maximum (4K).

If the statement involves a user-defined function (UDF),
the memory set controlled by the *udf_mem_sz* database
manager configuration parameter may be larger than the
memory available.

The command cannot be processed.

Action: Stop the application. Possible solutions
include:

- Change the MEMMAN NO SWAP, NO MOVE
  option in the CONFIG.SYS file to SWAP, MOVE.
- Remove background processes.
- Decrease the values of the configuration parameters
  that define allocation of memory, including
  udf_mem_sz if UDFs are involved in the failing
  statement.
- Install more random access memory (RAM).
- If a remote procedure was invoked, ensure that the
  remote procedure uses a local variable space that is
  less than or equal to 4K.
- If you are using Remote Data Services, increase the
  Remote Data Services heap size (rsheapsz) in the
  server and client configuration because at least one
  block is used per application.
- On OS/2 systems, change PROTECT to
  NOPROTECT in the MEMMAN statement of the

CONFIG.SYS file. This will make more memory
space available to applications, but disables certain
protection features of OS/2. Refer to the OS/2
documentation for more information and to
determine if this will be appropriate for your
environment.
- On OS/2 systems, increase the value of the
  min_priv_mem database manager configuration
  parameter. This will cause the database manager
  to reserve more private memory space at db2start
  time.

  NOTE: This is applicable only for releases of DB2
  prior to Version 2.

sqlcode: -1022

sqlstate: 57011

SQL1023C

The communications conversation failed.

Cause: An error occurred in the communications
conversation.

The command cannot be processed.

Action: Retry the original command. If the error
persists, contact your communications administrator.

DataJoiner users: this situation can be detected by
DataJoiner or by the data source.

sqlcode: -1023

sqlstate: 08001

SQL1024N

A database connection does not exist.

Cause: There is no connection to a database. Other
SQL statements cannot be processed unless an SQL
CONNECT was previously executed.

The command cannot be processed.

Action: If the error occurred when disconnecting from
the database, continue processing. If the error occurred
on another SQL statement, issue an SQL CONNECT
statement and resubmit the command or statement.

sqlcode: -1024

sqlstate: 08003

SQL1025N   SQL1031N

SQL1025N

The database manager was not stopped because databases are still active.

Cause: The stop database manager command cannot be processed if there are any applications connected to databases under control of the database manager, or if any databases have been activated.

No action is taken.

Action: Usually no action is required. To stop the database manager, all active applications must disconnect from all the databases they are using. Alternatively, the user can use the FORCE command to force applications to disconnect, and the DEACTIVATE command to deactivate the database.

SQL1026N

The database manager is already active.

Cause: A start database manager command is already processed.

The command cannot be processed.

Action: Because the command is already processed, the application may continue processing.

SQL1027N

The node directory cannot be found.

Cause: The list node directory command cannot be processed because the node directory could not be found.

The command cannot be processed.

Action: Issue a CATALOG NODE command with the appropriate parameters and resubmit the current command.

SQL1028N

"<adapter number>" is not a valid adapter parameter. It must be '0' for the primary adapter, or '1' for the secondary adapter.

Cause: The adapter number specified in the Database Environment command is not valid.

The command cannot be processed.

Action: Resubmit the command with a valid adapter number.

SQL1029N

The partner_lu alias "<name>" specified in the CATALOG NODE command is not valid.

Cause: The *partner_lu* alias specified in the CATALOG NODE command was not specified or contains characters that are not valid. The *partner_lu* alias is the partner SNA logical unit alias and must be 1 to 8 characters and must not contain blank characters.

The command cannot be processed.

Action: Examine the *partner_lu* for a typing error. Verify that the alias is a permitted logical unit name. Verify the characters used in the alias. Resubmit the command with a correct *partner_lu*.

SQL1030C

The database directory is full.

Cause: Either the system database directory or the local database directory cannot hold any more entries.

The command cannot be processed.

Action: Uncatalog any unnecessary entries in the directory. Create new databases on another file system if the local database directory is full.

SQL1031N

The database directory cannot be found on the indicated file system.

Cause: The system database directory or local database directory could not be found. A database has not been created or it was not cataloged correctly.

The command cannot be processed.

Action: Verify that the database is created with the correct path specification. The Catalog Database command has a path parameter which specifies the directory where the database resides.

sqlcode: -1031

sqlstate: 58031

SQL1032N

No start database manager command was issued.

Cause: The start database manager command has not been processed. It must be processed before a stop database manager, any SQL statement, or utility can be issued.

The command cannot be processed.

Action: Issue a start database manager command and resubmit the current command.

sqlcode: -1032

sqlstate: 57019

SQL1033N

The database directory cannot be scanned or used if it is currently being updated.

Cause: The database directory cannot be scanned or used if it is currently being updated. Also, the directory cannot be accessed for update if it is already being accessed for any reason. This situation could exist with either the system database directory or the local database directory.

The command cannot be processed.

Action: Wait until the update finishes, then resubmit the command.

sqlcode: -1033

sqlstate: 57019

SQL1034C

The database is damaged. The application has been disconnected from the database. All applications processing the database have been stopped.

Cause: Damage has occurred to the database. It cannot be used until it is recovered. All applications connected to the database have been disconnected and all processes running applications on the database have been stopped.

The command cannot be processed.

Action: Issue a RESTART DATABASE command to recover the database. If the RESTART command consistently fails, you may want to restore the database from a backup.

If you encountered this error during roll forward processing, you must restore the database from a backup and perform roll forward again.

If you are installing the sample database, drop it and install the sample database again.

sqlcode: -1034

sqlstate: 58031

SQL1035N

The database is currently in use.

Cause: One of the following conditions exists:

- Exclusive use was requested, but the database is already in use as a shared database by another user (in the same process).
- Exclusive use was requested, but the database is already in use as an exclusive database. (Two different processes are trying to access the same database.)
- The maximum number of connections to the database has been reached.
- The database is being used by another user on another system.

The command cannot be processed.

Action: Possible solutions include:

- Resubmit the command later when the database is not in use.
- Change the authorization to match the current user or wait until the database is not in use.
- Wait until the database is not in exclusive use.
- Wait until other users on another system have disconnected from the database.

sqlcode: -1035

sqlstate: 57019

SQL1036C

An I/O error occurred while accessing the database.

Cause: At least one database file has an I/O error:

- The system cannot open, read from, or write to a database file.
- The system cannot create the database because an error occurred while the system was creating a database file or a directory for the database.
- The system cannot drop the database because an error occurred while the system was deleting a database file or a directory for the database.

SQL1037W SQL1041N

The database cannot be used.

Action: If the error occurred during the processing of a database, resubmit the command. If the error continues, restore the database from a backup version.

If the error occurred during CREATE DATABASE or DROP DATABASE, a subsequent CREATE DATABASE or DROP DATABASE command attempts to erase files and directories left from the unsuccessful CREATE DATABASE or DROP DATABASE command.

If installing the sample database, drop it and install the sample database again.

sqlcode: -1036

sqlstate: 58030

SQL1037W

The Node directory is empty.

Cause: An attempt was made to read the contents of the node directory, but no entries existed.

Action: No action is required.

sqlcode: +1037

sqlstate: 01606

SQL1038C

An I/O error occurred while accessing the node directory.

Cause: The node directory could not be accessed because of an I/O error.

The command cannot be processed.

Action: Resubmit the command. If the error continues, remove the node directory (sqlnodir under the sqllib directory) and recatalog the node names on the network.

sqlcode: -1038

sqlstate: 58031

SQL1039C

An I/O error occurred while accessing the database directory.

Cause: Either the system database directory or the local database directory cannot be accessed. This error may occur not only when the system is cataloging or uncataloging a database but also when the system is accessing a database that is cataloged in the directory.

The command cannot be processed.

Action: Possible solutions include:

- If the error occurred on a diskette system, verify that the correct diskette is in the drive and ready to use. Verify that it is not write-protected.
- If the database directory is damaged, restore the databases that were cataloged there from backup versions and catalog them.

If installing the sample database, drop it and install the sample database again.

sqlcode: -1039

sqlstate: 58031

SQL1040N

The maximum number of applications is already connected to the database.

Cause: The number of applications connected to the database is equal to the maximum value defined in the configuration file for the database.

The command cannot be processed.

Action: Wait for other applications to disconnect from the database. If more applications are required to run concurrently, increase the value for *maxappls*. After all applications disconnect from the database and the database is restarted, the new value takes effect.

sqlcode: -1040

sqlstate: 57030

SQL1041N

The maximum number of concurrent databases have already been started.

Cause: The application attempted to start an inactive database but the number of active databases already equals the maximum value defined in the system configuration file.

The command cannot be processed.

Action: Wait for one of the databases to become inactive. If more databases are required to be active at the same time, increase the value for *numdb*. The new value takes effect after the next successful database manager start.

sqlcode: -1041

sqlstate: 57032

SQL1042C

An unexpected system error occurred.

Cause: A system error occurred. One possible reason for this error is that the database manager is not installed correctly or the environment is not set up correctly.

On OS/2, while trying to start the database manager, a very common reason for this error is a corrupted NET.ACC file.

The command cannot be processed.

Action: If the error occurred on OS/2 while trying to start the database manager, and the NET.ACC file is suspected, replace the NET.ACC files on the system with those on diskette 1 of the DB2 for OS/2 installation diskettes.

If the error occurred while trying to connect to a database, get a trace (instructions below) and call IBM support.

If the problem falls outside of the above suggestions, ensure that the system time and date is correctly set, and make sure your system has enough memory and swapping/paging space available.

Resubmit the current command.

If the error continues, stop the database manager and restart it.

If the error still continues, reinstall the database manager.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE or message number
- SQLCA contents if possible
- Trace file if possible.

DataJoiner users: if necessary, isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source). If the problem is with a data source, follow the problem determination procedures for that data source.

sqlcode: -1042

sqlstate: 58004

SQL1043C

The database manager could not initialize the system catalogs. Error "<error>" was returned.

Cause: The CREATE DATABASE command failed while initializing the system catalogs.

Action: Record the message number (SQLCODE) and error in this message.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Then contact your technical service representative with the following information:

- Environment: Application
- Required information:
  - Problem description
  - SQLCODE or message number and error identifier
  - SQLCA contents if possible
  - Trace file if possible.

SQL1044N

Processing was cancelled due to an interrupt.

Cause: The user may have pressed the interrupt key sequence.

Processing is stopped.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Continue processing to handle the interrupt.

If installing the sample database, drop it and install the sample database again.

sqlcode: -1044

sqlstate: 57014

SQL1045N

The database was not found because it was not cataloged correctly.

Cause: An indirect entry in the database directory points to another non-HOME entry.

The command cannot be processed.

Action: Verify the entries in all the associated database directories with the directory scan commands.

sqlcode: -1045

SQL1046N SQL1051N

sqlstate: 58031

SQL1046N

The authorization ID is not valid.

Cause: The authorization specified at logon is not valid for either the data source or the database manager. One of the following occurred:

- The authorization contains more than 8 characters.
- The authorization contains characters not valid for an authorization. Valid characters are A through Z, a through z, 0 through 9, #, @ and $.
- The authorization is PUBLIC or public.
- The authorization begins with SYS, sys, IBM, ibm, SQL or sql.
- The authorization violates some data source-specific naming convention.

The command cannot be processed.

Action: Log on with a valid authorization ID.

DataJoiner users: if necessary isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and use an authorization ID valid for that data source.

sqlcode: -1046

sqlstate: 28000

SQL1047N

The application is already connected to another database.

Cause: An application cannot create a database while connected to another database.

Binding a bind file to one database while already connected to another database is not permitted.

The command cannot be processed.

Action: Disconnect from the active database and resubmit the command.

SQL1048N

The use parameter "<parameter>" in the START USING DATABASE or CONNECT TO command is not valid. it must be an S for shared access or an X for exclusive use. For DDCS connections, only S is supported.

Cause: The *use* parameter in the START USING DATABASE or CONNECT TO command must be either an S for shared or an X for exclusive use. If connecting to a database using Distributed Database Connection Services (DDCS), only shared access is allowed. Mnemonic equates for these values are provided in the SQLENV.H file.

The command cannot be processed.

Action: Resubmit the command with a valid *use* parameter (preferably from the mnemonic equates).

SQL1049C

The application state is in error. The database connection has been lost.

Cause: The connection to the database has been lost.

The command cannot be processed.

Action: Issue a STOP USING DATABASE command.

SQL1050N

The database cannot be uncataloged because it is a home database.

Cause: The database specified in the UNCATALOG DATABASE command is a home database. Home databases cannot be uncataloged because their directory entries are deleted when the database is dropped.

The command cannot be processed.

Action: Continue processing if the database name was specified correctly.

SQL1051N

The path "<path>" for the database directory does not exist.

Cause: The path specified in the database directory parameter of the command or in the database directory entry is not valid. There is no file system with that name.

The command cannot be processed.

Action: Resubmit the command with the correct path for the database directory.

sqlcode: -1051

sqlstate: 57019

SQL1052N

The database path "<path>" does not exist.

Cause: The path specified in the *path* parameter of the command is not valid. There is no path with that name.

The command cannot be processed.

Action: Resubmit the command with the correct database path.

SQL1053N

An interrupt is already being processed.

Cause: The system is currently processing an interrupt so another interrupt is not honored.

The interrupt request is ignored.

Action: Wait until the current interrupt processing finishes and resubmit the command.

SQL1054N

A COMMIT is in progress and cannot be interrupted.

Cause: The system is currently processing a COMMIT. The user entered an interrupt key sequence.

The interrupt request is ignored.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Wait until COMMIT is finished and resubmit the request.

SQL1055N

A ROLLBACK is in progress and cannot be interrupted.

Cause: The system is currently processing a ROLLBACK. The user entered an interrupt key sequence.

The interrupt request is ignored.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Wait until ROLLBACK is finished and resubmit the request.

SQL1056N

There are already eight database directory scans open.

Cause: Eight database directory scans in this process are already open. No more than eight open scans are allowed.

The command cannot be processed.

Action: Issue one or more CLOSE DIRECTORY SCAN commands and resubmit the command.

sqlcode: -1056

sqlstate: 54029

SQL1057W

The system database directory is empty.

Cause: An attempt was made to read the contents of the system database directory, but no entries existed.

Action: No action is required.

sqlcode: +1057

sqlstate: 01606

SQL1058N

The handle parameter in the Directory Scan command is not valid.

Cause: The *handle* parameter specified in the Directory Scan command is not valid. The *handle* must be the one returned from the OPEN DIRECTORY SCAN or the OPEN NODE DIRECTORY SCAN command.

The command cannot be processed.

Action: Resubmit the command with a valid *handle* parameter.

SQL1059N   SQL1065W

SQL1059N

A Get Next command cannot be processed because no Open Scan command was issued.

Cause: A directory scan command was issued before the scan was opened.

The command cannot be processed.

Action: Issue an OPEN DIRECTORY SCAN or OPEN NODE DIRECTORY SCAN command and resubmit the current command.

SQL1060N

User "<authorization-ID>" does not have the CONNECT privilege.

Cause: The specified authorization does not have the CONNECT privilege to access the database. The CONNECT privilege must be granted before the user can connect to a database.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

The command cannot be processed.

Action: Contact the system administrator or database administrator for the database and request a GRANT CONNECT for the authorization ID. Resubmit the command.

DataJoiner users: if necessary, isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and ensure that correct authorizations exist on that data source.

sqlcode: -1060

sqlstate: 08004

SQL1061W

RESTART completed successfully, but indoubt transactions exist against the database.

Cause: A RESTART was successfully completed, except indoubt transactions were discovered. The database is usable, but if the indoubt transactions are not resolved before the last connection to the database is dropped, then the database will again require a RESTART before it can be used.

Action: Either resolve the indoubt transactions, or be prepared to RESTART the database whenever you wish

to use it. If the transaction manager (TM) that was using the database (in an XA/DTP environment) is available, then the administrator should instruct the TM to resolve the indoubt transactions. Alternatively, with great caution, the administrator can use the CLP to heuristically complete the transactions.

SQL1062N

The database path "<path>" was not found.

Cause: The database *path* parameter specified in the command does not exist. If the path is not specified, the default path defined in the system configuration file is used, and it does not exist.

The command cannot be processed.

Action: Verify the path or the default path and make sure they exist in the system. Resubmit the command.

SQL1063N

The start database manager processing was successful.

Cause: The command to start the database manager completed successfully.

SQL1064N

The stop database manager processing was successful.

Cause: The command to stop the database manager completed successfully.

SQL1065W

The database was created, but an error occurred on one or more bind files in list "<list-name>". The following files were not bound: "<list>".

Cause: One or more utilities were not bound to the database. The list file "<list-name>" contains a list of the bind files. The numbers in "<list>" indicate the relative positions of the unbound files in the list file.

The listed utility bind files are not bound to the newly created database.

Action: The user may bind the indicated utilities to the database. The format option must not be used on the binder call.

SQL1066N

DB2START processing was successful. IPX/SPX protocol support was not successfully started.

Cause: The IPX/SPX protocol support was not successfully started. Remote clients cannot use IPX/SPX to connect to the server. Possible causes are:

- The workstation is not logged in to the NetWare file server.

- The workstation does not have authority to create an object in the NetWare file server bindery.

- Another database manager on the network is using the same object name specified in the database manager configuration file.

Action: Ensure that the workstation is logged in to the NetWare file server, and has sufficient authority to create an object in the bindery at the file server. The user must be logged in as SUPERVISOR or equivalent. Also ensure that the object name specified in the database manager configuration file is unique for all database managers in the network. Make any corrections, run DB2STOP, and then run DB2START again.

If the problem continues, at the operating system command prompt type DB2TRC ON -L 0X100000. Run DB2START again, then at the command prompt, type DB2TRC DUMP filename to save the trace information. To turn trace off, type DB2TRC OFF. Contact your service coordinator with the trace information.

SQL1067N

DB2STOP processing was not successful. IPX/SPX protocol support was not successfully stopped.

Cause: The IPX/SPX protocol support was not successfully stopped. Possible causes are:

- The workstation is not logged in to the NetWare file server.

- The workstation does not have authority to delete an object in the NetWare file server bindery.

Action: Ensure that the workstation is logged in to the NetWare file server, and has sufficient authority to delete an object in the bindery at the file server. The user must be logged in as SUPERVISOR or equivalent. Make any corrections, and run DB2STOP again.

If the problem continues, at the operating system command prompt type DB2TRC ON -L 0X100000. Run DB2STOP again, then at the command prompt, type DB2TRC DUMP filename to save the trace information. To turn trace off, type DB2TRC OFF. Contact your service coordinator with the trace information.

SQL1069N

The database "<name>" is not a home database.

Cause: The database is not a local database. A local database is cataloged as indirect in the system database directory and this entry references a home entry in the local database directory on the same node. Remote databases cannot be dropped.

The command cannot be processed.

Action: If the specified database name was incorrect, resubmit the command with the correct database name. If the specified database name was correct and the intent is to remove the database name from the database directory, use the UNCATALOG DATABASE command.

SQL1070N

The address of the database name parameter is not valid.

Cause: The application program used an address that is not valid for the database name parameter. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

The command cannot be processed.

Action: Correct the application program so a correct address is used and the input string is null terminated.

SQL1071N

The address of the database alias name parameter is not valid.

Cause: The application program used an address that is not valid for this parameter. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program and the input string is null terminated.

SQL1072C SQL1074N

SQL1072C

The database manager resources are in an inconsistent state. the database manager may have been incorrectly terminated, or another application may be using system resources in a way which conflicts with the way the database manager uses system resources. System resource cleanup may be required.

Cause: The request failed because the database manager resources are in an inconsistent state. This can occur if:

- DB2 was incorrectly terminated (on UNIX-based systems, for example, this may occur if the processes were terminated with the "kill" command, rather than the stop database manager command).
- Another application or user may have removed the database manager resources (on UNIX-based systems, for example, a user with sufficient privilege may have accidentally removed an Interprocess Communication (IPC) resource owned by the database manager, using the "ipcrm" command).
- Another application's use of system resources conflicts with the database manager's use of system resources (on UNIX-based systems, for example, another application may be using the same keys that the database manager uses for creating IPC resources).
- Another instance of the database manager may be using the same resources. This could occur on UNIX-based systems if the two instances are on different file systems and the sqllib directories happen to have the same i-node (i-nodes are used to get the IPC keys).

Action: Resource cleanup may be required:

- Remove all the database manager processes running under the instance id.

  On UNIX-based systems, you can use "ps -eaf -u <instance id> | grep db2" to list all the database manager processes running under the instance id, and the "kill -9 <process id>" command to remove them.

  On NT-based systems:

  - For started service processes: Open the Control Panel, bring up the Services window, and select services that begin with "db2". Click the Stop button for each DB2 started service.

  - For general DB2 processes: Bring up the Task Manager, select the Processes tab, and search for processes that begin with "db2". Select a DB2 process and then click the End Process button. Repeat for each DB2 process.

- Ensure that no other application is running under the instance id, and then remove all resources owned by the instance id (on UNIX-based systems, you can use the "ipcs | grep <instance id>" command to list all IPC resources owned by the instance id, and the "ipcrm -[q|m|s] <id>" command to remove them).

- If there is another instance of the database manager running, and it is on a different filesystem, and an i-node conflict is suspected, then change the i-node by the following steps:

  - create another directory at the same level as sqllib
  - move everything from under sqllib to the new directory
  - delete sqllib
  - rename the new directory to sqllib

- Restart the database manager instance.

SQL1073N

The node directory release number is not correct.

Cause: The node directory release number does not match the product expected release number. The node directory could be from a previous release.

Action: Re-catalog all the node entries and try the command again.

SQL1074N

The address of the password parameter is not valid.

Cause: The application program used an address that is not valid for this parameter. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program and the input string is null terminated.

SQL1075N

The address of the database comment parameter is not valid.

Cause: The application program used an address that is not valid for this parameter. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program and the input string is null terminated.

SQL1076N

The address of the count parameter is not valid.

Cause: The application program used an address that is not valid for the *count* parameter.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program.

SQL1077N

The address of the handle parameter is not valid.

Cause: The application program used an address that is not valid for the *handle* parameter.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program.

SQL1078N

The address of the buffer parameter is not valid.

Cause: The application program used an address that is not valid for the *buffer* parameter.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program.

SQL1079N

The address of the nodename parameter is not valid.

Cause: The application program used a *nname* parameter address that is invalid. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program and the input string is null terminated.

SQL1080N

The address of the local_lu name parameter is not valid.

Cause: The application program used an address that is not valid for the *local_lu* name parameter. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program and the input string is null terminated.

SQL1081N

The address of the partner_lu parameter is not valid.

Cause: The application program has used an address that is not valid for the *partner_lu* parameter. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program and the input string is null terminated.

SQL1082N

The address of the mode parameter is not valid.

Cause: The application program has used an address that is not valid for the *mode* parameter. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

The command cannot be processed.

Action: Ensure that a valid address is used in the

SQL1083N   SQL1087W

application program and the input string is null terminated.

SQL1083N

The database description block cannot be processed, reason code="<reason-code>".

Cause: The application issued a CREATE DATABASE command, but the database descriptor block (DBDB) could not be processed for one of the following reason codes:

- The address of the DBDB is not valid (reason code 01).
- The value of the SQLDBDID field of the DBDB is not valid (reason code 02). It should be set to the value SQLDBDB1.
- The value of the SQLDBCSS field of the DBDB is not valid (reason code 04).

The command cannot be processed.

Action: Correct the error and resubmit the command.

SQL1084C

Shared memory segments cannot be allocated.

Cause: The database manager could not allocate segments while processing a Database Environment command or an SQL CONNECT statement.

The dbheap parameter may be too small.

Action: Record the message number (SQLCODE). The cause of this error may be insufficient memory resources either for the database manager or the environment where operation of the database manager is being attempted. Validate that sufficient memory resources are available to satisfy the database manager requirements. You may also close any background processes that do not need to be active.

If sufficient memory resources exist and the problem continues, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE or message number

- SQLCA contents if possible
- Trace file if possible.

sqlcode: -1084

sqlstate: 57019

SQL1085N

The application heap cannot be allocated.

Cause: The application could not connect to the database because the database manager could not allocate the number of application heap 4K pages specified in the database configuration file. The system is out of 4K pages. The command cannot be processed.

Action: Possible solutions are:

- Lower the size of the application heap (applheapsz) in the database configuration file.
- Reduce the maximum number of applications in the database configuration file.
- Remove background processes.
- Install more memory.

sqlcode: -1085

sqlstate: 57019

SQL1086C

An operating system error "<error>" occurred.

Cause: The command received an error from the operating system that prevents further processing.

The command cannot be processed.

Action: Refer to the operating system user publications to determine the nature of the error.

SQL1087W

The database was created, but an error occurred while opening the list file "<name>". The utilities are not bound to the database.

Cause: CREATE DATABASE could not open the list file containing the list of utility bind files. The list file should be located in the *bnd* subdirectory of the *sqllib* subdirectory.

The utility bind files are not bound to the newly created database.

Action: Bind the utilities to the database. Do not use the format option on the binder call.

SQL1088W

The database was created, but an error occurred while binding the utilities. The utilities are not bound to the database.

Cause: CREATE DATABASE or MIGRATE DATABASE could not bind the utility bind files to the database.

The utility bind files are not bound to the newly created or migrated database.

Action: Bind the utilities to the database. Do not use the format option on the binder call.

SQL1089W

The database was created, but the binding of the utilities was interrupted. The utilities are not bound to the database.

Cause: CREATE DATABASE was interrupted while binding the utilities to the database. The interrupt key sequence may have been pressed.

The utility bind files are not bound to the newly created database.

Action: Bind the utilities to the database. Do not use the format option on the binder call.

SQL1090C

The release number of the precompiled application program or utility is not valid.

Cause: The release number of the precompiled application program or utility is not compatible with the release number of the installed version of the database manager.

The command cannot be processed.

Action: Reinstall the database manager if the error occurred during the processing of a utility.

If the error occurred during the processing of an application program:

- Repeat the precompile process, if possible, with the current database manager.
- Use only application programs precompiled with a compatible release level of the database manager.

SQL1091C

The release number of the database is not valid.

Cause: The release number of the database is not compatible with the release number of the installed version of the database manager.

The command cannot be processed. If the error occurred during a Migrate or Restore, you are attempting to migrate or restore an incompatible release of database.

Action: Use only databases created with a compatible release of database manager. If the error occurred during a Migrate or Restore, you must first migrate the database to a release which can be migrated by the current release of the database manager.

SQL1092N

"<authorization-ID>" does not have the authority to perform the requested command.

Cause: The user attempted to execute a command or operation without having the proper authority for that command or operation.

The command cannot be processed.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Log on as a user with the correct authority and retry the failed command or operation. Correct authorities may include SYSADM, SYSCTRL, SYSMAINT, and DBADM. DBADM is granted on databases and all other authorities are determined by membership in the groups defined in the database manager configuration (eg. if *sysctrl_group* in the database manager configuration file is defined as 'beatles', then you must belong to the group 'beatles' to have SYSCTRL authority). Refer to the *Command Reference* or the *SQL Reference* for the listing of required authorities for the attempted command or operation.

DataJoiner users: if necessary, isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and ensure that correct authorizations exist on that data source.

Contact the System Administrator for authority request assistance. Do not attempt to execute the command without appropriate authorization.

SQL1093N SQL1097N

SQL1093N

The user is not logged on.

Cause: A user must be logged on before any
command requiring authorization can be processed.
Possible causes for this error include:

- Unable to get a user ID.
- An unexpected operating system error occurred
  when attempting to log on.
- The application is running in a background process.
- The user cancelled an attempted logon.

The command cannot be processed.

Action: Log on with a valid user ID and resubmit the
command. If several concurrent processes are
attempting to log on, wait a few seconds and retry the
logon procedure.

sqlcode: -1093

sqlstate: 51017

SQL1094N

**The node directory cannot be accessed because
it is being updated.**

Cause: The node directory cannot be scanned or used
while it is being updated. Also, the directory cannot be
accessed for update if it is already being accessed for
any reason.

The command cannot be processed.

Action: Resubmit the command after the update is
finished.

sqlcode: -1094

sqlstate: 57009

SQL1095N

**There are already eight node directory scans
open.**

Cause: Eight node directory scans are already open in
this process and no more than eight are allowed.

The command cannot be processed.

Action: Issue one or more CLOSE NODE
DIRECTORY SCAN commands. Resubmit the
command.

sqlcode: -1095

sqlstate: 54029

SQL1096N

The command is not valid for this node type.

Cause: A command was issued on a node that does
not support the command, or the system environment
was found to be set up incorrectly for this node type.
For example, a database was cataloged as LOCAL on a
client node.

The command cannot be processed.

Action: Verify that the command and parameters are
correct for the node type. Also verify that the
environment where the command is to be processed is
correct. Resubmit the command.

SQL1097N

**The node name was not found in the node
directory.**

Cause: Either the node name listed in the database
directory for a remote database or the node name
specified on an attach command was not cataloged in
the node directory.

The command cannot be processed.

DataJoiner users: this situation can be detected by
DataJoiner or by the data source.

Action: Verify that the node name listed in the
database directory or as the object of the attach
command is cataloged in the node directory. If the node
is not listed in the node directory, submit a CATALOG
NODE command.

DataJoiner users: in addition to the actions listed above,
also verify that the node names listed in all the
SYSCAT.SERVERS entries are correct. If the node is
not listed in the node directory and the server is a
member of the DB2 family, submit a CATALOG NODE
command for that node.

sqlcode: -1097

sqlstate: 42720

SQL1098N

The application is already connected to this database.

Cause: A connection to a database was requested but the application is already connected to the specified database.

The command cannot be processed.

Action: No action is necessary.

sqlcode: -1098

sqlstate: 53056

SQL1099N

The diskette is write-protected.

Cause: A write operation was attempted to a database residing on a write-protected diskette.

The statement cannot be processed.

Action: Verify you are using the correct diskette. Remove protection from the diskette, if necessary.

SQL1100W

The node name "<name>" specified in the Catalog Database command has not been cataloged in the Node Directory.

Cause: The Catalog Database command specified node name "<name>" but that is not cataloged in the node directory. The node name must be cataloged before the remote database can be used.

The CATALOG DATABASE command completed successfully.

Action: Issue a CATALOG NODE command.

SQL1101N

Remote database "<name>" on node "<node-name>" could not be accessed with the specified authorization id and password.

Cause: A connection to the remote database "<name>" on node "<node-name>" was requested, and the remote node does not accept the authorization ID and password combination specified for this node (either in the remote authorization tables or at runtime). The request cannot be processed.

Action: Resubmit the request with a valid authorization ID and password combination for the remote system.

SQL1102N

A database name was not specified.

Cause: No database name was provided to proceed with migration.

The statement cannot be processed.

Action: Specify a database name for migration.

SQL1103W

The Migrate Database command processing was successful.

Cause: The Migrate command completed successfully.

Note that this message will also be returned if the database was already at the current level and did not need to be migrated.

Processing continues.

Action: No action is required.

SQL1104N

The address of the program name parameter is not valid.

Cause: The application program has used an address for the program name that is not valid.

The command cannot be processed.

Action: Use a correct address in the application program.

SQL1105N

Neither the Stop Using Database environment command nor the SQL CONNECT RESET statement is allowed in a Remote Application Interface Procedure.

Cause: The remote application procedure contains an SQL CONNECT RESET statement or a call to the stop using database function.

The remote procedure is not allowed to continue.

Action: Remove the SQL CONNECT RESET statement or the call to the stop using database function and retry the remote procedure.

sqlcode: -1105

sqlstate: 42987

SQL1106N   SQL1111N

SQL1106N

The specified DLL "<name>" module was loaded, but the function "<function>" could not be executed.

Cause: The function within the DLL (dynamic link library) cannot be found.

The command cannot be processed.

Action: Ensure that the DLL module is built correctly. Reference the module in the definition file.

sqlcode: -1106

sqlstate: 42724

SQL1107N

An interrupt was received while loading the specified DLL "<name>".

Cause: When loading the DLL (dynamic link library) module, the command was interrupted, perhaps by Ctrl+Break.

Processing is stopped.

Action: Resubmit the command.

sqlcode: -1107

sqlstate: 42724

SQL1108N

An unexpected I/O error or operating system error was received while loading the specified DLL "<name>".

Cause: An unexpected error occurred when trying to load the DLL (dynamic link library) module specified in the program name field.

The command cannot be processed.

Action: Resubmit the current command. If the error continues, reinstall the database manager.

If reinstallation does not correct the error, record the message number (SQLCODE) and all information in the SQLCA if possible.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Then contact your technical service representative.

If an ODBC or CLI application receives this error when using DataJoiner for WindowsNT, the error might be due to a memory allocation problem between two DataJoiner DDLs. This problem can sometimes be resolved through the use of the DB2DBMSADDR environment variable. However, this variable should be set only under the direction of an authorized IBM service representative.

sqlcode: -1108

sqlstate: 42724

SQL1109N

The specified DLL "<name>" could not be loaded.

Cause: The DLL (dynamic link library) module specified could not be found.

The command cannot be processed.

Action: Ensure that the specified file is in a subdirectory identified in the system LIBPATH.

sqlcode: -1109

sqlstate: 42724

SQL1110N

A supplied data area was invalid and could not be used.

Cause: The data area is not initialized correctly.

The command cannot be processed.

Action: Ensure that the user-supplied input SQLDA or output SQLDA fields are initialized correctly.

SQL1111N

The program name "<name>" specified was not valid.

Cause: The DLL (dynamic link library) module or program name syntax is not correct.

The command cannot be processed.

Action: Ensure that the DLL or program name is specified correctly.

sqlcode: -1111

sqlstate: 42724

SQL1112N

There are not enough system resources to load the specified DLL "<name>".

Cause: There is not enough random access memory (RAM) to load the specified DLL (dynamic link library) module.

The command cannot be processed.

Action: Stop the application. Possible solutions are:

- Change the MEMMAN NO SWAP, NO MOVE option in the CONFIG.SYS file to SWAP, MOVE.
- Remove background processes.
- Decrease the values of the configuration parameters that define the allocation of memory.
- Install more random access memory (RAM).

sqlcode: -1112

sqlstate: 42724

SQL1113N

The data type for sqlvar "<n>" in the output SQLDA was changed from "<type 1>" to "<type 2>".

Cause: The remote stored procedure changed the data type of the *n*th sqlvar in the output SQLDA; *n* is the sequence number of the first mismatched sqlvar.

The stored procedure does not return any data.

Action: Correct the remote stored procedure so the data type information in the output SQLDA is not changed.

sqlcode: -1113

sqlstate: 39502

SQL1114N

The data length for the sqlvar "<n>" in the output SQLDA was changed from "<length 1>" to "<length 2>".

Cause: The remote stored procedure changed the data length of the *n*th sqlvar in the output SQLDA; *n* is the sequence number of the first mismatched sqlvar.

The stored procedure does not return any data.

Action: Correct the remote stored procedure so the data length information in the output SQLDA is not changed.

sqlcode: -1114

sqlstate: 39502

SQL1115N

The number of sqlvars in the output SQLDA was changed from "<count 1>" to "<count 2>".

Cause: The remote procedure changed the sqld field in the output SQLDA; sqld is the number of used sqlvars in the SQLDA.

The stored procedure does not return any data.

Action: Correct the remote stored procedure so the sqld field in the output SQLDA is not changed.

sqlcode: -1115

sqlstate: 39502

SQL1116N

A connection to or activation for database "<name>" cannot be made because of BACKUP PENDING.

Cause: The specified database requires a backup to be taken to provide a starting point for roll-forward recovery.

No connection was made.

Action: Back up the database by calling the BACKUP routine or, if roll-recovery is not required, set the log retain and user exit database configuration parameters off.

sqlcode: -1116

sqlstate: 57019

SQL1117N

A connection to or activation for database "<name>" cannot be made because of ROLL FORWARD PENDING.

Cause: The specified database is enabled for roll-forward recovery and it has been restored but not rolled forward.

No connection was made.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Roll forward the database or indicate that you do not wish to roll forward by using the ROLLFORWARD command. Note that if you do not roll forward the database, the records written since the last backup of the database will not be applied to the database.

SQL1118N   SQL1121N

DataJoiner users: if necessary, isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and take recovery action appropriate to that data source to bring the data source to a point of consistency.

sqlcode: -1117

sqlstate: 57019

```
 ─── SQL1118N ─────────────────────────────
  A connection to or activation of database
  "<name>" cannot be made because a previous
  backup is incomplete.
```

Cause: The database is in an inconsistent state because a system error occurred during a backup function.

No connection was made.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Issue the BACKUP command and then try the command again.

DataJoiner users: if necessary, isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and issue the BACKUP command against that data source prior to trying the command again.

sqlcode: -1118

sqlstate: 57019

```
 ─── SQL1119N ─────────────────────────────
  A connection to or activation of database
  "<name>" cannot be made because a previous
  restore is incomplete.
```

Cause: The database is in an inconsistent state because a system error occurred during a restore function.

No connection was made.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Issue the RESTORE command and then try the command again.

DataJoiner users: if necessary, isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and issue the RESTORE command against that data source prior to trying the command again.

sqlcode: -1119

sqlstate: 57019

```
 ─── SQL1120N ─────────────────────────────
  A connection to or activation of database
  "<name>" cannot be made because a previous
  backup or restore is incomplete.
```

Cause: The database is in an inconsistent state because a system error occurred during a backup or restore function. It cannot be determined if backup or restore was being processed.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

No connection was made.

Action: Issue the BACKUP or RESTORE command, and then try the command again.

DataJoiner users: if necessary, isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and issue the BACKUP or RESTORE command against that data source prior to trying the command again.

sqlcode: -1120

sqlstate: 57019

```
 ─── SQL1121N ─────────────────────────────
  The address of the node structure parameter is
  not valid.
```

Cause: The application used an address for the node structure parameter that is not valid. Either the address points to an unallocated buffer or the buffer is not large enough to contain the required input.

The command cannot be processed.

Action: Verify that the program allocates the required buffer area and resubmit the command.

SQL1122N

The address of the protocol structure parameter is not valid.

Cause: The application used an address for the protocol structure parameter that is not valid. Either the address points to an unallocated buffer or the address points to the incorrect protocol buffer.

The command cannot be processed.

Action: Verify that the program allocates the required buffer area based on the protocol field in the node structure and resubmit the command.

SQL1123N

The protocol "<type>" is not valid.

Cause: The protocol type specified in the node structure for the Catalog command does not represent a recognized value. The valid protocol types are defined in the *sqlenv* header files.

The command cannot be processed.

Action: Verify the protocol type in the node structure and resubmit the command.

SQL1124N

The remote workstation name "<name>" is not valid.

Cause: The remote workstation name specified in the NETBIOS protocol structure for the Catalog command is not specified or contains characters that are not valid. The workstation name must be 1 to 8 characters. Valid characters are A through Z, a through z, 0 through 9, #, @, and $. The first character must be alphabetic or a special character: #, @, or $. Lowercase characters are changed to uppercase by the system.

The command cannot be processed.

Action: Verify the characters specified in the remote workstation name. Resubmit the command with a valid workstation name.

SQL1125N

The adapter number "<number>" is not valid.

Cause: The adapter number specified in the NETBIOS protocol structure for the Catalog command is not valid.

The command cannot be processed.

Action: Verify that the adapter number is valid and resubmit the command.

SQL1126N

The network ID "<ID>" is not valid.

Cause: The network ID specified in the APPN protocol structure for the Catalog command is not valid. The network ID identifies the SNA network where the remote logical unit (LU) resides. The network ID must be 1 to 8 characters. Valid characters are A through Z, a through z, 0 through 9, #, @, and $. The first character must be alphabetic or a special character: #, @, or $. Lowercase characters are changed to uppercase by the system.

The command cannot be processed.

Action: Verify the characters specified in the network ID. Resubmit the command with a valid network ID.

SQL1127N

The remote LU name "<name>" is not valid.

Cause: The remote logical unit (LU) name specified in the APPN protocol structure for the Catalog command is not valid. The remote LU name is the remote SNA logical unit name and must be 1 to 8 characters. Valid characters are A through Z, a through z, 0 through 9, #, @, and $. The first character must be alphabetic or a special character: #, @, or $. Lowercase characters are changed to uppercase by the system.

The command cannot be processed.

Action: Verify the characters specified in the remote LU name. Resubmit the command with a remote LU name.

SQL1129N   SQL1134N

SQL1129N

New DARI (Stored Procedure) process could not be started because there are insufficient resources for a new process to be created.

Cause: New DARI (Stored Procedure) process could not be started because there are insufficient resources for a new process to be created.

Action: Do one of the following:

- Reduce the number of users using DB2
- Increase the system process limit.

sqlcode: -1129

sqlstate: 42724

SQL1130N

New DARI (Stored Procedure) process could not be started because the maximum number of DARI processes allowed to be present at the server at the same time has been reached.

Cause: New DARI (Stored Procedure) process could not be started because the maximum number of DARI processes allowed to be present at the server at the same time has been reached.

Action: Increase the maximum number of DARI processes, *maxdari*, configuration parameter. See the database configuration parameter *maxdari* for more details.

sqlcode: -1130

sqlstate: 42724

SQL1131N

DARI (Stored Procedure) process has been terminated abnormally.

Cause: The cause of this error may be:

- There was a coding error (e.g. segmentation violation) within the DARI routine.
- The DARI process has been terminated by another process through the use of a signal.

Action: Reinitiate the DARI request after doing the following :

- Ensure that the DARI procedure is free from programming errors.
- Make sure that no user is sending a termination signal to the DARI process.

sqlcode: -1131

sqlstate: 38503

SQL1132N

The command is not allowed to execute within a DARI (Stored Procedure).

Cause: An illegal command has been made within the scope of DARI (Stored Procedure).

The DARI procedure is not allowed to continue.

Action: Remove the illegal command in the DARI procedure and retry.

sqlcode: -1132

sqlstate: 42987

SQL1133N

A pointer address inside an sqlvar (index = "<n>") of the output SQLDA was modified within the DARI (Stored Procedure) function.

Cause: The "sqlind" or the "sqldata" pointer within the sqlvar of the output SQLDA has been altered within the user prepared DARI function.

The stored procedure does not return any data.

Action: Correct the usage of the indicated sqlvar within the output SQLDA so that pointer addresses are not changed within the DARI (Stored Procedure) function routine.

sqlcode: -1133

sqlstate: 39502

SQL1134N

For database authentication type CLIENT, the command is not allowed within the scope of DARI (Stored Procedure).

Cause: All SYSADM commands are not allowed to execute within DARI (Stored Procedure) when the database authentication type is CLIENT.

The stored procedure does not return any data.

The DARI procedure is not allowed to continue.

Action: Remove the illegal command in the DARI procedure and retry.

sqlcode: -1134

sqlstate: 42987

SQL1135N

An invalid value was specified for the number of segments when creating a database.

Cause: The value that was specified for the number of segments was out of range. The valid range is from 1 to 256.

Action: Respecify the number of segments and try to create the database again.

SQL1136N

An invalid value was specified for the default table space extent size (dft_extentsize) when creating a database.

Cause: The value that was specified for the default table space extent size (dft_extentsize) is out of range. The valid range is from 2 to 256.

Action: Correct the table space extent size and try again.

SQL1137W

The database manager was unable to remove the database path or some of the containers when dropping database "<dbalias>". Cleanup is required.

Cause: Either the list of containers could not be accessed or some failure occurred while attempting to remove them or the database directory.

Action: System managed containers (directories) and database managed file containers may be manually removed using operating system commands. To release device containers, please call your IBM service representative for assistance.

If the log directory has been changed via the New Log Path configuration parameter, then manually unmount the log directory file system and remove the log and database directories.

SQL1150N

The address of the user id parameter is not valid.

Cause: The application program used an address that is not valid for this parameter. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

The command cannot be processed.

Action: Ensure that a valid address is used in the application program and the input string is null terminated.

SQL1160N

The DOS "<network protocol>" TSR has not been loaded.

Cause: The Terminate and Stay Resident (TSR) network driver for the specified communications protocol has not been loaded. The TSR must be loaded before attempting to use network communications.

Action: Ensure that the TSR for the specified communications protocol has been successfully loaded, then run your application again.

SQL1200N

The object parameter is not valid.

Cause: The value specified for the object parameter in the COLLECT DATABASE STATUS function call is not valid. Valid values are:

SQLE_DATABASE
Status is to be collected for a single database.

SQLE_DRIVE
Status is to be collected for all LOCAL databases on a single path.

SQLE_LOCAL
Status is to be collected for all LOCAL databases.

The command cannot be processed.

Action: Correct the object parameter and resubmit the COLLECT DATABASE STATUS function call.

SQL1201N   SQL1207N

SQL1201N

The status parameter is not valid.

Cause: The value specified for the status parameter in the COLLECT DATABASE STATUS function call is not valid. Valid values are:

SQLE_SYSTEM
> System status is to be collected.

SQLE_DATABASE
> System status and database status is to be collected.

SQLE_ALL
> System status, database status and user status is to be collected.

The command cannot be processed.

Action: Correct the status parameter and resubmit the COLLECT DATABASE STATUS function call.

SQL1202N

Status has not been collected.

Cause: The value specified for the *handle* parameter in the GET NEXT DATABASE STATUS BLOCK or the FREE DATABASE STATUS RESOURCES function calls is not valid. The handle must be the one returned as the positive function value from the COLLECT DATABASE STATUS function call.

This is the second COLLECT DATABASE STATUS call made from the process. The first COLLECT DATABASE STATUS call terminates and its handle can no longer be used.

The command cannot be processed.

Action: Correct the *handle* parameter and resubmit the COLLECT DATABASE STATUS function call.

SQL1203N

The database has no users connected.

Cause: A request was made for user status on a database, but the database has no users connected.

The command cannot be processed.

Action: Verify the database name and the connection status. Resubmit the command with a database currently being used.

SQL1204N

The code page "<code page>" and/or country code "<country code>" is not supported by the installed version of the database manager.

Cause: This version of the database manager does not support the active code page or country code or both.

The command cannot be processed.

Choose an active code page and country code supported by this version of the database manager.

DataJoiner users: if necessary, isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and choose an active code page and country code supported by both DataJoiner and the data source.

Action: Exit the current program and return to the operating system.

sqlcode: -1204

sqlstate: 22522

SQL1205N

The code page "<code page>" and/or country code "<country>" that has been specified is not valid.

Cause: This version of DB2 does not support the active code page or country code or both, as specified by the Create Database command.

The command cannot be processed.

Action: Refer to the Create Database command in the *Command Reference* for details of valid code page and country code pairs supported by DB2/2.

SQL1207N

Communication Manager configuration file "<name>" cannot be found.

Cause: The Communication Manager configuration file name specified on the CATALOG NODE command was not found either in the path specified or in the CMLIB directory on the default drive.

The command cannot be processed.

Action: Resubmit the command with the correct configuration file name and path.

SQL1209W

The partner_lu name "<name>" specified in the CATALOG NODE function does not exist. One was created.

Cause: The logical partner unit name specified in the CATALOG NODE function does not exist in the Communications Manager configuration file located in the CMLIB directory on the default drive.

A logical unit profile of the specified name was created.

Action: No action is required.

SQL1210W

Default values were returned for one or more DOS Requester/WINDOWS Requester configuration file parameters.

Cause: One or more of the DOS Requester/WINDOWS Requester configuration values returned was defaulted. The parameter may not be defined in the DOS Requester/WINDOWS Requester configuration file or the configuration file could not be opened or an error occurred reading the file.

Action: Ensure that the DOS Requester/WINDOWS Requester configuration file exists in the appropriate path and that the parameters are explicitly defined.

SQL1216N

Graphic data and graphic functions are not supported for this database.

Cause: The code page of the database does not support graphic data. The data types GRAPHIC, VARGRAPHIC, and LONG VARGRAPHIC are not valid for this database. The GRAPHIC literal and the VARGRAPHIC scalar functions are not valid for this database.

The statement cannot be processed.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Resubmit the command with valid data types.

sqlcode: -1216

sqlstate: 56031

SQL1219N

The request failed because private virtual memory could not be allocated.

Cause: The instance was unable to allocate enough private virtual memory to process the request. This may be a result of shared memory allocations made in other (unrelated) processes.

Action: The problem may be corrected by:

- On OS/2, increasing the *min_priv_mem* configuration parameter. This will result in more private virtual memory being reserved when the instance is started.
- Stopping other applications running on the machine, especially those that use large amounts of shared memory.

sqlcode: -1219

sqlstate: 57011

SQL1220N

The database manager shared memory set cannot be allocated.

Cause: The database manager could not allocate its shared memory set. The cause of this error may be insufficient memory resources either for the database manager or the environment in which its operation is being attempted. Memory resources that can cause this error include:

- The number of shared memory identifiers allocated in the system
- The amount of paging or swapping space available in the system
- The amount of physical memory available in the system

Action: One or more of the following:

- Validate that sufficient memory resources are available to satisfy the database manager's requirements, and those of the other programs running on the system.
- Reduce database manager's memory requirement for this memory set by reducing the database manager configuration parameters which affect it. These are: *maxagents*, *maxdari* and, *numdb*.
- Where appropriate, stop other programs using the system.

SQL1221N SQL1224N

SQL1221N

The Application Support Layer heap cannot be allocated.

Cause: The Application Support Layer heap could not be allocated. The cause of this error may be insufficient memory resources either for the database manager or the environment in which its operations are being attempted. Memory resources that can cause this error include:

- The number of shared memory identifiers allocated in the system
- The amount of paging or swapping space available in the system
- The amount of physical memory available in the system

Action: One or more of the following:

- Validate that sufficient memory resources are available to satisfy the database manager's requirements, and those of the other programs running on the system.
- Reduce the *aslheapsz* configuration parameter.
- Where appropriate, stop other programs using the system.

sqlcode: -1221

sqlstate: 57011

SQL1222N

Not enough storage is available in the Application Support Layer heap to process the request.

Cause: All available memory in the Application Support Layer heap has been used.

Action: Increase the *aslheapsz* configuration parameter.

sqlcode: -1222

sqlstate: 57011

SQL1223N

An agent could not be started to service this request.

Cause: The request failed because the *maxagents* configuration parameter limit would be exceeded.

Action: Increase the *maxagents* configuration parameter, and/or reduce the number of users using the database.

sqlcode: -1223

sqlstate: 57019

SQL1224N

A database agent could not be started to service a request, or was terminated as a result of a database system shutdown or a force command.

Cause: The message may be attributed to any of the following cases.

- The database manager has not been started on the database server
- The database manager was stopped
- The database manager has already allocated the maximum number of agents
- The database agent was forced off by the system administrator
- The database agent was terminated due to an abnormal termination of a key database manager process
- The application is using multiple contexts with local protocol. In this case the number of connections is limited by the number of shared memory segments to which a single process can be attached. For example, on AIX, the limit is ten shared memory segments per process.

Additional DataJoiner cases are:

- The maximum number of processes per user (maxuproc on AIX) at the operating system level has been exceeded.
- In the client/server environment using the TCP/IP protocol, the port number assigned to the TCP/IP service name on the client is not the same as the port number on the server.

This situation can be detected by DataJoiner or by the data source.

Action: Reissue the database request. If a connection cannot be established, then ensure that the database manager has been started successfully. Additionally,

ensure that the *maxagents* database manager configuration parameter is appropriately configured.

DataJoiner users, should also:

- Isolate the problem to the data source rejecting the request (see the *Problem Determination Guide* for procedures to follow to identify the failing data source) and ensure that the communication subsystem is active and that the database manager and required communication protocol server processes have been started on the database server.
- For AIX operating systems, check the setting of maxuproc and change it if needed. maxuproc restricts the number of processes that can run under a given DataJoiner server. The default setting is 40.

You can check the current setting of maxuproc with the command:

```
.
   lsattr -E -l sys0
```

To view the number of processes currently running under a given DataJoiner server, use the command:

```
ps -ef | grep instdj1 | wc -l
```

where "instdj1" is the instance name of the DataJoiner server.

To change maxuproc, use the command:

```
chdev -l sys0 -a maxuproc='nn'
```

where **nn** is the new integer value of maxuproc.

If the application is using multiple contexts with local protocol, then either reduce the number of connections in the application, or switch to another protocol (for example, TCP/IP).

sqlcode: -1224

sqlstate: 55032

SQL1225N

The request failed because an operating system process or thread limit was reached.

Cause: An operating system process or thread limit was reached. On UNIX-based systems, this may mean that the maxuproc value is too low. On OS/2-based systems, this may mean that the THREADS CONFIG.SYS value is too low.

Action: Increase the limit which was reached (or have your system administrator increase it). On UNIX-based systems, the maxuproc limit can be changed with the chdev command.

sqlcode: -1225

sqlstate: 57049

SQL1227N

The catalog statistic "<value>" for column "<column>" is out of range for its target column, has an invalid format, or is inconsistent in relation to some other statistic. Reason Code = "<code>"

Cause: The value or format of a statistic provided for an updateable catalog is either invalid, out of range, or inconsistent. The most common checks for value, range and format are (corresponding to "<code>"):

1 Numeric statistics must be -1 or >= 0.

2 Numeric statistics representing percentages (eg. CLUSTERRATIO) must be between 0 and 100.

3 HIGH2KEY, LOW2KEY related rules :

- The datatype of any HIGH2KEY, LOW2KEY value must be the same as the datatype of the corresponding user column.
- The length of HIGH2KEY, LOW2KEY value must be the smaller of 33 or the target column's datatype's maximum length.
- HIGH2KEY must be > LOW2KEY whenever there are 3 or more distinct values in the corresponding column. In the case of less than 3 distinct values in the column, HIGH2KEY can be equal to LOW2KEY.

4 PAGE_FETCH_PAIRS related rules :

- Individual values in the PAGE_FETCH_PAIRS statistic must be separated by a series of blank delimiters.
- Individual values in PAGE_FETCH_PAIRS statistics must not be longer than 10 digits and must be less than the maximum integer value (MAXINT = 2147483647).
- There must always be a valid PAGE_FETCH_PAIRS value if the CLUSTERFACTOR is > 0.
- There must be exactly 11 pairs in a single PAGE_FETCH_PAIR statistic
- buffer size entries of PAGE_FETCH_PAIRS must be ascending in value. Also, any buffer size value in a PAGE_FETCH_PAIRS entry cannot be greater than MIN(NPAGES, 524287) where NPAGES is the number of pages in the corresponding table.
- "fetches" entries of PAGE_FETCH_PAIRS must be descending in value, with no

individual fetches entry being less than NPAGES. Also, any "fetch" size value in a PAGE_FETCH_PAIRS entry must be greater than CARD (cardinality) statistic of the corresponding table.

- If buffer size value is the same in two consecutive pairs, then page fetch value must also be the same in both the pairs.

5   CLUSTERRATIO and CLUSTERFACTOR related rules :

- Valid values for CLUSTERRATIO are -1 or between 0 and 100
- Valid values for CLUSTERFACTOR are -1 or between 0 and 1
- Either CLUSTERRATIO or CLUSTERFACTOR must be -1 at all times.
- If CLUSTERFACTOR is a positive value, it must be accompanied by a valid PAGE_FETCH_PAIR statistic.

6   The cardinality of a column (COLCARD statistic in SYSCOLUMNS) cannot be greater than the cardinality of its corresponding table (CARD statistic in SYSTABLES).

7   No statistics are supported for columns with datatype: LONG VARCHAR, LONG VARGRAPHIC, BLOB, CLOB, DBCLOB.

8   A statistic is inconsistent with other related statistics for this entity or is invalid in this context.

Action: Make sure the new catalog statistic satisfies the above range/length/format checks.

Make sure that any updates to statistics are consistent in their inter-relationships (eg. cardinality).

sqlcode: -1227

sqlstate: 23521

SQL1230W

At least one agent id specified could not be forced.

Cause: At least one agent id specified on the Force command could not be forced. Causes of this warning may be:

- A nonexistent or invalid agent id was specified.
- The agent disconnected from the database manager

between the time that the agent ids were collected and the Force command was issued.

Action: If a nonexistent or invalid agent id was specified, retry the command using the valid agent id.

SQL1231N

Invalid Force count specified.

Cause: The value specified for the count parameter of the Force command is not valid. The value specified must either be a positive integer or SQL_ALL_USERS. A value of 0 will result in an error.

Action: Correct the value of count and resubmit the command.

SQL1232N

Invalid Force mode specified.

Cause: The value specified for the mode parameter of the Force command is not valid. Only asynchronous mode is supported for the Force command. The parameter must be set to the value SQL_ASYNCH.

Action: Set the mode to SQL_ASYNCH and resubmit the command.

SQL1240N

The maximum number of quiescers has been reached.

Cause: An attempt was made to acquire a quiesce state on a table space which has already been quiesced by 5 processes.

Action: Wait for one of the processes to release its quiesce state and try again.

SQL1241N

An invalid value was specified for the "<tbs-name>" table space definition when creating a database. Attribute is "<string>".

Cause: The value for a table space attribute was out of range. See the Data Structures section of API Reference for the format of the sqletsdesc structure used for the create database api. The identified attribute is the field name of this structure.

Action: Correct the create database request.

SQL1244W

Disconnect for transaction manager database "<database-name>" will occur at next COMMIT.

Cause: A disconnect has been issued against a database which is acting as the TM database. The disconnect cannot be completed until the next COMMIT is processed.

Action: If it is necessary that the database which is acting as the TM database be disconnected immediately, issue a COMMIT statement before execution continues.

sqlcode: +1244

sqlstate: 01002

SQL1245N

Connection limit has been reached. No more connections are allowed from this client.

Cause: The maximum number of concurrent database connections has been reached for an environment where the number of connections is restricted or must be predefined. The primary example of where this would occur is when the NETBIOS protocol is being used.

Action: Possible actions:

- Use the SET CLIENT command or API to set the MAX NETBIOS CONNECTIONS field to the maximum number of concurrent connections required. This must be done before any connections are made.

sqlcode: -1245

sqlstate: 08001

SQL1246N

Connection settings cannot be changed while connections exist.

Cause: One of the following occurred:

- An attempt was made to change the connection settings for an application using the SET CLIENT API. This was rejected because one or more connections exist.
- The application contains both DB2 Call Level Interface API calls and calls to functions containing embedded SQL, and the connection management was not invoked using the CLI APIs.

Action: Possible actions:

- Ensure that the application disconnects from all servers before attempting to issue the SET CLIENT API (sqlesetc or sqlgsetc) or CLP command.
- Ensure that all connection management requests are issued via the DB2 Call Level Interface APIs if CLI is being used by the application.

SQL1247N

Applications running in an XA transaction processing environment must use the SYNCPOINT TWOPHASE connection setting.

Cause: The application was precompiled with the option SYNCPOINT ONEPHASE or SYNCPOINT NONE, or the SYNCPOINT connection setting was changed to one of these values using the SET CLIENT API. These settings are invalid for applications which cause transaction processing syncpoint commands (for example CICS SYNCPOINT) to be executed. Note that SYNCPOINT ONEPHASE is the default precompiler option.

Action: Possible actions:

- Re-precompile the application using the precompiler option SYNCPOINT TWOPHASE.
- Modify the program such that the SET CLIENT API is called to set the connection option to SYNCPOINT TWOPHASE prior to any other processing.

sqlcode: -1247

sqlstate: 51025

SQL1248N

Database "<database alias>" not defined with the transaction manager.

Cause: You attempted to access a database that had not been opened by the transaction manager. A database must be defined to the transaction manager in order to participate in the two-phase commit process.

Action: Define the database as a resource manager to the transaction manager of your Distributed Transaction Processing environment. For example, in a CICS environment this requires adding the database to the XAD file and provides the database alias name in the XAOpen string of the XAD entry.

sqlcode: -1248

sqlstate: 42705

SQL1251W   SQL1263N

SQL1251W

No data returned for heuristic query.

Cause: There are no in-doubt transactions in the database, nor any transactions that have ended and are waiting to enter the syncpoint process.

Action: No action is required.

SQL1258N

The log control file for database "<name>" could not be found.

Cause: File SQLOGCTL.LFH for the specified database cannot be found in the database directory.

The database was not started.

Action: Restore the database from a backup or recreate the database.

SQL1259N

An I/O error occurred while accessing the log control file for database "<name>".

Cause: An error occurred while accessing SQLOGCTL.LFH for the specified database.

The database cannot be used.

Action: Restore the database from a backup or recreate the database.

SQL1260N

Database "<name>" is not configured for roll-forward recovery.

Cause: The specified database is not configured for roll-forward recovery.

The database is not rolled forward.

Action: Ensure that recovery is needed, then restore the most recent backup version of the database.

SQL1261N

Database "<name>" does not need to be rolled forward.

Cause: The specified database does not need to be rolled forward. This may be because the database has not been restored, or the RESTORE or ROLLFORWARD command was called with the option that resets the ROLL FORWARD PENDING flag.

The database is not rolled forward.

Action: Ensure that a backup version of the database has been restored and then submit the ROLLFORWARD command.

SQL1262N

The point-in-time specified for rolling forward database "<name>" is not valid.

Cause: The timestamp parameter specified for the point-in-time stopping value is not valid. The timestamp must be entered in ISO format (YYYY-MM-DD-hh.mm.ss.<ssssss> where YYYY represents year, MM represents month, DD represents day, hh represents hours, mm represents minutes, ss represents seconds, and ssssss represents optional microseconds).

The database is not rolled forward.

Action: Ensure that the timestamp is entered in the correct format.

When you issue a ROLLFORWARD DATABASE command, ensure that you are not specifying a year greater than 2105.

SQL1263N

The archive file "<name>" is not a valid log file.

Cause: An archive log file was found in the specified database log directory or the overflow log directory, but the file is not valid.

Roll forward recovery processing stops.

Action: To determine the correct archive log file, call the ROLLFORWARD command with the QUERY STATUS option. Move the correct archive log file into the database log directory or, if the database is in a consistent state, change the Log Path to point to the correct archive file and resubmit the ROLLFORWARD command. Alternatively, resubmit the command with the overflow log path pointing to the correct archive file.

SQL1264N

The archive file "<name>" does not belong to database "<database-name>".

Cause: An archive log file was found in the log directory or the overflow log directory that does not belong to the specified database.

Roll forward recovery processing stops.

Action: To determine the correct archive log file, call the ROLLFORWARD command with the QUERY STATUS option. Move the correct archive log file into the database log directory or, if the database is in a consistent state, change the Log Path to point to the correct archive file and resubmit the ROLLFORWARD command. Alternatively, resubmit the command with the overflow log path pointing to the correct archive file.

SQL1265N

The archive log file "<name>" is not associated with the current log sequence for the database.

Cause: For roll forward recovery, the log files need to be processed in correct sequence. The log file sequence is determined by the database that has been restored or the log files that have been processed. In addition to this, for table space level roll forward recovery, the log files must be processed in the sequence through which the current state of the database has been reached. An archive log file was found in the log directory for the database or in the overflow log path but the log file is not in the correct log sequence.

Roll forward recovery processing stops.

Action: Move the correct archive log file into the database log directory or, if the database is in a consistent state, change the log path to point to the correct archive file and resubmit the ROLLFORWARD command. Alternatively, resubmit the command with the overflow log path pointing to the correct archive file.

SQL1266N

Database "<name>" has been rolled forward to "<timestamp>", which is past the specified point-in-time.

Cause: A stop position timestamp was specified that is before the current point in the specified database log file.

Database roll-forward processing stops.

Action: Specify the correct point in time or restore the

database from a backup version and resubmit the ROLLFORWARD command.

SQL1267N

The system could not find db2uexit in the current PATH environment variable.

Cause: The user-supplied file *db2uexit* cannot be found, either because it is not in the current PATH environment variable or it does not exist.

The command cannot be processed.

Action: Update the current PATH environment variable to include the path to *db2uexit* or create a *db2uexit* file and update the current PATH environment variable if necessary.

SQL1268N

Roll forward recovery stopped due to error "<error>" while retrieving log file "<logfile>". The database is still in a roll forward pending state.

Cause: Roll forward processing invokes *db2uexit* to retrieve the log file. The error may have occurred in *db2uexit*.

Roll forward processing stopped. The database is still in a roll forward pending state.

Action: See the user exit documentation in the *Administration Guide* for a description of the error and resume or terminate roll forward recovery.

SQL1269N

Table space level roll forward recovery is already running.

Cause: An attempt has been made to use table space level roll forward recovery but it is already running. Only one agent can be executing roll forward recovery at any given time.

Action: Wait until table space level roll forward recovery is complete. If more table spaces need recovery, initiate table space level roll forward recovery again.

SQL1270C SQL1276N

SQL1270C

The LANG environmental variable is set to "<string>". This language is not supported.

Cause: The LANG environmental variable is set to a language that is not supported by the database manager. Processing cannot continue.

Action: Set the LANG environmental variable to a language that is supported. For more information please refer to "National Language Support" appendix in the *Administration Guide*.

DataJoiner users: see the *IBM DataJoiner Planning, Installation, and Configuration Guide* for NLS information.

SQL1271W

The database is recovered but errors occurred on one or more table spaces.

Cause: This message can occur after crash recovery, database level roll forward recovery, or table space level roll forward recovery. In the first two cases, the database has been recovered and is available for use but there are one or more table spaces that received errors during recovery and have been taken off-line. In the third case, one or more of the table spaces being rolled forward received errors, so roll forward recovery for those table spaces only was stopped.

Action: The table spaces that are off-line can be identified by the list tablespaces command. Repair or restore the table spaces and run roll forward recovery.

SQL1272W

Table space level roll forward recovery has stopped before completion.

Cause: Table space level roll forward recovery has stopped before all off-line table spaces could be rolled forward. This can occur if the transaction table becomes full or if all table spaces that are being rolled forward receive I/O errors.

Action: Query the table space system catalog to determine if the table spaces received I/O errors again. If so, repair them and run table space level roll forward recovery again. If not, the transaction table became full. Try running table space level roll forward recovery offline.

SQL1273N

The archive log file "<name>" was not found.

Cause: The archive log file was not found in the specified database log directory or the overflow log directory.

Table space roll forward recovery has stopped.

Action: Move the specified archive log file into the database log directory or, if the database is in a consistent state, change the log path to point to the correct archive file and resubmit the ROLLFORWARD command. Alternatively, resubmit the command with the overflow log path pointing to the correct archive file.

SQL1274N

The database "<name>" requires table space level roll forward recovery and the point-in-time must be to the end of logs.

Cause: The database needs to be rolled forward at the table space level. For table space level roll forward recovery, the point-in-time must be to the end of logs.

The database is not rolled forward.

Action: Resubmit the ROLLFORWARD command specifying ROLLFORWARD TO END OF LOGS.

SQL1275N

The stoptime passed to Roll Forward must be greater than or equal to "<timestamp>".

Cause: The database to be Rolled Forward was backed up online. The stoptime passed to Roll Forward must be greater than or equal to the time the online backup ended. Time must be specified in CUT (Coordinated Universal Time).

Action: Resubmit the command with a stoptime greater than or equal to "<timestamp>".

SQL1276N

The database cannot be brought out of rollforward pending state until Roll Forward has passed a point in time greater than or equal to "<timestamp>".

Cause: A request was made to bring the database out of rollforward pending state by specifying the "stop" option (CallerAction: SQLUM_ROLLFWD_STOP or SQLUM_STOP). However, the database to be Rolled Forward was backed up online. The request cannot be

---

granted until the database has been Rolled Forward to the end of the online backup timestamp.

Action: Make sure the database has reached or advanced past "<timestamp>" before specifying the "stop" option.

SQL1277N

Restore has detected that one or more tablespace containers are inaccessible, or has set their state to "storage must be defined".

Cause: Restore validates that the containers needed by each tablespace being restored are currently accessible on the system. Where possible, restore will create the containers if they do not exist. If they cannot be created, or are currently in use by another tablespace, or are inaccessible for any other reason, then the list of containers needed must be corrected before the restore can continue.

If this is a redirected restore, the state of each container of the tablespaces being restored is set to "storage must be defined". This makes it possible to use the SET TABLESPACE CONTAINERS api or command against the containers to redefine their storage.

Action: To determine the list of containers for each tablespace being restored, use the TABLESPACE CONTAINER QUERY api or the LIST TABLESPACE CONTAINERS command. To specify an updated list for each tablespace, use the SET TABLESPACE CONTAINERS api or command. This api or command lets you specify whether this should be the initial list of containers (i.e. a subsequent rollforward will redo any "add container" operations described in the database log) or the final list (rollforward will not redo "add container" operations).

It is also possible that the container(s) are read-only, in which case the only action required before continuing with the restore is to give read/write access to the container.

SQL1279W

Some invalid indexes may not have been recreated.

Cause: The index recreation time for the database is during database restart. Errors were detected during restart which prevented some invalid indexes from being recreated. Details can be found in the system error log and/or the database manager error log.

Database restart was successful.

Action: Examine the error log to determine why the index(es) could not be recreated and correct the problem. The invalid indexes for a particular table will be recreated when the index is first accessed, or, all invalid indexes for a database will be recreated when the database is restarted.

SQL1281N

The connection to database "<database-alias>" has been severed because the pipe "<pipe-name>" has been broken.

Cause: The connection was lost because the DB2 server broke the pipe. The current transaction has been rolled back.

Action: Resubmit the current command. If the error continues, contact your technical service representative.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE or message number
- SQLCA contents, if possible
- Trace file, if possible.

sqlcode: -1281

sqlstate: 40504

SQL1282N

The attempt to connect to database "<database-alias>" failed because all the pipe instances on "<pipe-name>" are busy.

Cause: Attempts to connect to the named pipe failed because the connection was refused by DB2. There is a limit to the number of connections that are allowed on the named pipe.

Action: Increase the connection limit on the DB2 server, or terminate some of the applications using named pipe to free up connection resources.

sqlcode: -1282

sqlstate: 08001

SQL1283N · SQL1290N

SQL1283N

The attempt to connect to database "<database-alias>" failed because the pipe "<pipe-name>" is in use by another process.

Cause: The name of the named pipe is already in used by another process. Named pipe support did not start.

Action: Choose a different name by setting the environment variable DB2PIPENAME, or have the other programs which use named pipe use a different pipe name.

SQL1284N

The attempt to connect to database "<database-alias>" failed because the pipe "<pipe-name>" cannot be found.

Cause: The server has not started its named pipe support, or the server is using a different name for the named pipe.

Action: Start the database manager and ensure the named pipe support starts. If the named pipe support has been started, then ensure the name of the named pipe is the same between the client and server by setting the environment variable DB2PIPENAME to the same value.

sqlcode: -1284

sqlstate: 08001

SQL1285N

The attempt to connect to database "<database-alias>" failed because the pipe "<pipe-name>" is invalid.

Cause: The alternate pipe name set by the environment variable DB2PIPENAME is invalid.

Action: The value of the environment variable DB2PIPENAME must be a valid pipe name. The pipe name must be no longer than eight characters, and has the same syntactical restrictions as a normal file name.

sqlcode: -1285

sqlstate: 08001

SQL1286N

The connection to database "<database-alias>" has been severed because the operating system ran out of resources for pipe "<pipe-name>".

Cause: Named pipe failed because the operating system ran out of resources (swapping space, disk space, file handles). The current transaction has been rolled back.

Action: Free up the system resources and try again.

sqlcode: -1286

sqlstate: 40504

SQL1287N

ATTACH to instance "<instance>" failed because the named pipe "<pipe>" cannot be found.

Cause: The server has not started its named pipe support, or the instance name is incorrect.

Action: Ensure that the database manager is started at the server and that the named pipe support is started. Ensure that the instance name is correct.

SQL1290N

The value of either the DFT_CLIENT_COMM database manager configuration parameter or the DB2CLIENTCOMM environment variable is not valid.

Cause: Either an incorrect value was specified or the protocol(s) specified are not supported by the target database. The acceptable values are TCPIP and APPC. If both values are specified, then they must be delimited by a comma.

Note that this message may be returned from an intermediate node involved in your connection. For example, if you are trying to connect to a DRDA server via a DDCS gateway and your client workstation does not use global directory services, this message may be returned from the DDCS gateway.

Action: Correct the value and try again.

sqlcode: -1290

sqlstate: 08001

SQL1291N

A directory services error has been detected. Service: "<directory-services-type>", API: "<API>", function: "<function>", error code: "<rc>".

Cause: An error has been detected by the directory services subsystem. Refer to the token values for details. The following is the explanation of the token values:

"<directory-services-type>"

> The type of the directory services being used. Valid tokens include:
>
> • DCE

"<API>"

> The application programming interface used to access the above directory services. Valid tokens include:
>
> • XDS/XOM

"<function>"

> The name of the directory services subsystem function that returned the error code.

"<rc>"

> The error code returned from the above function. The meaning of the value is specific to the API being used.
>
> For XDS functions such as ds_read, the return code values can be found in the /usr/include/xds.h file.
>
> For XOM functions such as om_get, the return code values can be found in the /usr/include/xom.h file.

Note that this message may be returned from an intermediate node involved in your connection. For example, if you are trying to connect to a DRDA server via a DDCS gateway and your client workstation does not use global directory services, this message may be returned from the DDCS gateway.

Action: Ensure that:

• The product providing the directory services is properly installed and operational.

• You are logged in to the directory services with the appropriate authorization to access the directory entries, if login is required by the directory services provider, for example, DCE.

If the problem persists, consult with your system administrator and/or database administrator to determine the cause of the problem using the set of tokens provided.

sqlcode: -1291

sqlstate: 08001

SQL1292N

The global name of the database or database manager instance is invalid.

Cause: The global name of the database or database manager instance cannot be NULL and cannot be longer than 255 characters. It must start with either "/.../" or "/.:/".

Note that this message may be returned from an intermediate node involved in your connection. For

SQL1294N SQL1296N

30 Entry not a routing information object

31 Target database information not found in the routing information object

32 Insufficient information for the target database in the routing information object

33 No appropriate gateway found in the routing information object

34 Invalid flag for authentication at gateway

35 Database locator object name for the gateway invalid

36 Database name in the target database information attribute of the routing information object is either not found or too long

37 Database protocol in the target database information attribute of the routing information object is either not found or too long

Note that you may also see this message if the DCE subsystem is not operational, or if you do not have sufficient privilege to read the DCE directory entry.

Note that this message may be returned from an intermediate node involved in your connection. For example, if you are trying to connect to a DRDA server via a DDCS gateway and your client workstation does not use global directory services, this message may be returned from the DDCS gateway.

Action: Make sure the DCE subsystem is operational and that you have the appropriate privilege to read the directory entry. If the problem persists, inform the database administrator to correct the error in the directory entry. Refer to the *Administration Guide* for the format of these directory objects.

sqlcode: -1293

sqlstate: 08001

┌─ **SQL1294N** ──────────────────────────

The directory path name being used for global directory access is not specified or not valid.

└──

Cause: In order to use global directory services, a directory path name must be specified either in the *dir_path_name* database manager configuration parameter, or in the DB2DIRPATHNAME environment variable. You either did not specify it, or the name you specified is not valid.

Note that this message may be returned from an intermediate node involved in your connection. For example, if you are trying to connect to a DRDA server via a DDCS gateway and your client workstation does not use global directory services, this message may be returned from the DDCS gateway.

Action: Consult your database administrator for the correct name to use, specify it, and try again.

sqlcode: -1294

sqlstate: 08001

┌─ **SQL1295N** ──────────────────────────

The routing information object name being used for global directory access is not specified or not valid.

└──

Cause: In order to use global directory services to access a remote database with a database protocol that is not native to this client, the name of a routing information object must be specified either in the *route_obj_name* database manager configuration parameter, or in the DB2ROUTE environment variable. You either did not specify it, or the name you specified is not valid.

Note that this message may be returned from an intermediate node involved in your connection. For example, if you are trying to connect to a DRDA server via a DDCS gateway and your client workstation does not use global directory services, this message may be returned from the DDCS gateway.

Action: Consult with your database administrator for the correct object name to use, specify it and try again.

sqlcode: -1295

sqlstate: 08001

┌─ **SQL1296N** ──────────────────────────

Valid values must be specified for the DIR_PATH_NAME and DIR_OBJ_NAME database manager configuration parameters if the DIR_TYPE parameter has a value other than NONE.

└──

Cause: These three parameters are interrelated. If the DIR_TYPE value is NONE, then the other two are ignored. If the DIR_TYPE value is not NONE, then the other two must both contain valid values. The following rules apply when the DIR_TYPE value is not NONE:

1. The DIR_PATH_NAME and DIR_OBJ_NAME values cannot be NULL (or blanks).

2. If the DIR_TYPE value is DCE, then the DIR_PATH_NAME value must start with either of the special DCE strings enclosed in the following quotes: "/.../" or "/.:/".

Action: If you want to change the DIR_TYPE value, make sure the DIR_PATH_NAME and DIR_OBJ_NAME

parameters have been specified with the valid values first. If you want to blank out either the DIR_PATH_NAME or the DIR_OBJ_NAME parameter, make sure you set DIR_TYPE to NONE first.

SQL1297N

This command is currently not supported on this platform.

Cause: The function requested using this command is not supported on this platform.

Action: Do not use this command.

SQL1310N

The Database Connection Services directory service failed while attempting to access the Database Connection Services directory.

Cause: The Database Connection Services Directory Service failed because of file errors while trying to access the Database Connection Services directory file.

The function cannot be processed.

Action: Rerun the function after doing one of the following:

- If you are adding an entry to the Database Connection Services directory, ensure that there is sufficient space for the directory file to become larger.
- Ensure that no other concurrently running program is trying to access the file.
- Verify that the directory file has not been damaged. If it is unrecoverable, you must erase it and recreate it or restore it from a backed-up version.

SQL1311N

The Database Connection Services directory cannot be found.

Cause: The directory cannot be found. The directory may have been deleted.

The function cannot be processed.

Action: Use the CATALOG DCS DATABASE command to add an entry to the Database Connection Services directory, or restore the directory from a backed-up version.

SQL1312W

The Database Connection Services directory is empty.

Cause: An attempt was made to read the contents of the Database Connection Services directory, but no entries existed.

Processing continues, but subsequent commands expecting entries to be available cannot be processed.

Action: Use the Catalog DCS Database command to add entries to the directory or restore a backed-up version that contains entries.

SQL1313N

The Database Connection Services directory is full.

Cause: The entry cannot be added to the Database Connection Services directory because the directory has already reached its maximum size.

The function cannot be processed.

Action: Delete an entry or entries from the directory before adding any more entries.

SQL1314N

The address of the Database Connection Services directory entry parameter is not valid.

Cause: The application program has used an address for this parameter that is not valid. Either the address points to an unallocated buffer or the buffer is not large enough to contain the required input.

The function cannot be processed.

Action: Ensure that the application program has allocated the required buffer area and resubmit the command.

SQL1315N

The local database name is not valid.

Cause: A local database name was specified that contains characters that are not valid. All characters must be from the database manager base character set.

The function cannot be processed.

Action: Ensure that the characters used in the local database name are from the database manager base character set and resubmit the command.

SQL1316N   SQL1321N

SQL1316N

No entry was found in the Database Connection Services directory for the specified local database name.

Cause: The Database Connection Services directory service failed because no entry could be found in the Database Connection Services directory that corresponds to the input local database name.

The function cannot be processed.

Action: Ensure that the local database name is correct and resubmit the command.

SQL1317N

The local database name already exists in the Database Connection Services directory.

Cause: The entry could not be added to the directory because an entry for that local database name already exists in the directory.

The function cannot be processed.

Action: Specify a unique local database name or delete the existing entry and add the new entry.

SQL1318N

The length of element "<name>" in the input structure for parameter 1 is not valid.

Cause: The length values in the Database Connection Services directory entry structure should not be less than zero nor greater than the maximum length for the element it represents.

The function cannot be processed.

Action: If an element of the directory entry structure is specified, the associated length value should represent the number of bytes in the element. Otherwise, the length value should be zero. Ensure that all the directory entry structure elements have the required specifications and lengths for the command and resubmit the command.

SQL1319N

The Database Connection Services directory entries have not been collected.

Cause: A request to copy all the directory entries was received, but either a prior request to collect the entries was not received or the prior request to collect the entries failed.

The function cannot be processed.

Action: Issue a request to open the directory and collect the entries. Then resubmit this command.

SQL1320N

The Database Connection Services directory cannot be accessed at this time.

Cause: A request to access the Database Connection Services directory failed. Access to the Database Connection Services directory is dependent on the type of access requested and the current activity in the directory. If the request is for the intention of updating the directory, then the directory must not be active. If the request is to read the directory, then access is allowed if the directory is not being updated.

The function cannot be processed.

Action: Wait for the current activity to complete and resubmit the command.

SQL1321N

The structure ID specified in the directory entry structure is not valid.

Cause: The structure ID received in the directory entry structure does not represent a recognized value.

The function cannot be processed.

Action: Verify the structure ID passed in the directory entry structure is a valid value and resubmit the command.

SQL1325N

The remote database environment does not support the command or one of the command options.

Cause: An attempt has been made to issue a DB2 workstation database specific command or command option against a host database through DDCS or DataJoiner. The following commands generate this error when issued against a DB2 for MVS*, DB2 for OS/400* or SQL/DS* database:

- OPSTAT (Collect Operational Status)
- DARI (Database Application Remote Interface)
- GETAA (Get Administrative Authorizations)
- GETTA (Get Table Authorizations)
- PREREORG (Prepare to Reorganize Table)
- REORG (Call Reorganize Function)
- RQSVPT/ENSVPT/RLBSVPT (Subtransaction Requests)
- RUNSTATS (Run Statistics).
- COMPOUND SQL ATOMIC STATIC (Atomic Compound SQL)
- ACTIVATE DATABASE
- DEACTIVATE DATABASE

As well, the following commands generate this error with incorrect options:

- IMPORT (Import table) The filetype must be IXF, commitcount must be 0, and the first word in the tcolstrg must be INSERT.
- EXPORT (Export table) The filetype must be IXF.

The command cannot be processed.

Action: Do not attempt to issue this command against a host database through DDCS or DataJoiner.

SQL1326N

The file or directory "<name>" cannot be accessed.

Cause: The file or directory "<name>" cannot be accessed because either the file permissions are incorrect, or the file path is incorrect.

Action: Verify that any path or filename specified in the command is valid and that you have the proper permissions to access the path or filename. Correct the problem and resubmit the command. If the problem persists, contact your System Administrator.

SQL1327N

An implicit connection attempt failed. "<database-name>" is not a valid database name.

Cause: An attempt to perform an implicit connection failed. The syntax of the database alias name specified by the DB2DBDFT environment variable is not valid. The database name must contain 1 to 8 bytes and all the characters must be from the database manager base character set.

The command cannot be processed.

Action: Correct the database alias name specified by the DB2DBDFT environment variable and resubmit your command. If using the command line processor, then a "db2 terminate" should be issued prior to resubmitting the command. If you do not want an implicit connection to be performed, remove the DB2DBDFT environment variable.

sqlcode: -1327

sqlstate: 2E000

SQL1328N

An implicit connection attempt failed. The database alias name or database name "<name>" was not found in the local database directory.

Cause: An attempt to perform an implicit connection failed. The database name specified by the DB2DBDFT environment variable does not name an existing database. The database could not be found in the database directory.

The command cannot be processed.

Note that a CONNECT RESET request issued within a distributed unit of work will attempt an implicit connect to the default database. This may be the reason for this error.

Action:

- Correct the database alias name specified by the DB2DBDFT environment variable and resubmit the command.

- If the intended action is to remove a connection while operating in a distributed unit of work environment, consider replacing the CONNECT RESET statement with a DISCONNECT or RELEASE statement.

SQL1329N SQL1333N

- If using the command line processor, then a "db2 terminate" should be issued prior to resubmitting the command.
- If you do not want an implicit connection to be performed, remove the DB2DBDFT environment variable.

sqlcode: -1328

sqlstate: 42705

SQL1329N

The resolved path specified on the command is too long.

Cause: The resolved path specified on the command exceeds the maximum length supported by the database manager. The resolved path must be no longer than 215 characters in length. During a Create Database, Catalog Database, Open Database Directory for scan, and change database comment commands, the database manager instance name is appended to the end of the specified path.

Action: Check that the fully resolved absolute or relative path name, including the database manager instance name, does not exceed 215 characters in length. Correct the path and resubmit the command.

SQL1330N

The symbolic destination name "<name>" is not valid.

Cause: The symbolic destination name in the CPIC protocol structure of the Catalog Node command is either not specified or longer than the allowable length. The name must be 1 to 8 bytes in length.

Action: Verify that the symbolic destination name is specified and that it is not longer than 8 bytes in length. Resubmit the command with a valid symbolic destination name.

SQL1331N

The CPIC security type "<type>" is not valid.

Cause: The CPIC security type specified in the CPIC protocol structure of the Catalog Node command is invalid. The security type specifies the security information that will be included when a database client allocates a conversation with a partner LU, as specified by the LU 6.2 architecture. Correct values for security type are:

- SQL_CPIC_SECURITY_NONE
  - No access security information will be included.

    Note: This is not supported when using DDCS or DataJoiner.

- SQL_CPIC_SECURITY_SAME
  - A userid will be included together with an indicator specifying that the userid is already verified. This is not supported when authentication type DCS is used with DDCS or DataJoiner

- SQL_CPIC_SECURITY_PROGRAM
  - Both a userid and a password will be included. This is not supported when authentication type CLIENT is used with DDCS.

Action: Set the security type to one of the above and resubmit the command.

SQL1332N

The Host Name "<name>" is not valid.

Cause: The host name in the TCP/IP protocol structure of the Catalog Node command is either not specified or longer than the allowable length. The name must be 1 to 255 characters in length and cannot be all blanks.

Action: Verify that the host name is specified and that it is not longer than 255 characters in length. Resubmit the command with a valid host name.

SQL1333N

The Service Name "<name>" is not valid.

Cause: The service name is the TCP/IP protocol structure of the Catalog Node command is either not specified or longer than the allowable length. The name must be 1 to 14 characters in length and cannot be all blanks.

Action: Verify that the service name is specified and that it is not longer than 14 characters in length. Resubmit the command with a valid service name.

SQL1334N

The database server cannot be used to route a remote request to a second database server in this configuration.

Cause: An attempt was made to route a request through a database server node using an unsupported combination of client and target database server. Either a client or target database prior to release version 2 was used or an attempt was made to route the request from a DRDA client to a DRDA target database. The request must be routed directly from the client to the node on which the target database is running.

Action: Uncatalog the database at the client machine and then catalog the database specifying the node on which the database actually resides. Ensure that the node is also cataloged.

SQL1335N

The application requestor name is not valid

Cause: An application requestor name was specified that contains characters that are not valid. All characters must be from the database manager base character set.

Action: Ensure that the characters used in the application requestor name are from the database manager base character set and resubmit the command.

SQL1336N

The remote host "<hostname>" was not found.

Cause: The system cannot resolve the address of the remote host. Possible causes are:

- An incorrect hostname value was specified when the TCP/IP node was cataloged.

- The correct hostname was specified, but was not defined in any TCP/IP name server accessible to your client node, nor in the client's hosts file.

- The TCP/IP name server where the hostname was defined was not available at the time you tried to connect.

- TCP/IP is not running.

Action: Make sure that TCP/IP is running and that the hostname specified when cataloging the TCP/IP node is a correct name, and that it is defined in an accessible name server or in the local hosts file.

DataJoiner users: verify that the remote host is cataloged correctly in the SYSCAT.SERVERS view.

SQL1337N

The service "<service-name>" was not found.

Cause: The system cannot resolve the port number associated with the service-name. Possible causes are:

- An incorrect service-name value was specified when the TCP/IP node was cataloged.

- The correct service-name was specified, but was not defined in the client's services file.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Make sure that the service-name specified when cataloging the TCP/IP node is a correct name, and that it is defined in the local services file.

DataJoiner users must also ensure that the name is defined in the services file at the data source.

SQL1338N

The symbolic destination name "<symbolic-destination-name>" was not found.

Cause: The system cannot find the Side Information associated with the symbolic-destination-name you specified. Possible causes are:

- An incorrect symbolic-destination-name value was specified when the CPIC NODE was cataloged.

- The symbolic destination name and its associated side information have not been defined to the SNA communications subsystem.

Action: Make sure that the symbolic-destination-name specified when cataloging the CPIC node is a correct name, and that it has been defined to the local SNA communication subsystem.

SQL1339N

"<n>" SQL error(s) were detected in the execution of a Not Atomic Compound SQL statement, and have the following identification: "<error1>" "<error2>" "<error3>" "<error4>" "<error5>" "<error6>" "<error7>".

Cause: One or more SQL substatements within a Compound SQL statement resulted in SQL errors (negative return codes).

SQL1340N SQL1350N

Action: Examine the error identification information provided. There will be <n> <errorX> tokens filled in, up to a maximum of 7. Each <errorX> represents an SQL statement error. These errors are listed in the order that they were encountered. If the message text has not been formatted, this information can be found in the second and subsequent tokens of the SQLERRMC field (tokens are separated by a one byte hexadecimal value 0xFF).

Each <errorX> is of the format PPPSSSSS where:

PPP	PPP represents the position of the statement that caused the error within the Compound SQL block, and is left-justified. For example, if the first statement failed, this field would contain the number one ("1 ").
SSSSS	SQLSTATE of the statement that failed.

More information can be found by examining the SQLCA itself. The third SQLERRD field contains the number of rows affected by the compound SQL statement, the fourth SQLERRD field contains the position of the last statement that succeeded, the 5th SQLERRD field contains the number of rows affected by referential integrity when DB2 Client/Server and SQL/DS databases are accessed, and the 6th SQLERRD field contains the number of statements that failed (returned negative SQLCODES).

sqlcode: +1339

sqlstate: 56091

SQL1340N

 The file server "<fileserver>" was not found.

Cause: The system cannot find the specified file server on the network. Possible causes are:

- An incorrect *fileserver* name was specified when the IPX/SPX node was cataloged.
- The correct *fileserver* name was specified, but the file server was not available at the time the connect was tried.

Action: Make sure that the *fileserver* name specified when cataloging the IPX/SPX node is correct, and that the file server is available on the network.

SQL1341N

 A workstation name must be specified in the client database manager configuration file.

Cause: A workstation name has not been specified in the client database manager configuration file. A workstation name must be specified when using NetBIOS to communicate with the server.

Action: Specify a workstation name in the client database manager configuration file.

SQL1342N

 The File Server name "<name>" is missing or invalid."

Cause: The File Server name specified in the command/API is missing or invalid.

Action: Verify that a File Server name was specified, that the name does not contain invalid characters, and is not longer than 48 characters in length. Resubmit the command/API with a valid File Server name.

SQL1343N

 The Object Name "<name>" is missing or invalid.

Cause: The Object Name specified in the command/API is missing or invalid.

Action: Verify that the Object Name was specified, that the name does not contain invalid characters, and is not longer then 48 characters in length. Resubmit the command/API with a valid Object Name.

SQL1350N

 The application is not in the correct state to process this request. Reason code="<rc>".

Cause: Corresponding to "<rc>":

01	The application is currently processing SQL and cannot process the utility command requested.
02	A Backup request is in progress. A warning was received from the initial utility call indicating that further requests are required before Backup can complete.
03	A Restore request is in progress. A warning was received from the initial utility call indicating that further requests are required before Restore can complete.

04   A Rollforward request is in progress. A warning was received from the initial utility call indicating that further requests are required before Rollforward can complete.
05   A Load request is in progress. A warning was received from the initial utility call indicating that further requests are required before Load can complete.
06   An Unload request is in progress. A warning was received from the initial utility call indicating that further requests are required before Unload can complete.
07   DataJoiner users: the application cannot process this command after executing SQL statements.

Action: Corresponding to "<rc>":

01   Complete the unit of work (using either COMMIT or ROLLBACK) before reissuing this command.
02-06 Issue the required call(s) to complete the utility in progress, then reissue this command.
07   DataJoiner users: the application must issue the commands after establishing a connection to the database manager but prior to any other SQL statements.

--- **SQL1360N** ---

Current processing cannot be interrupted.

Cause: The user attempted to interrupt a process which is not interruptible.

Action: Do not interrupt the current process.

--- **SQL1361W** ---

Execution time has exceeded timeout value. Do you wish to interrupt it?

Cause: If a command takes longer than a pre-defined timeout period, a dialog box pops up asking if the (Windows client) user wants to interrupt it or not.

Note that this message is applicable in Windows environments only, and will only be seen in a dialog pop up box.

Action: There are three choices: YES - interrupt now; NO - continue, don't prompt me; CANCEL - continue, same timeout.

--- **SQL1370N** ---

An attempt to quiesce instance or database "<name1>" failed because instance or database "<name2>" is already quiesced by user "<username>". Quiesce type: "<type>".

Cause: An attempt was made to quiesce an instance or database which would have caused an overlapping quiesce, such as quiescing an instance while a database is already quiesced by another user.

Quiesce type "<type>" refers to the instance or database already quiesced and is a '1' for an instance and a '2' for a database.

Action: Contact the user who currently has the instance or database quiesced to determine when DB2 will no longer be quiesced, and retry the request at that time.

--- **SQL1371N** ---

Instance or database "<name>" is already quiesced by user "<username>". Quiesce type: "<type>".

Cause: An attempt was made to quiesce an instance or database which is already quiesced.

Quiesce type "<type>" refers to the instance or database already quiesced and is a '1' for an instance and a '2' for a database.

Action: Contact the user who currently has the instance or database quiesced to determine when DB2 will no longer be quiesced, and retry the request at that time.

--- **SQL1372N** ---

Cannot perform a quiesce while in a transaction.

Cause: An attempt was made to quiesce a database or instance while the user issuing the quiesce has uncompleted units of work. Quiesce cannot be issued while in this state.

Action: Complete the unit of work (COMMIT or ROLLBACK) and retry the request.

SQL1373N   SQL1393C

Cause: QUIESCE RESET failed because the instance or database is not quiesced.

Action: Ensure that the quiesce reset is being issued for the correct instance or database and try again.

Cause: The instance or database is quiesced, but you are not the quiescer.

Action: Ensure that the quiesce reset is being issued for the correct instance or database.

Cause: "<parm-code>" indicates the parameter in error:

1   scope
2   options

The value may be out of range or invalid.

Action: Check the syntax of the api, correct the parameter and try again.

Cause: The environment variable DB2INSTANCE is not defined or is not set to a valid instance owner.

Action: Set the DB2INSTANCE environment variable to the name of the instance to use. If you do not know the name of the instance to use or how to set the DB2INSTANCE environment variable to the instance name, refer to the *Administration Guide*.

DataJoiner users: refer to the *IBM DataJoiner Planning, Installation, and Configuration Guide* for DB2INSTANCE information.

Ensure your PATH environment variable contains the sqllib/adm path in the home directory of the instance you want to use (for example, /u/instance/sqllib/adm, where /u/instance is the home directory of the instance owner on a UNIX system).

Cause: The request failed because the database is in use by another instance of the database manager (a database may only be used by one instance.) This may be caused by attempting to access a database located on a mounted file system accessible to another instance located on another machine.

This can also occur if you have an open connection (via SNA) to a database and the database manager was brought down abnormally.

Action:

- Validate that you are using the correct database and ensure no other instance uses this database.

- If the database manager was brought down abnormally, and you have a command line processor connection to it, perform a db2 terminate to close the offending open connection before attempting to connect again.

sqlcode: -1391

sqlstate: 51023

Cause: Only 1 instance of prep, bind, import, or export can be running at any one time within WINDOWS.

Action: Within WINDOWS, do not attempt to start more than 1 application that uses prep, bind, import, or export.

Cause: The environment variable DB2PATH is not defined or is not set to a valid directory path.

Action: Set the DB2PATH environment variable to the directory where the database manager is installed.

SQL1400N

Authentication is not supported.

Cause: The authentication type supplied is not one of the supported types.

The command can not be processed.

Action: The user should reissue the command with a valid parameter value.

See *API Reference* for a list of supported authentication types.

SQL1401N

Authentication types do not match.

Cause: The user is attempting to connect to a remote database which is cataloged on the local node with a different authentication type from the remote node.

DataJoiner users: this message can also appear when:

- A data source is identified in SYSCAT.SERVER_OPTIONS with SETTING='N' for OPTION ='PASSWORD' and the data source is not running in trusted client mode (that is, the data source expects a password).
- The data source is identified in SYSCAT.SERVER_OPTIONS with SETTING='Y for OPTION ='PASSWORD" and the data source is running in trusted client mode (that is, the data source does not expect a password).
- There is no server option specified for OPTION='PASSWORD' in SYSCAT.SERVER_OPTIONS and the system default value for PASSWORD violates the data source password requirement.

Action: The command can not be processed.

The user should recatalog the database alias on the client node with the same authentication type as the remote database. Resubmit the command.

DataJoiner users:

- If the problem is that the data source requires a password but SYSCAT.SERVER_OPTIONS contains SETTING='N' for OPTION='PASSWORD' for that server, alter SYSCAT.SERVER_OPTIONS to reflect the true data source password requirement using the ALTER SERVER OPTION sql statement.
- If the problem is that the data source does not require a password but SYSCAT.SERVER_OPTIONS contains SETTING='Y' for OPTION='PASSWORD' for that

server, alter SYSCAT.SERVER_OPTIONS to reflect the true data source password requirement using the ALTER SERVER OPTION sql statement.
- If the server option for OPTION='PASSWORD' is not set in SYSCAT.SERVER_OPTIONS, create an entry using the CREATE SERVER OPTION sql statement to reflect the true data source password requirement.

sqlcode: -1401

sqlstate: 08001

SQL1402N

Unable to authenticate user due to unexpected system error.

Cause: Call your system administrator. On unix-based systems, the file *db2ckpw* may not have the correct permission bits set. On Windows NT, the DB2 security service may not have been started.

The command can not be processed.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: On unix-based systems, have the system administrator set the correct access permissions for *db2ckpw*. On Windows NT, have the system administrator verify that the DB2 security service is installed and started.

SQL1403N

The username and/or password supplied is incorrect.

Cause: The username and/or password supplied is incorrect or not a valid username/password combination or the database to which you are attempting to connect to has an authentication type of SERVER and no username and password was supplied on the CONNECT TO statement.

If DDCS is being used, the problem may be that the DCS directory entry for the host connection could not be found.

If a connection is being made from an OS/2 client to a UNIX-based server with authentication server, and the userid and password are being picked up from UPM, then the userid on the server must be defined in lowercase and have an uppercase password.

The command can not be processed.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action: Supply the correct username and password combination.

DataJoiner users: ensure that the entry in SYSCAT.REMOTEUSERS contains the correct username and password for the data source being accessed.

sqlcode: -1403

sqlstate: 08004

SQL1404N

Password expired.

Cause: Your password has expired.

Action: Change your password, then retry the request using the new password.

sqlcode: -1404

sqlstate: 08004

SQL1405N

Unable to communicate with local DB2 authentication server.

Cause: The application failed to authenticate due to an error while communicating to the local DB2 authentication server.

Action: Ensure that the DB2 authentication server is started by entering the following command from an OS/2 window:

    detach db2upm

SQL1415N

The statement was compiled for diagnostic purposes only and has not been executed.

Cause: The statement was processed through parts of the system to collect diagnostic information using service features. The necessary steps to allow further processing of the statement have not been completed.

Action: This error is returned to prevent further processing by the system of statements prepared using service features and is expected.

SQL1419N

Package "<pkgname>" cannot be executed.

Cause: The package "<pkgname>" can not be executed from the current EUC application (application running in an Extended Unix Code codepage) because it was bound on a previous database manager level which lacks EUC support.

Action: To execute the package from the EUC application, explicitly rebind the named package using either the REBIND or the BIND command. Otherwise, execute the package from a non-EUC application.

sqlcode: -1419

sqlstate: 51028

SQL1420N

Too many concatenation operators.

Cause: The database manager has reached an internal limit while evaluating an expression, of long or large object string result type, that contains concatenation operators.

Action: Reduce the number of concatenations in the expression and try again.

sqlcode: -1420

sqlstate: 54001

SQL1421N

MBCS conversion error occurred when converting host variable or sqlvar "<number>" to or from wchar_t format. Reason code "<rc>".

Cause: A C/C++ application with embedded SQL statements was precompiled with the WCHARTYPE CONVERT option. At runtime, the application received an error which occurred during conversion in either wcstombs(), for input host variables, or mbstowcs(), for output host variables. The host variable or sqlvar number indicates which data item experienced the problem. Valid reason codes are:

1   the problem occurred with input data
2   the problem occurred with output data

Action: If application data is already in MBCS format, re-precompile the application with WCHARTYPE NOCONVERT and re-build. If application data is intended to be in wchar_t format, then input data failing in wcstombs() may be corrupt. Correct the data and re-execute the application.

sqlcode: -1421

sqlstate: 22504

SQL1422N

The size of the container is invalid.

Cause: One of the containers to be used in a database managed table space is either too large or too small. A container must be at least extentsize + 1 pages in length. The maximum size of a container is operating system dependent. The most common system limit is 2 gigabytes (524288 4K pages).

Action: Check the diagnostic log for details. Then correct the SQL statement.

sqlcode: -1422

sqlstate: 54039

SQL1423N

The query includes a column with a large object data type.

Cause: The query includes a column with a data type of BLOB, CLOB or DBCLOB. In general, such data types cannot be handled from clients available prior to Version 2.1.

The error occurred for a condition that corresponds to warning SQLCODE +238. See this message for details that explain the condition. The client level receiving this message cannot process BLOB data types. It may be able to handle CLOB and DBCLOB data types by using the SUBSTR function or by setting the data type in the SQLDA to one of the character data types available in Version 1 if the LOB column is not larger than the supported length of the character data type.

Action: Change the query to exclude any column that has a data type of BLOB, CLOB or DBCLOB. This is the only possible action if the query includes a column of type BLOB. If the column (say C1) is a CLOB, you could use CAST(C1 AS LONG VARCHAR) to get the first 32700 characters. Similarly for DBCLOB column (DC1) use CAST(DC1 AS LONG VARGRAPHIC) to get the first 16350 characters. If the application code can be changed, code can be added to change the SQLDA to use LONG VARCHAR or LONG VARGRAPHIC for CLOB and DBCLOB.

sqlcode: -1423

sqlstate: 56093

SQL1424N

Too many references to transition variables and transition table columns or the row length for these references is too long. Reason code=“<rc>”.

Cause: The trigger includes a REFERENCING clause that identifies one or more transition tables and transition variables. The triggered action of the trigger contains references to transition table columns or transition variables with one of the following conditions identified by the reason code:

1 references total more than the limit of the number of columns in a table

2 sum of the lengths of the references exceeds the maximum length of a row in a table.

Action: Reduce the number of references to transition variables and transition table columns in the trigger action of the trigger so that the length is reduced or the total number of such references is less than the maximum number of columns in a table.

sqlcode: -1424

sqlstate: 54040

SQL1425N

A password has been supplied without a userid.

Cause: Any command/API that accepts a userid and password will not accept a password without a userid.

Action: Resubmit the command/API and supply a userid if you are also supplying a password.

SQL1426N

A default instance cannot be determined.

Cause: If an explicit 'attach to instance' has not been performed, instance commands try to form an implicit attachment to a default instance. The default instance is determined from the DB2INSTDFT and DB2INSTANCE environment variables. If neither of these are set, then no implicit attachment can be formed.

Action: Set one of these environment variables to a valid instance name and resubmit the command.

SQL1427N

An instance attachment does not exist.

Cause: The application is not attached to an instance. The attempted command/API cannot be executed unless there is an existing instance attachment.

Action: If the error occurred when detaching from an instance, continue processing. If the error occurred during the execution of any other command, attach to an instance and reissue the failed command.

SQL1428N

The application is already attached to "<nodename1>" while the command issued requires an attachment to "<nodename2>" for successful execution.

Cause: In order to successfully process the command, an attachment to a node other than that which currently exists is required. The application must either: 1) have no attachment when the command is issued, or 2) already be attached to the node required by the command.

Action: Ensure that the application has no attachment before issuing the command or that the attachment that exists is to the correct node.

SQL1429N

A node directory entry cannot be created in which the node name matches the value of the DB2INSTANCE environment variable.

Cause: The CATALOG NODE command or API does not allow an entry whose nodename matches the value of the DB2INSTANCE environment variable.

Action: Choose another node name for the node being cataloged and try again.

SQL1430N

The database name "<database>" cannot be found in the system database directory on node "<nodename>".

Cause: When specific database names are provided to the database monitor, those databases must reside either at the node to which you are currently attached or at the local node.

Action: Make sure the databases listed in the request

all reside at the node to which you are attached or at the local node. Reissue the request.

SQL1431N

The relative path "<path>" is not allowed when executing remotely.

Cause: When the application is remote from the server, relative paths are not allowed.

Action: Specify a fully qualified path that is valid at the server and reissue the command.

SQL1432N

A request has been sent to a server using a database protocol which the server does not recognize.

Cause: This error is caused by sending a DB2 request to a server which does not understand the database protocol being used to transmit the request. This situation will occur most frequently when sending a DB2 ATTACH request to a server listed in your node directory which is not a DB2 Version 2 or greater server. This error will also arise if you send an attach request to DB2 for AS/400, DB2 for MVS, or DB2 for VM and VSE servers.

Action: Do not attempt to attach to any of the servers listed above.

SQL1433N

The application is already connected to "<database1>" while the command issued requires a connection to "<database2>" for successful execution.

Cause: In order to successfully process the command, a connection to a database other than that which currently exists is required. The application must either: 1) have no connection when the command is issued, or 2) already be connected to the database required by the command.

Action: Ensure that the application has no connection before issuing the command or that the connection that exists is to the correct database.

SQL1441N

Invalid parameter. Reason code "<code>".

Cause: The following is a list of valid reason codes:

1   A NULL was passed in for the context pointer.
3   The context pointer has been initialized, but not to a valid context area.
4.   Invalid option
5   reserved parameter was not NULL

Action: Ensure that the application context pointer has been properly initialized and that any options used are valid and try again.

SQL1442N

The context is not in use or is not in use by the current thread. Reason code "<code>".

Cause: The call failed because:

1   The context is not in use by any thread (no attach was done)
2   The context is not in use by the current thread.
3   The current thread is not using a context.

Action: For a detach call, ensure that the context is the one in use by the current thread, and that a corresponding attach was done.

For a get current context call, ensure that the thread is currently using a context.

SQL1443N

Thread is already attached to a context.

Cause: The user attempted to attach a context to a thread, and the thread is already using a context.

Action: Detach from the previous context before attaching to a new one.

SQL1444N

The application context cannot be destroyed because it is in use.

Cause: The user attempted to destroy an application context while it is still in use. Either there is a thread attached to the context, or the context has a CONNECT or ATTACH associated with it. A CONNECT RESET or DETACH must be done (if a CONNECT or ATTACH has been done), and all threads must detach from the context before it can be destroyed.

Action: Ensure that all calls to attach to a context have a corresponding detach, all CONNECTS have a corresponding CONNECT RESET, and all ATTACHES have a corresponding DETACH.

SQL1445N

Thread or process does not have a context to use.

Cause: A context type of SQL_CTX_MULTI_MANUAL is in effect, but the current thread or process has not been attached to a context.

Action: Make sure the current thread or process is attached to a context before making any database calls.

SQL1450N

Invalid registration information pointer.

Cause: An invalid registration information pointer was passed to the register/deregister DB2 server command/API.

Action: Verify that a valid pointer was passed to the register/ deregister DB2 server command/API.

SQL1451N

Register/deregister DB2 server must be issued from a server node.

Cause: Register/deregister DB2 server was issued from an invalid node.

Action: Re-issue the register/deregister DB2 server command/API from a server node.

SQL1452N

An invalid registration location was specified.

Cause: An invalid registration location was passed to the register/deregister DB2 server command/API.

Action: Verify that a valid registration location was passed to the register/deregister DB2 server command/API.

SQL1453N

The entry in the database manager configuration file for file server name is missing or invalid.

Cause: The file server name specified in the configuration command/API or in the database manager configuration file is missing or invalid.

Action: Verify that a file server name was specified, that the name does not contain invalid characters, and is not longer than 48 characters in length. Update the file

server name in the database manager configuration file and resubmit the command/API.

SQL1454N

The entry in the database manager configuration file for object name is missing or invalid.

Cause: The object name specified in the configuration command/API or in the database manager configuration file is missing or invalid.

Action: Verify that an object name was specified, that the name does not contain invalid characters, and is not longer than 48 characters in length. Update the object name in the database manager configuration file and resubmit the command/API.

SQL1455N

The entry in the database manager configuration file for IPX socket number is missing or invalid.

Cause: The IPX socket number specified in the configuration command/API or in the database manager configuration file is missing or invalid.

Action: Verify that an IPX socket number was specified, that the number does not contain invalid characters, and is not longer than 4 characters in length. Update the IPX socket number in the database manager configuration file and resubmit the command/API.

SQL1456N

The object name specified in the database manager configuration file already exists on the NetWare file server.

Cause: A duplicate object name was found when trying to register the DB2 server object name on the NetWare file server.

Action: The object name specified in the database manager configuration file is already in use. Change the object name and then re-register the DB2 server.

SQL1457N

Register/deregister failed to login to the NetWare file server specified because a NetWare directory services connection is already established to that file server.

Cause: A bindery login using NWLoginToFileServer cannot be performed if a NetWare directory services connection is already established to the specified file server.

Action: Sever the directory services connection by logging out and detaching from Directory Services, and then reissue register/deregister.

SQL1458N

IPX/SPX has been configured in the database manager configuration file for direct addressing. It is not necessary to register/deregister the DB2 server at/from a NetWare file server.

Cause: Issuing register/deregister is not necessary since the database manager configuration file has been configured for IPX/SPX direct addressing. ie. Fileserver and objectname have been specified as '*'.

Action: Note that since the DB2 server has been configured for direct addressing only, IPX/SPX clients using fileserver addressing will not be able to connect to this server. For the server to support both types of IPX/SPX client addressing, specify a fileserver and objectname in the database manager configuration file.

SQL1460N

The environment variable "<variable>" required for SOCKS server name resolution is not defined or not valid.

Cause: The SOCKS environment variable SOCKS_NS or SOCKS_SERVER is not defined. SOCKS protocol support requires that both these environment variables be defined.

SOCKS_NS This is the IP address of the domain name server where the SOCKS server is defined.

SOCKS_SERVER This is the hostname of the SOCKS server.

Action: Define the missing environment variable and resubmit the command.

SQL1461N

The security option "<security>" is not valid.

Cause: The SECURITY option for a TCP/IP node has a value other than 'SOCKS'. This option is used to enable the TCP/IP node being cataloged to use SOCKS protocol support to traverse a firewall. No value other than 'SOCKS' is permitted.

Action: Verify that SOCKS protocol support is required. If so, re-catalog the node with SECURITY SOCKS. If not, re-catalog the node, but omit the SECURITY option.

SQL1490W

The database "<name>" has already been activated.

Cause: The specified database has already been activated.

Action: No response is required.

SQL1491N

The database "<name>" was not deactivated because the database is still being used.

Cause: The database cannot be deactivated if there are any applications connected to the specified database.

Action: Ensure that all applications have done a CONNECT RESET, then try again.

SQL1492N

The database "<name>" was not deactivated because it was not activated.

Cause: The database cannot be deactivated because the specified database was not active.

Action: No action is required.

SQL1493N

The application is already connected to an active database.

Cause: The ACTIVATE DATABASE and DEACTIVATED DATABASE commands cannot proceed because the application is already connected to a database.

Action: Disconnect from the database then re-submit the command.

SQL1512N

ddcstrc could not write to the specified file.

Cause: *ddcstrc* was not able to write the trace information to the filename that it has been instructed to write to.

Action: Ensure that the filename specified is valid for your file system. If a filename is not specified, ensure that you have the necessary permission to write to the file.

SQL1513W

ddcstrc has not been turned off.

Cause: Because of an error condition *ddcstrc* has not been turned off. This was done to ensure that the trace information will not be lost before being safely placed in a file.

Action: Correct the *ddcstrc* error condition reported prior to this error and attempt to turn off the trace again.

SQL1520N

Buffersize must be a numeric value greater than 65536.

Cause: The user specified an invalid buffersize for the ddcstrc command.

Action: Ensure that the buffersize used is numeric and greater than 65536 (64K). Note that the memory to be used will be a multiple of 64K. ddcstrc rounds down the buffersize specified to the nearest multiple of 64K.

SQL1580W

Trailing blanks were truncated while performing conversion from codepage "<source>" to codepage "<target>". The maximum size of the target area was "<max-len>". The source string length was "<source-len>" and its hexadecimal representation was "<string>".

Cause: During the execution of the SQL statement, a code page conversion operation has resulted in a string that is longer than the maximum size of the target object. Processing continued because only blank characters were truncated.

Action: Ensure that the output is as expected and that the truncation has not caused any unexpected consequences.

SQL1590N   SQL1606W

sqlcode: 1580

sqlstate: 01004

SQL1590N

LONG VARCHAR and LONG VARGRAPHIC fields are not permitted in TABLESPACEs which are built on DEVICEs.

Cause: Devices (raw i/o) on HP require i/o to be aligned on 1024 byte boundaries. LONG VARCHAR and LONG VARGRAPHIC fields are handled in 512 byte pieces, and thus can only be used in SYSTEM MANAGED TABLESPACEs or in DATABASE MANAGED TABLESPACEs with only FILE containers.

Action: Alternatives:

- Choose one of the LOB columns types (BLOB,CLOB,DBCLOB) instead of LONG.

- Use a tablespace with the correct attributes.

sqlcode: 1590

sqlstate: 56097

SQL1601N

The Database System Monitor input parameter "<parameter>" is a null pointer.

Cause: The user called one of the Database System Monitor APIs and provided a null pointer instead of a required parameter.

The command can not be processed.

Action: The user should reissue the command with a valid parameter value.

SQL1602N

An object type supplied in the input data structure (sqlma) is not supported.

Cause: An object type specified in the variable data area of the input data structure (sqlma) for Database System Monitor Snapshot API is not supported.

The command can not be processed.

Action: Reissue the command using a valid object type. Refer to the *Application Programming Guide* and *API Reference* for further information about valid object types.

SQL1603N

The parameter "<parameter>" is not specified in the input data structure (sqlma).

Cause: A required parameter is not specified in the input data structure (sqlma) of the Database System Monitor Snapshot or Estimate Buffer Size API.

The command cannot be processed.

Action: The user should reissue the command specifying a valid parameter value.

SQL1604N

The parameter "<parameter>" is not null terminated.

Cause: A null character is expected at the end of the character string parameter.

The command can not be processed.

Action: Add a null character at the end of the character string parameter and reissue the command.

SQL1605W

The database "<db-alias>" is not active.

Cause: The Database System Monitor Reset API was called for a specific database, but the database was not active.

The command completed successfully but no action was taken.

Action: Verify that the database alias is correct and the database has been started.

SQL1606W

The Database System Monitor output buffer is full.

Cause: The Database System Monitor output buffer area is not large enough to accommodate the returned data. Likely causes are intense system activity when the call was made, or in the case of a Database Monitor API call within a user application, the user allocated a buffer too small to contain the returned data.

The command completed successfully and data collected prior to the buffer overflow is returned in the user's buffer.

Action: The user should reissue the command, or in the case of a Database Monitor API call within a user

application, allocate a larger buffer or reduce the amount of information requested.

SQL1607N

There is not enough working memory to execute the requested Database System Monitor function.

Cause: The database manager is out of working memory to process the Database System Monitor command.

The command can not be processed.

Action: Reduce the buffer size in the input parameter and reissue the command.

SQL1608W

Two or more database aliases specified on input refer to the same database.

Cause: A Database System Monitor Snapshot or Estimate Buffer Size API call was issued specifying the same request for two or more database aliases in the sqlma input data structure, and they point to the same database.

Database System Monitor executes successfully, but returns only one copy of the information in the output buffer.

Action: No action is required. However, the user should verify that the database aliases specified on input are correct if information on different databases was expected.

SQL1609N

The database "<db-alias>" is a remote database and cannot be monitored.

Cause: A Database System Monitor API call was issued specifying the database alias of a remote database. Database System Monitor does not support monitoring of remote databases.

The command cannot be processed.

Action: The user should verify that the database alias specified on input is correct, and reissue the command with the correct database alias.

SQL1610N

The Database System Monitor input parameter "<parameter>" is invalid.

Cause: The user called one of the Database System Monitor APIs and specified an invalid value for the specified parameter.

The command can not be processed.

Action: The user should reissue the command with a valid parameter value.

SQL1611W

No data was returned by Database System Monitor.

Cause: None of the monitoring information requested by the users was available at the time the Database System Monitor API call was issued. This can occur when a requested database or application is inactive, or when a monitoring group such as the Table group is turned OFF, and Table information is requested.

Action: The command completed successfully, but no data is returned to the user.

The user should make sure that the databases or applications for which monitoring is desired are active at the time the Database System Monitor API is called, or that the desired monitoring groups are active.

SQL1612N

The specified event monitor target path is invalid.

Cause: The target path specified in the CREATE EVENT MONITOR statement is not a valid path name. The command could not be processed.

Action: Resubmit the statement with the correct event monitor target path.

sqlcode: -1612

sqlstate: 428A3

SQL1613N SQL1617N

SQL1613N

The specified event monitor option is invalid.

Cause: The option specified in the CREATE EVENT MONITOR statement is not valid. Possible causes include:

- The specified MAXFILES, MAXFILESIZE, or BUFFERSIZE is too low.
- MAXFILESIZE is smaller than BUFFERSIZE.
- MAXFILESIZE NONE was specified with MAXFILES not equal to 1.

The command could not be processed.

Action: Resubmit the statement with the corrected event monitor options.

sqlcode: -1613

sqlstate: 428A4

SQL1614N

An I/O error occurred when activating an event monitor. Reason code = "<reason-code>".

Cause: An I/O error was detected when an event monitor was activated. <reason-code> is one of the following:

1   Encountered an unknown event monitor target type.

2   The Event monitor target path was not found.

3   Access to event monitor target path was denied.

4   Event monitor target path is not the name of a pipe.

5   No process has opened the event monitor target pipe for reading.

6   Encountered an unexpected I/O error.

7   The target pipe was not opened in message mode. (This reason code is applicable on OS/2 only.)

8   The target pipe buffer is too small. The inbound pipe buffer must be at least 4096 bytes in size. (This reason code is applicable on OS/2 only.)

Action: Where possible, fix the problem described by the reason code, and resubmit the SET EVENT MONITOR statement.

sqlcode: -1614

sqlstate: 58030

SQL1615W

The specified event monitor is already in the requested state.

Cause: An attempt was made to either activate an already active event monitor, or deactivate an already inactive event monitor. The SET EVENT MONITOR statement was ignored.

Action: No user response is required.

sqlcode: +1615

sqlstate: 01598

SQL1616N

The limit on the maximum number of active event monitors has already been reached.

Cause: A maximum of 32 event monitors can be active simultaneously per database. This limit has already been reached. The specified event monitor cannot be activated.

Action: If possible, deactivate one of the active event monitors and resubmit the SET EVENT MONITOR statement.

sqlcode: -1616

sqlstate: 54030

SQL1617N

The specified event monitor has already reached its MAXFILES and MAXFILESIZE limit.

Cause: The specified event monitor was created with a limit on the amount of data that would be allowed in the event monitor target directory. This limit has already been reached. The specified event monitor cannot be activated.

Action: If possible, delete some of the event monitor data files from the target directory, and resubmit the SET EVENT MONITOR statement.

sqlcode: -1617

sqlstate: 54031

SQL1618N

The target path of the specified event monitor is in use by another event monitor.

Cause: The specified event monitor was created with the same target path as another event monitor. Since this other event monitor is currently active, the specified event monitor cannot be activated.

Action: If possible, deactivate the event monitor that is using the target path and resubmit the SET EVENT MONITOR statement. Alternatively, recreate the required event monitor, specifying a different target path, and resubmit the SET EVENT MONITOR statement.

sqlcode: -1618

sqlstate: 51026

SQL1619N

Cannot DROP an active event monitor.

Cause: The specified event monitor is currently active, and therefore cannot be dropped.

Action: Deactivate the event monitor and resubmit the DROP EVENT MONITOR statement.

sqlcode: -1619

sqlstate: 55034

SQL1621N

The transaction in which the specified event monitor was created has not yet been committed. The event monitor cannot be activated.

Cause: An event monitor cannot be activated until the transaction in which it was created has been committed.

Action: Commit the transaction in which the event monitor was created and then reissue the SET EVENT MONITOR statement.

sqlcode: -1621

sqlstate: 55033

SQL1622N

The STATE value specified in the SET EVENT MONITOR STATE statement is invalid.

Cause: The STATE value specified in the SET EVENT MONITOR STATE statement is not within the range of valid values, or the value is NULL as a result of an indicator variable.

Valid values for the Event Monitor State include:

0   to deactivate the event monitor
1   to activate the event monitor

The statement cannot be executed.

Action: Correct the event monitor state value and/or any indicator variable and reissue the statement.

sqlcode: -1622

sqlstate: 42815

SQL1623N

The sqlmonsz or sqlmonss API was called with too many objects specified in the sqlma input structure.

Cause: The limit imposed on the number of objects that are permitted in the sqlma input structure has been exceeded.

Action: Reduce the number of objects in the sqlma parameter and try the call again.

SQL1624N

All databases referenced by the sqlmonsz or sqlmonss API must be located at the same node.

Cause: The sqlma parameter contained references to databases that reside on different nodes.

Action: Modify the sqlma parameter so that all database objects reference the same node and then try the call again.

SQL1650N   SQL1704N

SQL1650N

The function invoked is no longer supported.

Cause: The user has attempted to call an API that is no longer supported in this version of the database manager.

Action: The function required may be supported by a different API call. Check in the *API Reference* and *Application Programming Guide* to determine if the function is supported through a different API call.

SQL1700N

The reserved schema name "<name>" was found in the database during database migration.

Cause: The database contains one or more database objects which use the schema name "<name>" that is reserved by the current Database Manager.

The command cannot be processed.

Action: Ensure that all the database objects that use the reserved schema name are dropped and recreate the objects using a different schema name. This correction must be made in the release of Database Manager where the database was originally created. Ensure that no reserved schema names are in use before attempting the migration again. Then, resubmit the database migration command under the current release of Database Manager.

SQL1701N

The database cannot be migrated because the last session was abnormally terminated.

Cause: The last database operation was abnormally terminated (for example, a power failure) under the previous database manager. Database migration cannot be done until the database has been restarted.

The command cannot be processed.

Action: The database must be restarted. You must issue the RESTART DATABASE command using the release of Database Manager where the database was last accessed. Then, resubmit the database migration command under the current release of Database Manager.

SQL1702W

The "<protocol>" connection managers have started successfully.

Action: No action required.

SQL1703W

The db2event directory could not be created during database migration.

Cause: The database was migrated successfully but the db2event directory could not be created.

This is a warning only.

Action: The db2event directory must be created if the event monitor is to be used. The db2event directory must be created in the database directory where the migrated database resides. The database directory of the migrated database can be determined by invoking LIST DATABASE DIRECTORY.

SQL1704N

Database migration failed. Reason code "<reason-code>".

Cause: Database migration failed. The reason codes are as follows:

1 an invalid schema name was found.

2 database is not migrateable. The database could be in one of the following states:

- backup pending state
- roll forward pending state
- transaction inconsistent state.

3 database logs are full.

4 insufficient disk space.

5 cannot update database configuration file.

Action: Possible solutions based on the reason code are:

1 The reserved schema names are: SYSIBM, SYSCAT, and SYSSTAT. Ensure that all the database objects that use one or more of these schema names are dropped and recreate the objects using a different schema name. This correction must be made in the release of database manager in use prior to the current release. Resubmit the database migration command under the current release.

2 Correct the database state by going back to release of database manager in use prior to the current release and perform any necessary corrective action to the database. Resubmit the database migration command under the current release.

3 Increase the database configuration parameters *logfilsiz* or *logprimary* to a larger value. Resubmit the database migration command.

4 Ensure that there is sufficient disk space and resubmit the database migration command.

5 There was a problem in updating the database configuration file. Ensure that the database configuration file is not being held exclusively by any users and is updateable. Resubmit the database migration command. If the problem persists, inform your IBM service representatives.

SQL1705W

Unable to update a database directory entry to the current release level.

Cause: One or more database aliases could not be updated in the database directory for a database which has just been migrated from a previous release.

Action: Uncatalog the database alias(es) for the migrated database and re-catalog the database alias using the same information.

SQL1820N

Action on the LOB value failed. Reason code = "<reason-code>".

Cause: The possible reasons codes are:

1. Not enough buffer space to store the LOB value.

2. The remote data source does not support the current action on LOB datatype.

3. Some internal programming limitation exceeded.

Action: Reduce the size of the LOB or replace the function which is being applied on the LOB datatype. As a last resort, remove the LOB datatype from the statement.

sqlcode: -1820

sqlstate: 560A0

SQL1821W

The LOB value retrieved may have changed.

Cause: The LOB value was evaluated on a deferred retrieval basis. The LOB value may have changed between the time it was first accessed and when it was actually retrieved.

Action: Set the "deferred_lob_retrieval" to "N" in SYSSERVEROPTIONS, resubmit your query, or ignore the warning. See the *Application Programming and SQL Reference Supplement* for information on setting the value used for the "deferred_lob_retrieval" setting.

sqlcode: +1821

sqlstate: 01621

SQL1822N

Unexpected error code "<error code>" received from data source "<data source name>". Associated text and tokens are "<tokens>".

Cause: While referencing a data source, DataJoiner received an unexpected error code from the data source that does not map to a DB2 equivalent.

Action: Identify and correct the root cause of the problem by locating the appropriate message text and corrective action for the specified error at the data source.

SQL1823N

No data type mapping exists for data type "<data type name>" from server "<server name>".

Cause: An attempt was made to create a nickname for an object. One or more columns of that object are of a type currently unknown to DataJoiner. The name of (one of) the unknown type(s) is listed in the message.

Action: Create a mapping for the specified type name on the specified server using the CREATE TYPE MAPPING statement.

sqlcode: -1823

sqlstate: 428C5

SQL1824W   SQL1828N

SQL1824W

Some base tables in the operands of this UNION ALL may be the same table.

Cause: A nickname can refer to a remote base table, a remote view, a remote alias/synonym, or a remote nickname. If two operands of a UNION ALL view refer to different nicknames, they may potentially be pointing to the same table (if not both of them are known to be remote base tables). This message is issued to warn the user that potentially one remote base table can get updated/deleted twice via updates/deletes through two operands.

Action: Verify if all operands point to different remote tables. If two operands point to the same remote base table, consider issuing a rollback to reverse the update/delete operation.

sqlcode: +1824

sqlstate: 01620

SQL1825N

This SQL statement cannot be handled in DataJoiner.

Cause: Due to some limitation, the current SQL statement cannot be handled in DataJoiner. Possible limitations include:

- A cursor update/delete statement involves a refetch of a column into DataJoiner that is not being fetched in the cursor select statement.
- Some internal programming limitation is exceeded.

Action: Based on the cause:

- If this is the cursor update/delete statement that involves refetch of a column not being fetched in the cursor select statement, modify the cursor select statement so that the needed column is fetched.
- If some internal programming limitation is exceeded, try to simplify or rewrite the part of the statement that seems complex, e.g. expression.

sqlcode: -1825

sqlstate: 429A9

SQL1826N

An invalid value "<value>" was specified for column "<column-name>" in a system catalog object "<object-name>".

Cause: An invalid value "<value>" was specified for a column "<column-name>" in a system catalog object "<object-name>".

The statement cannot be processed.

Action: Refer to the *IBM DataJoiner Application Programming and SQL Reference Supplement* for valid values for the specified column in the specified system catalog object. Correct the statement and re-issue it.

sqlcode: -1826

sqlstate: 23521

SQL1827N

No user mapping is defined from local authorization ID "<auth-ID>" to server "<server-name>".

Cause: An attempt was made to drop or alter a user mapping that is not defined.

Action: If it is an ALTER USER MAPPING statement, first create a user mapping using the CREATE USER MAPPING statement. Then, alter the user mapping. If it is a DROP USER MAPPING statement, no extra action is needed because the user mapping does not exist.

sqlcode: -1827

sqlstate: 42704

SQL1828N

No server option "<option-name>" is defined for remote server "<server-name>" or for a group of remote servers: server type "<server-type>", version "<server-version>", and protocol "<server-protocol>".

Cause: An attempt was made to drop or alter a server option that is not defined.

Action: If it is an ALTER SERVER OPTION statement, first create a server option using the CREATE SERVER OPTION statement. Then, alter the server option. If it is a DROP SERVER OPTION statement, no extra action is needed because the server option for the servers does not exist.

sqlcode: -1828

sqlstate: 42704

SQL1900N

The command completed successfully.

Cause: The command line utility successfully completed the command.

Action: No action is required.

SQL1901N

The syntax of the command is not correct.

Cause: The command line utility could not process the command.

Action: Correct and resubmit the command.

SQL2000N

The drive specified in the utility command is not a valid diskette drive or fixed disk.

Cause: The input or output drive specified in the utility command does not exist.

The utility stops processing.

Action: Resubmit the utility command with the correct drive specifier.

NOTE: This message is applicable to DB2 for OS/2 only.

SQL2001N

The utility was interrupted. The output data may be incomplete.

Cause: The user may have pressed the interrupt key sequence, or had called the utility with a terminate caller action.

The utility stops processing.

Action: Restart the application or reissue the command, if needed. The output data from the interrupted command may be incomplete, and should not be used.

SQL2002N

The specified database utility command is not valid for remote databases. The database specified in the command must reside on your local workstation.

Cause: The database utility command is only valid for local databases.

The utility stops processing.

Action: Run the utility locally.

SQL2003C

A system error occurred.

Cause: An operating system error occurred. The return code can be found in the SQLERRD[0] field of the SQLCA.

The utility stops processing.

Action: Look at the error return code in the SQLERRD[0] field of the SQLCA. Correct the error, if possible, and resubmit the command.

SQL2004N

An SQL error "<sqlcode>" occurred during processing.

Cause: The utility was using an SQL statement when an error occurred.

The utility stops processing.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL2005C

An I/O error occurred during a read operation. The data may be incomplete.

Cause: Incomplete data was read during an I/O operation.

The utility stops processing.

Action: Determine if the I/O error can be corrected and resubmit the command.

SQL2006C SQL2014N

SQL2006C

An I/O error occurred during a write operation. The data may be incomplete.

Cause: Incomplete data was written during an I/O operation.

The utility stops processing.

Action: Determine if the I/O error can be corrected and resubmit the command.

SQL2008N

The callerac parameter is not within valid range or the requested action is out of sequence.

Cause: Either the *callerac* parameter value is not one of the acceptable values or the requested action is out of sequence.

The command cannot be processed.

Action: Each utility has its own list of valid *callerac* values. Check the *API Reference* for the valid values for the *callerac* for the utility in use. Resubmit the command with a valid *callerac* parameter.

SQL2009C

There is not enough memory available to run the utility.

Cause: More memory is needed to run the specified utility.

The command cannot be processed.

Action: Possible actions include:

- Verify that your system has sufficient real and virtual memory.
- Remove background processes.

SQL2010N

The utility encountered an error "<error>" while connecting to the database.

Cause: The utility could not connect to the database.

The utility stops processing.

Action: Look at the error number in the message for more information. Make changes and resubmit the command.

SQL2011N

The utility encountered an error "<error>" while disconnecting from the database.

Cause: The utility could not disconnect from the database.

The utility stops processing.

Action: Look at the error number in the message for more information. Make changes and resubmit the command.

SQL2012N

Interrupt handling could not be enabled.

Cause: The utility could not enable interrupt handling. The actual return code can be found in the SQLERRD[0] field of the SQLCA.

The utility stops processing.

Action: Look at the error return code in the SQLERRD[0] field of the SQLCA. Correct the error, if possible, and resubmit the command.

SQL2013N

The utility could not access the database directory. Error "<error>" was returned.

Cause: An error occurred while the utility was accessing the database directory.

The utility stops processing.

Action: If the path in the *database* parameter is not the database directory path, resubmit the command with the correct path. Otherwise, look at the error number in the message for more information. Make changes and resubmit the command.

SQL2014N

A database environment error occurred.

Cause: The utility received an error from a database environment command. The database manager configuration file and the database configuration file may contain incompatible values.

The utility stops processing.

Action: Check the database manager configuration file and the database configuration file for inconsistent values. Resubmit the command.

SQL2015N

The database parameter is not valid. The database name is too long, it was not specified, or the address of the name is not valid.

Cause: The database name is required. It must contain 1 to 8 characters and the characters must be chosen from the database manager. base character set and the name must be located at a valid address for the application.

The utility stops processing.

Action: Resubmit the command with a valid database name.

SQL2016C

The PATH command does not contain a path to "<program-name>".

Cause: The utility could not find the required program using the operating system Select Path.

The utility stops processing.

Action: Update the PATH command to include the path to the indicated program name.

SQL2017N

Too many sessions are already started or OS/2 Start Session did not complete successfully.

Cause: The BACKUP or RESTORE utility could not start the new session because:

- The maximum number of sessions is already started.
- The OS/2 Start Session program returned an error.

The utility stops processing.

Action: Wait until some of the current sessions stop processing and resubmit the command. Or, see the SQLERRD[0] field in the SQLCA for more information and resubmit the command.

SQL2018N

The utility encountered an error "<error>" while attempting to verify the user's authorization ID or database authorizations.

Cause: The user attempted to execute a utility and one of the following has occurred:

- The user authorization ID is not valid.
- An error occurred attempting to access the user's authorizations for the database.

The utility stops processing.

Action: Look at the error number in the message for more information. Make changes and resubmit the command.

SQL2019N

An error occurred while binding the utilities to the database.

Cause: Because the current level of the utility being executed was not bound to the database, the system attempted to bind all utilities to the database and this binding process failed. Possible causes for the error include:

- The system may have run out of disk space.
- A system resource problem such as too many files opened.
- A missing or invalid list of utility programs to be bound (db2ubind.lst).
- A missing or invalid bind file for one of the utilities (db2uxxxx.bnd).
- The user does not have the required authorization for binding utilities. The required privileges are:
  - BIND privilege for the utility programs
  - SELECT privilege for the system catalogs.

For the RESTORE utility, the database is restored, but at least one utility is not bound to the database. The other utilities stop processing.

Action: Complete all activities that may be competing for system resources and resubmit the utility command. If the error continues, perform one of the following actions:

- Have a user with SYSADM or DBADM authority retry the command.
- Reinstall the database manager and reapply any recent updates or do both.
- Bind the utility programs (db2uxxxx.bnd files) individually to the database (with no format option) to isolate the problem and, possibly enable some utilities to operate successfully.

SQL2020N

The utility is not bound to the database correctly.

Cause: All the utilities were rebound to the database because the utility was not bound to the database or the package of the utility that was bound to the database was not compatible with the installed version of the database manager, but there is still a timestamp conflict between the installed version of the database manager and the package so the utility and bind file are not compatible.

The utility stops processing.

Action: Reinstall the database manager or reapply any recent updates or do both. Resubmit the utility command.

SQL2021N

The correct diskette is not in the drive.

Cause: The diskette to be used for Backup Database or for Restore Database is not in the drive or is not valid.

The command cannot be processed.

Action: Verify that the correct diskette is in the drive or insert a new diskette.

SQL2023N

The utility encountered an I/O error "<code>" while accessing the log control file.

Cause: The read or write operation against the log control file failed.

The utility stops processing.

Action: Record the error return code value. Determine whether the I/O error can be corrected.

SQL2024N

The utility encountered an I/O error "<code>" while accessing the .BRG file.

Cause: An I/O error occurred while accessing the specified file.

The .BRG file is used to determine whether a Restore operation has terminated abnormally. The .BRG file resides in the local database directory of the database in question and consists of a filename equivalent to the database token and filetype of .BRG.

The utility stops processing.

Action: Record the error return code. Determine whether the I/O error can be corrected.

SQL2025N

An I/O error "<code>" occurred on media "<dir/devices>".

Cause: An I/O error occurred while accessing a file on the specified media.

The utility stops processing.

Action: Record the error return code. Determine whether the I/O error can be corrected.

SQL2026N

An error "<sqlcode>" occurred while trying to internally disconnect from the database.

Cause: The internal disconnect command failed. The SQLCODE is returned in the message.

The utility stops processing.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL2027N

An error "<sqlcode>" occurred while trying to internally connect to the database.

Cause: The internal connection failed. The SQLCODE is returned in the message. The database manager configuration file and the database configuration file may contain incompatible values.

The utility stops processing.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command. Check that the database manager configuration file values and the backup image's database configuration file values are compatible.

SQL2028N

Error "<sqlcode>" occurred while installing the interrupt handler.

Cause: The utility could not enable the interrupt handler. The SQLCODE is returned in the message.

The utility stops processing.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL2029N

An error "<error>" occurred while "<command-file-name>" was processing.

Cause: An error was returned from either the specified command file, or from the operating system.

Action: The user has either requested a "Backup changes only" of a database enabled for ROLLFORWARD recovery, or has requested a "Backup changes only" while requesting the use of the user exit.

SQL2030N

The "<name>" drive is full. At least "<number>" free bytes are needed on this drive.

Cause: The specified drive does not have enough space for creating an internal subdirectory and information file.

The utility stops processing.

Action: Free the indicated space on the specified drive and resubmit the command.

SQL2031W

Warning! Please mount the target or source media on device "<device>".

Cause: A database utility process either writes data to or reads data from the media on the specified device. The utility returns so that the user may mount the correct media for the operation.

The utility waits for a response to continue.

Action: Mount the media and return to the utility with the *callerac* parameter indicating if processing continues or ends.

SQL2032N

The "<parameter>" parameter is not valid.

Cause: The parameter is incorrectly specified. Either the value is out of range or is incorrect.

Action: Resubmit the command with a correct value for the parameter.

SQL2033N

The ADSM error "<error>" occurred.

Cause: During the processing of a database utility, ADSM was called and encountered an error.

Action: Consult the ADSM documentation for a description of the error, take corrective actions and resubmit the command.

SQL2034N

The address of the "<parm>" parameter is not valid.

Cause: The application program used an address that is not valid for this parameter. Either the address points to an unallocated buffer or the character string in the buffer does not have a null terminator.

Action: Ensure that a valid address is used in the application program and the input string is null terminated.

SQL2035N

The warning condition "<warn>" was encountered while executing the utility in non-interrupt mode.

Cause: The calling application invoked the utility with *no interrupt* mode. During the operation a warning condition was encountered.

Action: Resubmit the operation without specifying the *no interrupt* condition in the *callerac* parameter or take actions to prevent the warning and resubmit the operation.

SQL2036N   SQL2042W

SQL2036N

The path for the file or device "<path/device>" is not valid.

Cause: The application calling the utility has supplied a source or target path that is not valid. The path or device specified may not exist or is incorrectly specified.

Action: Reissue the utility command with a path that represents a correct path or device.

SQL2037N

ADSM could not be loaded.

Cause: A call to a database utility specified ADSM as the target or source of the backup. An attempt was made to load the ADSM client. Either the ADSM client is not available on the system or an error was encountered in the load procedure.

Action: Ensure that the system has ADSM available. Resubmit the command after ADSM is made available, or resubmit the command without utilizing ADSM.

SQL2038N

A database system error "<errcode>" occurred during processing.

Cause: A database system error was encountered during the processing of one of the utilities.

Action: Examine the error code in the message for more information. Take corrective action and resubmit the command.

SQL2039N

The application calling the utility has terminated.

Cause: The application calling the utility has terminated. The application side of the utility is in the same process as the calling application and terminates with the application. The agent side of the utility terminates as a result.

Action: Reissue the command after determining why the application terminated.

SQL2040N

The database alias parameter "<dbalias>" is not valid or not specified.

Cause: The application calling the Backup or Restore utility has supplied a database alias parameter that is not valid. The alias must be 1 to 8 bytes and the characters must be chosen from the database manager base character set.

Action: Reissue the Backup or Restore command with a valid database alias.

SQL2041N

The buffer size parameter specified is not valid. The buffer size must be specified as 0 or be between 16 and 16384 inclusive.

Cause: The application calling the utility has supplied a buffer size parameter that is not valid. The buffer size is used to determine the internal buffer size. The value is the number of 4K pages that are obtained for this buffer. The value must be specified as 0 or be between 16 and 16384 inclusive.

When running the backup or restore utility, if 0 is specified, the default buffer size defined in the database manager configuration is used.

If the target media is a diskette, the buffer size must be smaller than the diskette size.

SQL2042W

Warning! I/O error "<error>" accessing device "<device>". Please ensure media is mounted and positioned correctly.

Cause: The application calling the utility has encountered an I/O error while trying to read or write to the tape device. The utility returns so that the user may correctly mount and position the tape. The utility waits for a response to continue.

Action: Correctly mount and position the tape and return to the utility indicating if processing continues or ends.

SQL2043N

Unable to start a child process or thread.

Cause: Unable to start up the child processes or threads required during the processing of a database utility. On UNIX-based systems, this may mean that the maxuproc value is too low. On OS/2-based systems, this may mean that the THREADS CONFIG.SYS value is too low. The problem may also be that there was insufficient memory to create the new process or thread. The utility stops processing.

Action: Ensure the system limit for number of processes or threads has not been reached (either increase the limit or reduce the number of processes or threads already running). Ensure that there is sufficient memory for the new process or thread. Resubmit the utility command.

SQL2044N

An error occurred while accessing a message queue. Reason code: "<reason-code>"

Cause: An unexpected error or bad message was received on one of the message queues during the processing of a database utility. The following is a list of reason codes:

1   Unable to create a message queue. The number of message queues allowed may have been exceeded.
2   An error was encountered while reading from a message queue.
3   An error was encountered while writing to a message queue.
4   An invalid message was received from a message queue.
5   An error was encountered while opening a message queue.
6   An error was encountered while closing a message queue.
7   An error was encountered while querying a message queue.
8   An error was encountered while deleting a message queue.

The utility stops processing.

Action: Ensure the number of message queues allowed is not reached. Reduce the number of message queues in use if necessary and resubmit the utility command.

SQL2045W

Warning! SQLCODE "<sqlcode>" occurred while writing to media "<media>".

Cause: A database utility process encountered SQLCODE "<sqlcode>" while writing to the media "<media>". The utility returns so that the user may attempt to fix the problem or cancel the operation.

The utility waits for a response to continue.

Action: Correct the "<sqlcode>" condition and return to the utility with the correct caller action parameter to indicate if processing should continue or terminate.

SQL2048N

An error occurred while accessing object "<object>". Reason code: "<reason-code>".

Cause: An error occurred while accessing an object during the processing of a database utility. The following is a list of reason codes:

1   An invalid object type is encountered.
2   A lock object operation failed. The lock wait may have reached the lock timeout limit specified in the database configuration.
3   An unlock object operation failed during the processing of a database utility.
4   Access to an object failed.
5   An object in the database is corrupted.
6   The object being accessed is a table space and either the table space is in such a state that the operation is not allowed or one or more containers of the table space is not available. (LIST TABLESPACES will list the current table space state.)
7   A delete object operation failed.

The utility stops processing.

Action: If a lock object operation failed, ensure the lock timeout limit in the database configuration is adequate and resubmit the utility command. You may also consider using the QUIESCE command to bring the database to a quiesced state to ensure access.

If the error was encountered during backup or unload, take corrective action to recover the database and resubmit the utility command.

If the error was encountered during restore or load recovery, ensure the backup or copy image is correct and resubmit the utility command.

If the object is a table space and the error was encountered during a restore, the set table space

container api may be used to modify the table space before resubmitting the utility command.

SQL2054N

The backup or copy image is corrupted.

Cause: The backup or copy image in use is corrupted

The utility stops processing.

Action: Discard the image as it is not a valid image. Resubmit the utility command with a valid image.

SQL2055N

Unable to access memory from memory set "<memory-heap>".

Cause: A database utility was unable to access memory during processing.

The utility stops processing.

Action: Stop the database manager, then restart it and resubmit the utility command.

SQL2056N

An invalid media type was encountered on media "<media>".

Cause: An invalid media type was encountered during the processing of a database utility.

The utility stops processing.

Action: Ensure the media used is among of the types supported by the utility. Resubmit the command with a valid media list.

SQL2057N

The media "<media>" is already opened by another process.

Cause: The source or target media specified during the processing of a database utility is already open by another process. The utility does not allow shared access for the operation.

The utility stops processing.

Action: Ensure the media used is not in use currently. Resubmit the command with a valid media list.

SQL2058W

An end-of-media warning was encountered on media "<media>".

Cause: An end-of-media warning was encountered during the processing of a database utility.

The utility is waiting for a response to continue.

Action: Correct the end-of-media condition and return to the utility with the correct caller action parameter to indicate if processing should continue or terminate.

SQL2059W

A device full warning was encountered on device "<device>".

Cause: An device full warning was encountered during the processing of a database utility.

The utility is waiting for a response to continue.

Action: Correct the device full condition and return to the utility with the correct caller action parameter to indicate if processing should continue or terminate.

SQL2060W

The device "<device>" is empty.

Cause: An empty device was encountered during the processing of a database utility. The utility is waiting for a response to continue.

Action: Mount the media and return to the utility with the caller action parameter to indicate if processing should continue or terminate.

SQL2061N

An attempt to access media "<media>" is denied.

Cause: An attempt to access a device, file, ADSM or the vendor shared library is denied during the processing of a database utility. The utility stops processing.

Action: Ensure the device, file, ADSM or vendor shared library used by the utility allows the access requested and resubmit the utility command.

SQL2062N

An error occurred while accessing media "<media>". Reason code: "<reason-code>"

Cause: An unexpected error occurred while accessing a device, file, ADSM or the vendor shared library during the processing of a database utility. The following is a list of reason codes:

1 An attempt to initialize a device, file, ADSM or the vendor shared library failed.
2 An attempt to terminate a device, file, ADSM or the vendor shared library failed.

The utility stops processing.

Action: Ensure the device, file, ADSM or vendor shared library used by the utility is available and resubmit the utility command. If the command is still unsuccessful, contact your technical service representative.

SQL2065W

Caller action "<caller-action>" specified is not allowed when the media "<media>" specified is the only one left connected to the utility.

Cause: The caller action specified is not allowed because there is only one device left connected to the database utility.

Action: Return to the utility with the correct caller action parameter to indicate if processing should continue or terminate.

SQL2066N

Table space name "<name>" specified does not exist in the database or cannot be used for the backup operation.

Cause: The table space name specified is syntactically correct, but it does not exist in the database or cannot be used in a backup operation. The table space may be disallowed because it is a temporary table space or is in an inconsistent state.

Action: Check the table space name and resubmit the backup command with the correct table space name.

SQL2068N

An invalid image was encountered on media "<media>". There was no media header.

Cause: An invalid image was encountered during the processing of a database utility. The utility was not able to locate a valid media header. The utility stops processing.

Action: Resubmit the command with correct backup or copy images.

SQL2069N

An invalid image was encountered on media "<media>". The image was created for database alias "<dbalias>".

Cause: An invalid image was encountered during the processing of a database utility. The image provided was from a different database alias. The utility stops processing.

Action: Resubmit the command with correct backup or copy images.

SQL2070N

An invalid image was encountered on media "<media>". The image contained timestamp "<timestamp>".

Cause: An invalid image was encountered during the processing of a database utility. The image provided was from a backup or copy with a different timestamp. The utility stops processing.

Action: Resubmit the command with correct backup or copy images.

SQL2071N

An error occurred while accessing the shared library "<shr-lib-name>". Reason code: "<reason-code>"

Cause: An unexpected error occurred while accessing a vendor shared library during the processing of a database utility. The following is a list of reason codes:

1 An invalid shared library path was encountered.
2 An attempt to load the backup shared library failed.
3 An error was encountered while unloading the shared library.

The utility stops processing.

Action: Ensure the shared library provided is valid and resubmit the utility command or use another supported media.

SQL2072N

Unable to bind the shared library "<shr-lib-name>". Reason code: "<reason-code>".

Cause: An error was encountered while binding the shared library during the processing of a database utility. The utility stops processing.

Action: Note the reason code returned from the vendor utility in the message and take corrective action if possible. Resubmit the command with a valid shared library or using another supported media.

SQL2154N

The RESTORE was not successful. The backup used to restore the tablespace(s) is not associated with the current log sequence for the database.

Cause: For tablespace restore, the backup must have been taken during the current log sequence of the database. The log file sequence is determined by the database that has been restored previously or the log files that have been processed. In addition, the backup must have been taken after the last time the database was enabled for roll forward recovery.

Tablespace restore has stopped.

Action: Resubmit the command with a correct backup image.

SQL2155W

Changes have been made to the recovery history file since the open scan was issued.

Cause: Changes have been made to the recovery history file since the file was open for a scan. The data read may not be consistent.

Action: If it is essential to have consistent data from the scan, close the recovery history file and resubmit the command.

SQL2157N

There are already eight recovery history file scans open.

Cause: Eight recovery history file scans in this process are already open. No more than eight open scans are allowed.

The command cannot be processed.

Action: Issue one or more CLOSE RECOVERY HISTORY FILE SCAN commands and resubmit the command.

SQL2160W

A damaged recovery history file has been replaced. Processing continued.

Cause: An error was encountered during an access to the recovery history file. The utility is able to recover the file from an alternate copy. The utility continues processing.

Action: The utility continues processing successfully. The user should take appropriate precautions to ensure the recovery history file is not damaged again.

SQL2161N

A damaged recovery history file could not be fixed. The specified action failed.

Cause: An error was encountered during an access to the recovery history file. The utility is not able to recover the file. The utility stops processing.

Action: To continue processing, the user may remove the recovery history file and resubmit the command. The utility will regenerate a new file. Any data from the corrupted file are lost. The user may look at the corrupted file to see if any information can be saved. The user should take appropriate precautions to ensure the recovery history file is not damaged again.

SQL2165W

The SQLUHINFO structure did not provide enough TABLESPACE entries.

Cause: The SQLUHINFO structure was not large enough to hold the entry that was to be returned (including all of its TABLESPACES). The value of the SQLN field of the SQLUHINFO structure should be at least as large as the SQLD field returned.

The command cannot be processed.

Action: Increase the value of the SQLN field in the SQLUHINFO structure to the value indicated by the SQLD field (making sure the SQLUHINFO structure is large enough to support that amount) and resubmit the command.

SQL2171N

The update of the recovery history file failed because the specified object part does not exist in the file.

Cause: The entry specified to be updated in the recovery history file does not exist in the file. The utility stops processing.

Action: Resubmit the command with a valid entry.

SQL2200N

The qualifier for the table or index name is too long or it was not specified as part of the table or index name.

Cause: The table name or index must be fully qualified. The format is *authid.name* where *authid* must be 1 to 8 characters and *name* must be 1 to 18 characters (bytes in a MBCS environment).

The utility stops processing.

Action: Resubmit the command with a fully qualified name including the correct qualifier.

SQL2203N

The tablename parameter is not valid. Either the table name is too long, only an authorization ID was specified, the table name was not specified, or the address of the name is not valid.

Cause: The table name is required. It must be fully qualified and in the format *authid.name* where *authid* is 1 to 8 characters and *name* is 1 to 18 characters (bytes in MBCS environments). The table name must be located at a valid application address.

The utility stops processing.

Action: Resubmit the command with the correct table name.

SQL2204N

The iname parameter is not valid. The name of the index is too long, only an authorization ID was specified, or the address of the index is not valid.

Cause: If an index is specified, it must be fully qualified and in the format *authid.name* where *authid* is 1 to 8 characters and *name* is 1 to 18 characters (bytes in MBCS environments). The index must be located at a valid application address.

The utility stops processing.

Action: Resubmit the command with a valid index name.

SQL2205N

The specified index does not exist.

Cause: The specified index does not exist for the specified *authid.table* or *authid.index*.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

The utility or operation stops processing.

Action: Resubmit the command with a valid index.

SQL2207N

The filepath specified by the datafile parameter is not valid.

Cause: The datafile parameter is not any of the values that indicate the default filepath. Also the datafile parameter is not a valid non-default value. One of the following may apply:

- The pointer is not valid.
- The pointer points to a string that is too long for a filepath designation.
- The specified path value is not valid (on the server machine).
- The file path does not end with an appropriate separator for the server machine.

The utility stops processing.

Action: Resubmit the command with a valid datafile parameter.

SQL2208N   SQL2216N

SQL2208N

The table space specified by the table space parameter is not valid.

Cause: The table space parameter does not contain a valid value. One of the following conditions may exist:

- The pointer is not valid
- The pointer points to a string that is too long for a table space name
- The specified table space does not exist.

The utility stops processing.

Action: Resubmit the command with a valid table space parameter or do not use the table space name parameter. In the latter case, the Reorganize Table utility will use the table space(s) in which the table itself resides.

SQL2211N

The specified table does not exist.

Cause: The table does not exist in the database. Either the table name or the authorization ID is incorrect.

The command cannot be processed.

Action: Resubmit the command with a valid table name.

SQL2212N

The specified table is a view. The Reorganize Table utility cannot be run against a view.

Cause: The Reorganize Table utility cannot be run against a view.

The utility stops processing.

Action: Resubmit the command with a valid table name.

SQL2213N

The specified table space is not a temporary table space.

Cause: The Reorganize Table utility requires that any table space specified is a temporary table space. The table space name provided is not a table space defined to hold temporary tables.

The utility stops processing.

Action: Resubmit the command with the name of a temporary table space or do not use the table space name parameter. In the latter case, the Reorganize Table utility will use the table space(s) in which the table itself resides.

SQL2214N

The user does not have the authority to run the Reorganize Table utility on table "<name>".

Cause: The user attempted to reorganize the specified table without the appropriate authorization (either SYSADM or DBADM authority or the CONTROL privilege on the table).

The utility stops processing.

Action: Logon as a user with the appropriate authority or privilege and resubmit the reorganize table utility command.

SQL2215N

SQL error "<sqlcode>" occurred while committing previous work for the database.

Cause: The user was already connected to the database specified in the Reorganize Table command. An error occurred while committing the previous work in progress for the database.

The utility stops processing with no attempt to roll back the work or break the database connection.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL2216N

SQL error "<sqlcode>" occurred while reorganizing a database table.

Cause: An error occurred during the reorganization of a database table.

The utility stops processing.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL2300N

The identifier for the table name is too long or it was not specified as part of the table name.

Cause: The table name must be fully qualified. The format is *authid.tablename* where *authid* contains 1 to 8 characters and *tablename* contains 1 to 18 characters (bytes in MBCS environments).

The utility stops processing.

Action: Resubmit the command with the fully qualified table name, including the correct qualifier.

SQL2301N

The tablename parameter is not valid. Either it is too long, only an authorization ID was specified, or the address of the name is not valid.

Cause: The table name must be fully qualified and in the format *authid.name* where *authid* contains 1 to 8 characters and *name* contains 1 to 18 characters (bytes in MBCS environments) and must be located at a valid application address.

The utility stops processing.

Action: Resubmit the command with the correct table name.

SQL2302N

The index list is not valid. Either the address for the list is not valid, the number of entries in the list is less than the number of indexes specified, or the address for an index in the list is not valid.

Cause: The address for the list is not valid, the number of entries in the list is less than the number of indexes specified, or the address for an index in the list is not valid.

The utility stops processing.

Action: Resubmit the command with a valid index list.

SQL2303N

The statsopt parameter is not valid.

Cause: The *statsopt* parameter in the Run Statistics command must be one of the following:

- T for basic table only
- I for basic indexes only
- B for both basic table and indexes
- D for table and distribution
- E for table and distribution and basic indexes
- X for extended indexes only
- Y for extended indexes and basic table
- A for all

The utility stops processing.

Action: Resubmit the command with a valid *statsopt* parameter.

SQL2304N

The sharelev parameter is not valid. It must be 'R' for reference or 'C' for change.

Cause: The *sharelev* parameter in the RUN STATISTICS command must be R for reference or C for change.

The utility stops processing.

Action: Resubmit the command with a valid *sharelev* parameter.

SQL2305N

The table specified is a view. The utility cannot be run against a view.

Cause: The *tname* parameter specified a view rather than a table.

The utility stops processing.

Action: Resubmit the command with a valid *tname* parameter.

SQL2306N

The table or index "<name>" does not exist.

Cause: The table or index identified by "<name>" does not exist in the database, or the index identified by "<name>" is not defined on the specified table. The qualifier for the table or one of the indexes may be incorrect.

The utility stops processing.

SQL2307N   SQL2400N

Action: Resubmit the command with valid table names and indexes.

SQL2307N

The table specified is a system table. The Runstats utility cannot be run against a system table.

Cause: The Run Statistics utility command may not be run against a system table.

The utility stops processing.

Action: Resubmit the command with a valid table name.

SQL2308N

The qualifier for the index name "<name>" is too long, or it was not specified as part of the index name.

Cause: The index name must be a fully qualified name. The format is *authid.tablename* where *authid* contains 1 to 8 characters and *tablename* contains 1 to 18 characters (bytes in MBCS environments).

The utility stops processing.

Action: Resubmit the command with a fully qualified name, including the correct qualifier.

SQL2309N

The index name "<name>" is not valid. It is too long or only a qualifier was specified.

Cause: The index name must be a fully qualified name. The format is *authid.name* where *authid* contains 1 to 8 characters and *name* contains 1 to 18 characters (bytes in MBCS environments).

The utility stops processing.

Action: Resubmit the command with a valid index.

SQL2310N

The utility could not generate statistics. Error "<sqlcode>" was returned.

Cause: An error occurred while the utility was gathering statistics.

The utility stops processing.

Action: Look at the message error number for more information. Make changes and resubmit the command.

SQL2311N

The user does not have the authority to run the Run Statistics utility on table "<name>".

Cause: The user attempted to run statistics on the specified table without the appropriate authorization (either the SYSADM or DBADM authority or the CONTROL privilege on the table).

The utility stops processing.

Action: Do not attempt to call the Run Statistics utility command without appropriate authorization.

SQL2312N

Statistics heap size is too small to carry out the operation. Suggested heap size is "<num>" pages.

Cause: The setting of the database configuration parameter *stat_heap_sz* is not big enough to collect non-uniform distribution statistics on the table.

Action: Update the database configuration parameter *stat_heap_sz* to the suggested value and try again.

SQL2313W

All available memory in the statistics heap has been used. Statistics were collected on the first "<n1>" rows out of a total of "<n2>" rows.

Cause: The setting of the database configuration parameter *stat_heap_sz* is not big enough to collect non-uniform distribution statistics on the table. Only <n1> of <n2> rows could be processed.

Action: Update the database configuration parameter *stat_heap_sz* by 20 percent and try again.

SQL2400N

The type specified in the BACKUP command is not valid. It must be either 0 to back up the entire database or 1 to back up only the current changes.

Cause: The type must be either 0 for backing up the whole database or 1 for backing up changes only.

The utility stops processing.

Action: Resubmit the utility command with the correct type.

SQL2401N

"Backup changes only" cannot be performed until a backup of the entire database has been done. The type must be 0.

Cause: A backup of changes only has been requested without requesting a backup of the entire database first, or an internal file has been damaged such that the BACKUP utility cannot determine whether a full backup had been taken. The backup of changes only cannot be used until after a backup of the entire database.

The utility stops processing.

Action: Resubmit the utility command with the correct type.

SQL2403N

The user does not have the authority to run the utility against the database.

Cause: The user attempted to execute a database utility without either SYSADM or DBADM authority.

The utility stops processing.

Action: Do not attempt to call the utility command without appropriate authorization.

SQL2404N

The backup target media is full. At least "<number>" free bytes are needed on the target media.

Cause: The backup target media does not have enough space for creating an internal subdirectory and information file.

The utility stops processing.

Action: Prepare a backup media that has at least the specified number of bytes of free space and then resubmit the command.

SQL2405N

The BACKUP cannot be performed because there is a previously incomplete RESTORE.

Cause: The database is in an inconsistent state due to a system crash during a restore operation or due to an abnormally terminated RESTORE.

The utility stops processing.

Action: Resubmit the RESTORE command.

SQL2406N

The BACKUP cannot be performed because the database needs to be rolled forward.

Cause: The backup failed because the database is in an inconsistent state.

The utility stops processing.

Action: Use the ROLLFORWARD command to make the database usable. Then resubmit the BACKUP command.

SQL2407N

An I/O error occurred reading the file "<name>". The BACKUP cannot be performed because it cannot determine if a RESTORE was completed.

Cause: The indicated file cannot be read due to the I/O error. Since the file exists, a BACKUP or RESTORE command must have not completed. Upon successful completion of the process this file is deleted.

The utility stops processing.

Action: Determine which process was executed previously. Delete the indicated file and resubmit the previous command.

SQL2408W

The database was backed up, but an I/O error "<error>" occurred while deleting the file "<name>".

Cause: The BACKUP command ran successfully. The indicated file was not deleted due to an I/O error.

The utility completed processing.

Action: Try to delete the file indicated in the message.

SQL2409N

When doing a BACKUP of changes only, the most recent backup image must be used.

Cause: The user has attempted to perform a changes only backup but the backup image specified is not the most current backup or the previous backup of changes only failed.

The utility stops processing.

Action: Locate the most recent backup image and resubmit the command. If the most recent backup image cannot be located or if the previous backup of changes only failed, resubmit the command requesting a

backup of the entire database rather that a changes only backup.

SQL2410N

"Backup changes only" cannot be performed if the database is enabled for ROLLFORWARD or if the BACKUP is being done through a user exit.

Cause: The user requested a changes only backup of a database with roll-forward enabled, or has requested a changes only backup through the user exit program.

Action: Possible solutions include:

- If the database is roll-forward enabled, resubmit the BACKUP utility command requesting a backup of the entire database.

- Disable roll-forward by turning of the SQL_ENABLE_LOG_RETAIN flag and SQL_ENABLE_USER_EXIT flag in the database configuration file. Then resubmit the BACKUP utility command requesting a changes only backup.

- If the backup was requested through the user exit program, resubmit the BACKUP utility command to do a changes only backup not using the user exit program.

- If the backup was requested through the user exit program to a standard device, resubmit the BACKUP utility command to do a changes only backup not using the user exit program.

SQL2411C

An I/O error occurred while running a utility. Either the disk could not be physically written to or has been filled.

Cause: An operating system error occurred when the utility was writing to the disk or diskette. The actual return code can be found in the SQLERRD[0]field of the SQLCA.

Action: Look at the error return code in the SQLERRD[0] field of the SQLCA. Correct the error, if possible, and resubmit the command.

SQL2412C

A corrupted database page was encountered while running a database utility.

Cause: During the utility operation a corrupted database page was encountered. The database is in an unpredictable state and the utility cannot continue.

Action: Record the message number (SQLCODE).

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Then contact your technical service representative with the following information:

- Problem description
- SQLCODE or message number
- SQLCA if possible
- Trace file if possible.

SQL2413N

Online backup is not allowed because either logretain or userexit for roll-forward is not activated, or a backup pending condition is in effect for the database.

Cause: Online backup cannot be performed while the database is not logging for forward recovery as forward recovery will be required at restore time. Forward recovery is placed into effect by setting either the database configuration logretain parameter or the userexit parameter, and then performing an offline backup of the database.

Action: Execute an offline backup or reconfigure the database for roll-forward recovery and issue an offline backup so that subsequent online backups will be allowed.

SQL2414W

Warning! Insufficient space on device "<device>" to contain backup control information. This media will not contain backup information.

Cause: During the Backup operation a new media was mounted and positioned after one or more media became full. There is not enough space on this media to contain the Backup Control Information and the media should not be included during the Restore operation.

Action: Mount a new media, or reposition the current media to provide enough space for this header, then return to the utility with the *callerac* parameter indicating if processing continues.

SQL2416W

Warning! Device "<device>" is full. Please mount new media.

Cause: The tape which the utility is using has become full.

Action: Mount another tape and continue the operation with the *callerac* parameter indicating if processing continues.

SQL2418N

The database specified for Backup does not exist.

Cause: The database specified in the *dbase* parameter of the Database Backup command could not be found.

Action: Ensure that the correct database alias was specified for the Backup Database utility and that the database exists for this alias. Resubmit the command with the correct alias.

SQL2419N

The target disk has become full.

Cause: During the processing of a database utility, the target disk became full. The utility has stopped and the target is deleted.

Action: Ensure enough disk space is available for the utility or direct the target to other media, such as tape.

On unix-based systems, this disk full condition may be due to exceeding the maximum file size allowed for the current userid. Use the chuser command to update fsize. A reboot may be necessary.

SQL2420N

Insufficient space on device "<device>" to contain initial backup control information.

Cause: During the Backup operation an initial backup header needs to be created at the start of the backup image. This header must fit on one tape when backing up to tape. There is insufficient space on the tape to contain this header.

Action: Resubmit the Backup operation and either

direct the output to a new tape, or reposition the current tape to provide enough space for this header.

SQL2421N

Table space level backup is not allowed because roll forward recovery is not enabled.

Cause: Table space level backup cannot be performed while the database is not logging for forward recovery as forward recovery will be required at restore time. Forward recovery is placed into effect by setting either the database configuration logretain parameter or the userexit parameter, and then performing an offline backup of the database.

Action: Execute a full database backup or reconfigure the database for roll forward recovery and issue an offline backup so that subsequent table space level backups will be allowed.

SQL2422N

The database cannot be backed up with a back-level backup API due to incompatible tablespace locations.

Cause: This database contains tablespace(s) which have been defined in a location other than the database subdirectory. This is not compatible with the API being used.

Action: Use the current backup API to backup the database.

SQL2423N

The database cannot be backed up because some index files are missing.

Cause: Some of the index files required by the backup are missing. These index files must be recreated prior to backing up the database.

Action: Run the 'db2recri' program to recreate the missing index files and then resubmit the backup command.

SQL2501C • SQL2506W

SQL2501C

The database was restored but the data in the restored database was unusable.

Cause: The RESTORE utility either could not read data from the restored database or only a portion of the database was restored. Both cases indicate that the restored database is not usable.

The database is unusable and the RESTORE utility stops processing.

Action: Resubmit the RESTORE command.

SQL2502C

An error occurred while reading the backup files. Either the disk could not physically be read or the specified disk did not contain a valid backup.

Cause: An operating system error occurred when the RESTORE utility was reading the diskette or disk, or the diskette or disk does not contain a backup of the database directory or does not contain the results of any previous backup.

The utility stops processing.

Action: If the input drive specified is a diskette drive, check the diskette being used. If the input drive is a hard file, verify that it is the correct hard file. Resubmit the command with the correct input drive and the correct diskettes, if applicable.

SQL2503N

The RESTORE was not successful. The backup used to restore the database contained the wrong database.

Cause: The name of the database contained on the backup disk does not match the database name specified on the RESTORE command. The RESTORE utility cannot determine the name until after the database has been restored because a previous release backup image was restored.

The utility stops processing.

Action: If the wrong database name was used in the command, resubmit the command with the correct database name. If the input drive specified is a diskette drive, check the diskette being used. If the input drive is a hard file, verify that it is the correct hard file. Resubmit the command with the correct input drive and the correct diskettes, if applicable.

SQL2504W

Insert the first backup diskette into drive "<drive>".

Cause: The RESTORE utility reads the first backup diskette to determine the path of the database directory that was backed up. If the backup media is a diskette and the diskette is not found in the specified input drive, the utility returns to the caller with this prompt. The calling program is expected to query the user and return to the utility with the user response.

The utility waits for a response from the caller.

Action: Prompt the user for the diskette and return to the utility with the *callerac* parameter, indicating if processing continues or ends.

SQL2505W

Warning! Database "<name>" resides at "<drive>". The files for this database will be deleted.

Cause: If the database being restored already exists, its files will be deleted before the restore process starts. The utility returns to the caller with this warning. The caller is expected to query the user and return to the utility with the user response. Once the database is dropped, it cannot be accessed again.

The utility waits for response from the caller.

Action: Warn the user that the database files will be deleted and return to the utility with the *callerac* parameter indicating if processing continues or ends.

SQL2506W

The database was restored, but there may be extra files in the database.

Cause: If changes-only backups were performed and database files were deleted between backups, the RESTORE utility adds these deleted files back into the database. The restore process could not delete the extra files either because of an I/O error or by the failure of an internal connection to an internally stopped database.

The utility ends successfully.

Action: Either use the database as it is or attempt to restore it again. Ensure that the DB2 configuration is compatible with the restored database configuration before rerunning the RESTORE utility.

SQL2507W

The RESTORE utility is not bound to the database correctly.

Cause: Because the RESTORE utility was not bound to the database or the package of the utility that was bound to the database is not compatible with the installed version of DB2, all the utilities have been rebound to the database. But there is a timestamp conflict between the installed version of DB2 and the package so the utility and bind file are not compatible.

The database is restored, but the utility is not correctly bound.

Action: Reinstall DB2 or reapply recent updates and resubmit the utility command.

SQL2508N

The Database Restore timestamp parameter "<timestamp>" is not valid.

Cause: The timestamp parameter must contain either NULL or a portion of a valid timestamp, the portion consisting of complete components of a timestamp.

Action: Resubmit the Restore operation with a valid timestamp value.

SQL2509N

The database drive parameter is not valid.

Cause: The drive specified does not exist, or the database does not exist on this drive or is not cataloged on this drive. RESTORE should use the *db2uexit* command to perform the database restore.

The utility stops processing.

Action: Resubmit the utility command with a valid drive.

SQL2510N

An operating system semaphore error occurred.

Cause: An error occurred in the wait or post semaphore.

The utility stops processing.

Action: Stop the database manager, then restart it and resubmit the utility command.

SQL2511N

The utility encountered an error "<error>" while dropping the database.

Cause: The utility could not drop the database.

The utility stops processing.

Action: Look at the error number in the message for more information. Make changes and resubmit the command.

SQL2512N

The utility encountered an error "<error>" while creating the database.

Cause: The utility could not create the database.

The utility stops processing.

Action: Look at the error number in the message for more information. Make changes and resubmit the command.

SQL2513N

The utility encountered an error "<error>" renaming the database.

Cause: The utility could not rename the database.

The utility stops processing.

Action: Look at the error number in the message for more information. Make changes and resubmit the command.

SQL2514C

The RESTORE was not successful. The backup used to restore the database contained a database with a release number that is not compatible with the installed version of the database manager.

Cause: The release number of the restored database is not compatible with the release number of the installed version of the database manager.

The utility erases the restored database and stops processing.

Action: Resubmit the command with a backup that is compatible with the release level of the database manager.

SQL2515N   SQL2520W

SQL2515N

The user does not have the authority to run the RESTORE utility against the database.

Cause: The user attempted to execute the RESTORE utility without SYSADM authority.

The utility stops processing.

Action: Do not attempt to call the RESTORE utility command without appropriate authorization.

SQL2516N

The RESTORE utility cannot complete because at least one database on the workstation is in use.

Cause: In some situations, the RESTORE utility can attempt to move a database to a directory by renaming the directory associated with the database. This cannot be done if any databases are in use in any process on the workstation.

The utility stops processing.

Action: Wait until no databases on the workstation are in use and resubmit the command.

SQL2517W

The restored database was migrated to the current release.

Cause: The database being restored was backed up using a previous release of DB2. The RESTORE utility migrated it to the format of the current release.

The user who submitted the RESTORE command is given DBADM authority for the database. If there are other users, the database administrator with DBADM authority must grant authorization privileges to the specific database users.

The RESTORE utility completes successfully.

Action: No action is required.

SQL2518N

The RESTORE was not successful. An I/O error occurred while trying to restore the database configuration file.

Cause: The database configuration file could not be restored due to an I/O error.

The utility stops processing.

Action: Determine whether the I/O error can be corrected. Resubmit the command.

SQL2519N

The database was restored, but the restored database was not migrated to the current release. Error "<sqlcode>" with tokens "<tokens>" is returned.

Cause: The backup image contains a previous release of the database. After the database files were restored, an attempt was made to migrate the database to the current release. The migration failed.

The utility finishes processing but the database was not migrated.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and submit the Migrate command before using the database.

SQL2520W

The database was restored. The backup version of the configuration file was used.

Cause: When a database is restored, the current database configuration file is not always replaced with the backup version. The current database configuration file was not usable. It has been replaced by the backup version.

The utility completed successfully.

Action: Some values in the database configuration file may be different after the restore process. Check that the configuration parameters are set to the desired values.

SQL2521W

The database was restored, but an I/O error "<code>" occurred while deleting the .BRG file.

Cause: The restore process ran successfully. The .BRG file was not deleted due to an I/O error.

The .BRG file is used to determine whether a Restore operation has terminated abnormally. The .BRG file resides in the local database directory of the database in question and consists of a filename equivalent to the database token and filetype of .BRG.

The utility completed successfully.

Action: Try to delete the .BRG file. If you are unsuccessful, contact your technical service representative.

SQL2522N

More than one backup file matches the timestamp value provided for the backed up database image.

Cause: The backup image file filename consists of a database alias and a timestamp component. A filename is constructed from the source database alias and timestamp parameters provided in the Database Restore call. A portion of a timestamp may have been specified so that a match with more than one backup image filename was found.

Action: Resubmit the operation by specifying a timestamp with sufficient timestamp components to result in a match to a unique backup file.

SQL2523W

Warning! Restoring to an existing database that is different from the database on the backup image, but have matching names. The target database will be overwritten by the backup version. The Roll Forward recovery logs associated with the target database will be deleted.

Cause: The database alias and name of the target database are the same as the backup image database alias and name. The database seeds are not the same indicating that the databases are not the same. The

target database will be overwritten by the backup version. The Roll Forward recovery logs associated with the target database will be deleted. The current configuration file will be overwritten with the backup version.

Action: Return to the utility with the *callerac* parameter indicating processing to continue or end.

SQL2524W

Warning! Restoring to an existing database which appears to be the same but the alias "<dbase>" of the existing database does not match the alias "<dbase>" of the backup image. The target database will be overwritten by the backup version.

Cause: The database seeds of the target database and the database image are the same, indicating these are the same databases, the database names are the same, but the database aliases are not the same. The target database will be overwritten by the backup version.

Action: Return to the utility with the *callerac* parameter indicating processing to continue or end.

SQL2525W

Warning! Restoring to an existing database that is different from the database on the backup image, and the alias "<dbase>" of the existing database does not match the alias "<dbase>" of the backup image, but the database names are the same. The target database will be overwritten by the backup version. The Roll Forward recovery logs associated with the target database will be deleted.

Cause: The database aliases of the target database and database image are not the same, the database names are the same, and the database seeds are not the same, indicating these are different databases. The target database will be overwritten by the backup version. The Roll Forward recovery logs associated with the target database will be deleted. The current configuration file will be overwritten with the backup version.

Action: Return to the utility with the *callerac* parameter indicating processing to continue or end.

SQL2526W SQL2530N

```
┌─ SQL2526W ──────────────────────────┐
│ Warning!  Restoring to an existing database that │
│ is the same as the backup image database, and │
│ the alias names are the same, but the database │
│ name "<dbname>" of the existing database does │
│ not match the database name "<dbname>" of │
│ the backup image.  The target database will be │
│ overwritten by the backup version. │
└──────────────────────────────────────┘
```

Cause: The database aliases of the target database and database image are the same, and the database seeds are the same, however the database names are different. These are the same databases. The target database will be overwritten by the backup version.

Action: Return to the utility with the *callerac* parameter indicating processing to continue or end.

```
┌─ SQL2527W ──────────────────────────┐
│ Warning!  Restoring to an existing database that │
│ is different from the backup image database, │
│ and the database name "<dbname>" of the │
│ existing database does not match the database │
│ name "<dbname>" of the backup image, but the │
│ alias names are the same.  The target database │
│ will be overwritten by the backup version.  The │
│ Roll Forward recovery logs associated with the │
│ target database will be deleted. │
└──────────────────────────────────────┘
```

Cause: The database aliases of the target database and database image are the same, the database names are not the same, and the database seeds are not the same, indicating these are different databases. The target database will be overwritten by the backup version. The Roll Forward recovery logs associated with the target database will be deleted. The current configuration file will be overwritten with the backup version.

Action: Return to the utility with the *callerac* parameter indicating processing to continue or end.

```
┌─ SQL2528W ──────────────────────────┐
│ Warning!  Restoring to an existing database that │
│ is the same as the backup image database, but │
│ the alias name "<dbase>" of the existing │
│ database does not match the alias "<dbase>" of │
│ backup image, and the database name │
│ "<dbname>" of the existing database does not │
│ match the database name "<dbname>" of the │
│ backup image.  The target database will be │
│ overwritten by the backup version. │
└──────────────────────────────────────┘
```

Cause: The database aliases of the target database and database image are not the same, the database names are not the same, and the database seeds are the same, indicating these are the same databases. The current database will be overwritten by the backup version.

Action: Return to the utility with the *callerac* parameter indicating processing to continue or end.

```
┌─ SQL2529W ──────────────────────────┐
│ Warning!  Restoring to an existing database that │
│ is different from the backup image database, │
│ and the alias name "<dbase>" of the existing │
│ database does not match the alias name │
│ "<dbase>" of the backup image, and the │
│ database name "<dbname>" of the existing │
│ database does not match the database name │
│ "<dbname>" of the backup image.  The target │
│ database will be overwritten by the backup │
│ version.  The Roll Forward recovery logs │
│ associated with the target database will be │
│ deleted. │
└──────────────────────────────────────┘
```

Cause: The database aliases of the target and database image are not the same, the database names are not the same, and the database seeds are not the same, indicating these are different databases. The current database will be overwritten by the backup version. The Roll Forward recovery logs associated with the target database will be deleted. The current configuration file will be overwritten with the backup version.

Action: Return to the utility with the *callerac* parameter indicating processing to continue or end.

```
┌─ SQL2530N ──────────────────────────┐
│ The backup image is corrupted.  Restore of │
│ database from this backup image is not │
│ possible. │
└──────────────────────────────────────┘
```

Cause: The backup image being restored is corrupted making the database image non-restorable.

Action: Discard the backup image as it is unusable. Restore from a previous backup if possible.

SQL2531N

The backup image selected for restore is not a valid database backup image.

Cause: The file selected for restore is not a valid backup image. Either the file selected has become corrupted or a backup tape is not at the correct position.

Action: Determine the location of the correct backup image file, and resubmit the Restore command.

SQL2532N

The backup or unload file contains the backup or unload image of database "<dbalias>" taken at timestamp "<timestamp>". This is not the backup or unload image requested.

Cause: The file selected for restore does not contain the requested backup or unload image. The image is for a different database than that requested.

Action: If using tape, ensure that the correct tape is mounted. If a restore or load is being done from disk, then the file must have been renamed. Rename the file to the correct filename with a match on database name and timestamp. Resubmit the command after taking the appropriate action.

SQL2533W

Warning! The backup or unload file on device "<device>" contains image of database "<database>" taken at timestamp "<timestamp>". This is not the backup or unload image requested.

Cause: The backup or unload image read from the tape position contains a media header that does not match the header of the image of the first file of the backup or unload file sequence.

Action: Ensure that the tape is positioned at the correct backup or unload, then return to the utility with the *callerac* parameter indicating if processing continues.

SQL2534W

Warning! The media on device "<device>" is not positioned at a valid backup media header.

Cause: The data read from the tape position does not contain a valid backup media header.

Action: Ensure that the tape is positioned at the correct position, then return to the utility with the *callerac* parameter indicating if processing continues.

SQL2535W

Warning! End of media on device "<device>" has been reached. Please mount the next source media.

Cause: The end of tape has been reached and there is still more data to be processed. The remainder of the backup or load source exists on another tape or on several more tapes.

Action: Mount the next tape in sequence that contains the source image and resubmit the Restore or Load command with the *callerac* parameter set to continue or end.

SQL2536W

Warning! The backup image on device <device> contains an incorrect sequence number. Sequence number <number> is expected.

Cause: The tape is positioned on a backup image file that is out of sequence. The tape containing the backup image must be positioned to the file of sequence number "<sequence>" of the backup image.

Action: Position the tape containing the backup image to the correct file and resubmit the Restore command with the *callerac* parameter set to continue or end.

SQL2537N

Roll forward is required following the Restore.

Cause: SQLUD_NOROLLFWD was specified for the *rst_type* parameter of the Database Restore utility, indicating that a roll forward is not required to make the restored database usable. The database to be restored was backed up in online mode and a roll forward operation is necessary to make the database usable.

Action: Resubmit the Database Restore command without specifying SQLUD_NOROLLFWD for the *rst_type* parameter.

SQL2538N   SQL2544N

SQL2538N

Unexpected End of File of Backup Image reached on media "<media>".

Cause: While reading and restoring from the Backup Image file, an end of file condition was reached unexpectedly. The backup image is unusable and the restore operation terminates.

Action: Resubmit the Database Restore command with a usable backup image file.

SQL2539W

Warning! Restoring to an existing database that is the same as the backup image database. The database files will be deleted.

Cause: The database aliases, names and seeds of the target database and database image are the same, indicating these are the same databases. The current database will be overwritten by the backup version.

Action: Return to the utility with the *callerac* parameter indicating processing to continue or end.

SQL2540W

Restore is successful, however a warning "<warn>" was encountered during Database Restore while processing in No Interrupt mode.

Cause: The Database Restore utility was invoked in No Interrupt mode, i.e. SQLUB_NO_INTERRUPT or SQLUD_NO_INTERRUPT was specified. During the processing a warning was encountered but not returned at the time it was encountered. The Restore has completed successfully and the warning message found is shown at the completion in this message.

Action: Ensure that the action that caused this warning to be generated has not resulted in a condition not wanted.

SQL2541W

Restore is successful, however the file containing the backup image could not be closed.

Cause: The Database Restore was successful, but the file containing the backup image could not be closed.

Action: Attempt to close the file containing the backup image.

SQL2542N

No match for a database image file was found based on the source database alias and timestamp provided.

Cause: The backup image file filename consists of a database alias and a timestamp component. A filename is constructed from the source database alias and timestamp parameters provided in the Database Restore call. No filename existed in the source directory that matched based on the source database alias and timestamp provided.

Action: Ensure that the database backup image resides on the media source. Resubmit the operation by specifying a correct timestamp to result in a match.

SQL2543N

The target directory specified for the database is not valid.

Cause: The application calling the Restore utility has supplied a target directory for the new database to be created. Either this directory does not exist, or is not a valid directory for database creation. An invalid directory for database creation would be a directory of greater than 255 characters in length.

Action: Reissue the Backup or Restore command with a valid target directory.

SQL2544N

The directory where the database is being restored has become full.

Cause: While the database was being restored the directory to which it was being restored had become full. The database being restored is unusable. The Restore terminates and if the database being restored is a new database, then it is deleted.

Action: Free up sufficient space on the directory for the database, and reissue the Restore, or if restoring to a new database supply a directory with sufficient space to contain the database.

SQL2545W

Warning! The backup image on the ADSM server is currently stored on mountable media. The time required to make it available is unknown.

Cause: The backup image is not immediately accessible by the ADSM server. The restore process can continue and make the request to the server to retrieve the data. The time required is unknown.

Action: Return to the utility with the callerac parameter indicating processing to continue or end.

SQL2546N

The image on media "<media>" is not the first in sequence from the backup or copy.

Cause: During restore or load recovery, the first image from the backup or copy must be processed first. The image found on the media is not the first in sequence.

The utility waits for a response to continue.

Action: Mount the media with the correct backup or copy image and return to the utility with the correct caller action parameter to indicate if processing should continue or terminate.

SQL2547N

The database was not restored because the backup image is from a previous release and was an online backup.

Cause: The physical log file formats have changed between these releases, making rollforward impossible.

Action: Restore the database using the version of DB2 used to create the database and rollforward to the end of logs. Take an off-line full database backup at this time. This new backup image will be restorable on the new release of DB2.

SQL2548N

Database codepage indicated in backup image differs from that of the current on-disk database. The Restore operation has failed.

Cause: The database contained in the backup image contains data stored in a codepage different from that of the database into which the data is being restored.

This problem can be caused by one of the following situations:

1. The database you are restoring into has a different codepage than that of the database in the backup image.
2. The user is restoring into a new database from a session with a different codepage than the backup.
3. The backup image is corrupt, and contains invalid character set information.

Action:

1. If restoring over an existing database, make sure that the codepage of the existing database matches that of the backup image.
2. If restoring to a new database, make sure that the codepage of the session from which the restore command was issued is the same as that of the backup image.
3. Contact IBM Service.

SQL2560N

The target database is not identical to the source database for a restore from a table space level backup.

Cause: For a restore from a table space level backup, the target database must be either the original database from which the backup is taken or a new database.

Action: Specify the correct target database and resubmit the utility command.

SQL2561W

Warning! Restoring from a table space level backup to a database which does not exist. A database with the same attributes in the backup will be created.

Cause: For a restore from a table space level backup, the target database must have the same attributes (database name, alias and seed) as the source database. If the database does not already exist, it will be created.

Action: Return to the utility with the *callerac* parameter indicating processing to continue or end.

SQL2562N SQL2650N

SQL2562N

Table space level restore from a full database backup is not allowed.

Cause: For a restore from a full database level backup, the restore type cannot be table space level.

Action: Specify a correct restore type or use a correct backup image and resubmit the utility command.

SQL2563W

The restore process has completed successfully, but one or more table spaces from the backup are not restored because of a container access error.

Cause: The restore process has completed successfully. One or more table spaces that are in the backup are not restored because an error was encountered while accessing the containers of the table spaces. If the table spaces have been dropped since the backup is taken, no further action is necessary.

Action: Use the query table space function to check the states of the table spaces. If a table space is in the "storage definition pending" state, the storage definition for the table space must be corrected for restore to complete successfully. See the Administration Guide for details to recovery the table space.

SQL2565N

The option specified for RESTORE is not allowed with the backup image provided.

Cause: The database contained in the backup image is an existing database and does not match the database to be restored. The option you have specified for restore requires that the database being restored to is new or the same as in the backup image.

The utility stops processing.

Action: Resubmit the command with the correct database name.

SQL2600W

The input parameter pointer to the authorization block is invalid or the block's size is incorrect.

Cause: The pointer to the authorization structure parameter is NULL, the pointer to the authorization structure points to an area that is smaller than the length indicated in the structure length field, or the authorization structure length field is not set to the correct value.

The command cannot be processed.

Action: Correct the value of the input parameter and resubmit the command.

SQL2650N

An invalid parameter was passed to the asynchronous read log API. Reason Code = "<reason-code>".

Cause: An invalid parameter was passed to the asynchronous read log API. The possible reason codes are as follows:

01   Invalid action specified.
02   The starting log sequence number is greater than or equal to the current database active log sequence number.
03   The log sequence number range bounded by the starting and ending log sequence numbers is not large enough to span a log record.
04   The starting log sequence number does not represent the start of an actual log record.
05   The starting log sequence number's location could not be determined.
06   The ending log sequence number is less than or equal to the starting log sequence number.
07   The buffer is not valid for the indicated size.
08   The buffer is not of sufficient size to store a log record.
09   A pointer is not valid.

Action: For reason code 01, ensure that the action is either SQLU_RLOG_QUERY, SQLU_RLOG_READ, or SQLU_RLOG_READ_SINGLE.

For reason code 02, ensure that the starting log sequence number is less than the current active log sequence number returned in the read log information structure.

For reason code 03, ensure that the ending log sequence number is sufficiently larger than the starting log sequence number.

For reason code 04, ensure that the starting log sequence number is either the initial log sequence

number or one byte larger than the last read log sequence number returned in the read log information structure.

For reason code 05, ensure that the starting log sequence number resides in a log extent in the database log file path.

For reason code 06, ensure that the ending log sequence number is greater than the starting log sequence number.

For reason code 07, ensure that the buffer has been allocated and is of the size indicated by the log buffer size parameter.

For reason code 08, increase the size of the allocated buffer.

For reason code 09, ensure that memory has been correctly allocated and the pointers properly initialized.

SQL2651N

The log records associated with the database can not be asynchronously read.

Cause: The asynchronous read log API was used against a connected database which does not have LOG RETAIN or USER EXITS ON. Only databases which are forward recoverable may have their associated logs read.

Action: Update the database configuration for the database, identified to the asynchronous read log API, turning LOG RETAIN and/or USER EXITS ON.

SQL2652N

Insufficient memory to run the asynchronous log reader.

Cause: The allocation of the internal buffers used by the asynchronous read log API has failed.

Action: Increase the amount of available memory available to the process (increase real or virtual memory or remove unnecessary background processes) or decrease the log sequence number range provided to the asynchronous read log API.

SQL2653W

A Restore, Forward or Crash Recovery may have reused log sequence number ranges. Reason code "<reason-code>".

Cause: A Restore, Forward or Crash Recovery may have reused log sequence number ranges. The possible reason codes are as follows:

- 01 - Restore has taken place.
- 02 - Forward Recovery (ROLLFORWARD) has taken place.
- 03 - Crash Recovery has taken place.

Action: No action is required.

SQL2654W

The database log has been asynchronously read to the end of the current active log.

Cause: All log records in the database active log have been read by the asynchronous log reader.

Action: No action is required.

SQL2655N

The Asynchronous Log Reader encountered log file "<name>" which is not associated with the database being asynchronously read.

Cause: The Asynchronous Log Reader attempted to read log records from the specified log file. The specified log file is not associated with the database being asynchronously read.

Action: Remove this log file from the database log directory. Move the correct log file into the database log directory and resubmit the call to the Asynchronous Read Log API.

SQL2656N

The Asynchronous Log Reader has encountered log file "<name>" which cannot be read.

Cause: The Asynchronous Log Reader attempted to read log records from the specified log file. The specified log file has been corrupted and cannot be read.

Action: Increase the start read log sequence and resubmit the call to the Asynchronous Read Log API to start reading after the specified log file.

SQL2657N SQL3006C

SQL2657N

The Asynchronous Log Reader requires log file "<name>" which does not currently reside in the database log directory.

Cause: The Asynchronous Log Reader requires log records from the specified log file. The specified log file does not currently reside in the database log directory.

Action: Move the specified log file into the database log directory of the database being asynchronously read. Resubmit the call to the Asynchronous Read Log API.

SQL3001C

An I/O error (reason = "<reason>") occurred while opening the output file.

Cause: A system I/O error occurred while opening the output file.

The command cannot be processed. No data is processed.

Action: For IMPORT/LOAD, verify that the output file exists. For EXPORT/UNLOAD, verify that the output media has enough free space. Resubmit the command with a valid output file name, including the correct path. Look in the message file for additional information.

SQL3002C

An I/O error occurred while writing to the output data file.

Cause: A system I/O error occurred while writing to the output data file. The output may be incomplete or the disk may be full.

The utility stops processing.

Action: If the output data file is incomplete, erase it and resubmit the command.

SQL3003C

An I/O error occurred while closing the output data file.

Cause: A system I/O error occurred while closing the output data file.

The file is not closed.

Action: If the output data file is incomplete, erase it and resubmit the command.

SQL3004N

The filetype parameter is not valid.

Cause: The *filetype* parameter must be DEL, ASC, IXF, WSF, or DB2CS for the command.

The *filetype* parameter must be DEL, IXF, or WSF for the EXPORT command.

The *filetype* parameter must be ASC, DEL, IXF, or DB2CS for the LOAD command.

The *filetype* parameter must be DB2CS for the UNLOAD command.

The command cannot be processed.

Action: Resubmit the command with a valid *filetype* parameter.

SQL3005N

Processing was interrupted.

Cause: An interrupt was received during processing. The user may have pressed the interrupt key sequence.

The utility stops processing. Any database updates not committed are rolled back.

Action: Resubmit the command. If doing an import, see *Command Reference* for information on how to use commitcount and restartcount parameters. If doing a load, see *Command Reference* for information on how to restart the load.

SQL3006C

An I/O error occurred while opening the message file.

Cause: A system I/O error occurred while opening the message file. This error can refer to a problem on either the client or the server.

The command cannot be processed.

Action: Resubmit the command with a valid message file name, including the correct path.

SQL3007C

An I/O error occurred while writing to the message file.

Cause: A system I/O error occurred while writing to the message file.

Processing may not have finished.

Action: If the message file is incomplete, erase it and resubmit the command.

SQL3008N

The utility encountered an error "<error>" while connecting to the database.

Cause: The IMPORT or EXPORT utility could not connect to the database.

No data is imported or exported.

Action: Look at the error number in the message for more information. Make changes and resubmit the command.

SQL3009N

The tcolstrg parameter is not valid.

Cause: The *tcolstrg* parameter in the command is not valid. The *tcolstrg* pointer may be incorrect. The structure that the *tcolstrg* points to may be incorrect. The *tcolstrg* structure may contain characters that are not valid.

The command cannot be processed.

Action: Verify the *tcolstrg* pointer and the structure it points to. Resubmit the command with a valid *tcolstrg*.

SQL3010N

The dcoldata parameter is not valid.

Cause: The *dcoldata* parameter in the command is not valid. The *dcoldata* pointer may be incorrect. The structure that the *dcoldata* points to may be incorrect. The *dcoldata* structure may contain characters that are not valid.

The command cannot be processed.

Action: Examine the *dcoldata* pointer and the structure it points to. Resubmit this command with a valid *dcoldata*.

SQL3011C

There is not enough storage to process the command.

Cause: A memory allocation error occurred. Not enough memory available to process the command or there was an error freeing storage.

The command cannot be processed.

Action: Stop the application. Possible solutions include:

- Verify that your system has sufficient real and virtual memory.
- Remove background processes.
- Increase the *util_heap_sz* for the database.
- Decrease the size of buffers to use for the LOAD/UNLOAD or the number of buffers for UNLOAD.
- The *util_heap_sz* is shared by the UNLOAD, LOAD, BACKUP and RESTORE utilities, so have fewer concurrent instances of these utilities running.

SQL3012C

A system error occurred.

Cause: An operating system error occurred.

The command cannot be processed.

Action: Check your message file, correct the problem and resubmit the command.

SQL3013N

The length of the filetmod is out of the acceptable range. It should be greater than or equal to zero and less than or equal to 8000.

Cause: The filetmod specified is outside the acceptable range (greater than or equal to zero and less than or equal to 8000).

The command cannot be processed.

Action: Examine the *filetmod* pointer and the structure it points to. Resubmit the command with a valid *filetmod*.

SQL3014C SQL3021N

SQL3014C

An I/O error occurred while closing the message file.

Cause: A system I/O error occurred while closing the message file.

The message file is not closed.

Action: If the message file is incomplete, resubmit the command.

SQL3015N

An SQL error "<sqlcode>" occurred during processing.

Cause: An SQL error occurred during the utility call.

The utility stops processing.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL3016N

An unexpected keyword "<keyword>" was found in the filetmod parameter for the filetype.

Cause: A keyword that does not apply to the filetype for the utility was found in the filetype modifier (the *filetmod* parameter or the phrase following *MODIFIED BY* in a CLP command).

The utility stops processing.

Action: Remove the filetype modifier or specify at least one valid keyword in the filetype modifier. See the Command Reference for more information on filetype modifiers.

SQL3017N

A delimiter is not valid or is used more than once.

Cause: For Delimited ASCII (DEL) files, one of two errors occurred:

* The character specified for either the column delimiter, character string delimiter, or decimal point character is not valid.
* The same character is specified for more than one of the preceding items.

The command cannot be processed.

Action: Examine the specified delimiters for validity and uniqueness. Resubmit the command with valid delimiter overrides.

SQL3018N

The period was specified as a character string delimiter.

Cause: For Delimited ASCII (DEL), the period cannot be specified as the character string delimiter.

The command cannot be processed.

Action: Resubmit the command with valid delimiter overrides.

SQL3019N

No tcolstrg parameter was specified in the command.

Cause: No *tcolstrg* parameter is specified for this utility call. This parameter is required.

The command cannot be processed.

Action: Resubmit the command with a *tcolstrg* parameter.

SQL3020N

The user does not have the authority to run the specified Export command.

Cause: The user attempted to export data without the appropriate authorization (either the SYSADM or DBADM authority), or the CONTROL or SELECT privilege for each table involved in the export.

The export action is not performed.

Action: Request appropriate authorization before exporting data from the tables.

SQL3021N

The user does not have the authority to run the specified Import command on table "<name>".

Cause: The user attempted to import data without the appropriate authorization for the specified option and table. To execute this utility using the INSERT option, the user must have one or more of the following:

* SYSADM or DBADM authority
* CONTROL privilege for the table or view
* INSERT and SELECT privilege for the table or view.

To execute this utility using the CREATE option or the REPLACE_CREATE option when the table already exists, the user must also have one or more of the following:

- SYSADM or DBADM authority
- CREATETAB privilege.

To execute this utility using the REPLACE option or the REPLACE_CREATE option when the table already exists, the user must also have one or more of the following:

- SYSADM or DBADM authority
- CONTROL privilege for the table or view.

The import operation is not performed.

Action: Secure appropriate authorization before importing data to tables.

SQL3022N

An SQL error "<sqlcode>" occurred while processing the SELECT string in the tcolstrg parameter.

Cause: In IMPORT or EXPORT, during processing of the SELECT string from the *tcolstrg* structure, an SQL error occurred.

The command cannot be processed.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL3023N

The database name parameter is not valid.

Cause: The database name parameter is not valid. See the SQLERRD[0] field in the SQLCA for more information.

The database name must contain 1 to 8 characters and the characters must be from the database manager base character set.

The command cannot be processed.

Action: Resubmit the statement with a valid database name parameter.

SQL3025N

A parameter specifying a filename or path is not valid.

Cause: The parameter contains a path, directory, or file name that is not valid. For IMPORT and EXPORT, verify the *datafile* parameter. For LOAD and UNLOAD, verify each entry in the parameters of type *sqlu_media_list*: *datafile* should contain valid filenames; *lobpaths copytarget*, and *workdirectory* should contain valid paths on the server including the final separator and a null terminator. The pointer to lobpaths, copytarget and workdirectory must be valid or null. The pointer to the targets in the these structures, must be valid. Check that sessions and media_type are specified correctly. Also, if the lobpaths parameter is specified, check that the media_type is the same as that specified for the datafile structure.

If the filetype is IXF, the length of the file name may be too long.

The command cannot be processed.

Action: Resubmit the statement with a valid parameter.

SQL3026N

The msgfile or remotefile parameter is not valid.

Cause: For IMPORT or EXPORT, the *msgfile* parameter contains a path, directory, or file name that is not valid. For LOAD or UNLOAD, either the *msgfile* parameter contains a path, directory, or file name that is not valid on the client or the *remotefile* parameter contains a path, directory, or file name that is not valid on the server. If the database to which the application is connected is a remote database, the msgfile must be fully qualified. For a local database the utility will attempt to fully qualify the msgfile if it is not already fully qualified. Also verify that the pointer to the msgfile is valid.

The command cannot be processed.

Action: Resubmit the statement with a valid *msgfile* and/or *remotefile* parameter.

SQL3028N SQL3035N

Cause: The export method indicator must be either N for Names or D for Default.

The command cannot be processed.

Action: Resubmit the statement with a valid method indicator.

Cause: The *filetype* parameter was either not specified or null. The system does not know which format to use for the data file.

The command cannot be processed.

Action: Resubmit the statement with a valid *filetype*.

Cause: A system I/O error occurred while opening the input file. This error can refer to a problem on either the client or the server.

The command cannot be processed.

Action: Resubmit the command with a valid input file, including the correct path. Look in the message file for additional information.

Cause: A system I/O error occurred while reading from the input file. This error can refer to a problem on either the client or the server.

The command cannot be processed.

Action: Verify that the input file is readable.

Cause: For the IXF and WSF filetypes, the import method indicator must be either N for Names, P for Positional, or D for Default.

The command cannot be processed.

Action: Resubmit the command with a valid method indicator.

Cause: For IMPORT, the *tcolstrg* parameter does not contain the keyword INSERT, REPLACE, CREATE, INSERT_UPDATE, or REPLACE_CREATE. For LOAD, the *tcolstrg* parameter does not contain the keyword INSERT, REPLACE, or RESTART. The keyword must be followed by at least one blank.

The command cannot be processed.

Action: Resubmit the command with a valid *tcolstrg* parameter.

Cause: The INTO keyword is missing or spelled incorrectly. INTO must be followed by at least one blank.

The command cannot be processed.

Action: Resubmit the command with a valid *tcolstrg* parameter.

Cause: For IMPORT, the *tablename* in the *tcolstrg* is not valid. For LOAD, the *tablename* or the *exception tablename* in the *tcolstrg* is not valid.

The command cannot be processed.

Action: Resubmit the command with a valid *tablename*.

SQL3036N

The tcolumn-list in the target specification does not have a closed parenthesis.

Cause: The *tcolumn-list* must be delimited by parentheses. The list was not terminated by the closed parenthesis.

The command cannot be processed.

Action: Resubmit the statement with a valid, complete column list in the *tcolstrg* parameter.

SQL3037N

An SQL error "<sqlcode>" occurred during Import processing.

Cause: An SQL error occurred during processing of the *tcolstrg* parameter.

The command cannot be processed.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL3038N

The tcolstrg parameter contains unexpected characters.

Cause: For IMPORT, there are characters other than blanks after the closing parenthesis of the column list in the *tcolstrg* parameter. For LOAD, there are characters other than blanks after the closing parenthesis of the column list and/or the exception tablename in the *tcolstrg* parameter.

The command cannot be processed.

Action: Resubmit the command with a valid *tcolstrg* parameter.

SQL3040N

Unable to use the lobpath/lobfile parameters as specified. Reason code: "<reason-code>".

Cause: The utility is unable to use the lobpath or lobfile parameters as specified for the reason given by "<reason-code>". The codes are listed below.

Action: Given for each reason code:

1 The lobpath is not a valid sqlu_media_list or the values provided are not valid. The media_type must be SQLU_LOCAL_MEDIA and all pathnames must be terminated with a valid path separator.

2 The lobfile is not a valid sqlu_media_list or the values provided are not valid. The media_type must be SQLU_LOCAL_MEDIA.

3 There are not enough lobpath names provided. For export, there is not enough space on the paths provided to hold all the lobs.

4 There are not enough lobfile names provided. For export, there are more lobs than the number of lobfile names provided multiplied by SQLU_MAX_SESSIONS.

5 The combination of a lobpath name and lobfile name would exceed the maximum size of a lobfile name (255 bytes).

6 An error occurred while trying to access the file.

SQL3045N

A dcolumn position in the dcoldata parameter is less than one or greater than the maximum number of columns (1024) in a Delimited ASCII file.

Cause: The *dcolumn* position is less than 1 or greater than the maximum number of columns (1024) in a delimited file.

The command cannot be processed.

Action: Resubmit the command with valid *dcolumn* positions.

SQL3046N

The number of columns in the dcoldata parameter is less than one.

Cause: For *dcoldata* methods other than Default, the number of columns specified must be a positive number (greater than 0).

The command cannot be processed.

Action: Resubmit the command with a valid number of columns in the *dcoldata* parameter.

SQL3047N   SQL3053N

SQL3047N

The import method specified in dcoldata is not valid for Delimited ASCII files. It must be either 'P' or 'D'.

Cause: The only valid import methods for Delimited ASCII files are P for Positions or D for Default.

The command cannot be processed.

Action: Resubmit the command with a valid import method.

SQL3048N

There are fewer columns specified from the input file than database columns, but one of the database columns is not nullable.

Cause: When fewer columns are specified in the *dcoldata* list than are specified for the target table, the values for the missing input columns are loaded as nulls. Because the columns in the target table that correspond to one or more of these input columns are not nullable, null cannot be inserted.

The file cannot be loaded.

Action: Define a new table that has the same number of columns as the input file or that has nullable columns. Resubmit the command.

SQL3049N

The data type "<type>" of the database column "<name>" is not compatible with this format file, but the database column is not nullable.

Cause: The database column type is not valid for this format file. The database column is not nullable so the utility is ended.

The utility stops processing. No data is loaded.

Action: Redefine the database table so the columns are compatible with the columns loaded from the file.

SQL3050W

Conversions on the data will be made between the IXF file code page and the application code page.

Cause: If an IMPORT of an IXF data file has been issued and the code page of the character data in the IXF file is different from the code page of the application invoking the import operation, a conversion will be made from the code page of the data file to the code page of the application and the operation will continue.

If a LOAD of an IXF data file has been issued and the code page of the character data in the IXF file is different from the database code page, a conversion will be made from the code page of the data file to the code page of the database and the operation will continue.

Action: If you do not want a conversion to be performed, invoke the utility using the FORCEIN option; otherwise no action is required.

SQL3051W

The data to be loaded into "<column_name>" has been loaded, but conversion from the IXF file code page to the application code page has not been performed.

Cause: The data that was loaded into the CLOB or DBCLOB column is stored in a separate file and no conversions have been performed on it.

To properly load the data, call the utility from an application that has the same code page as the IXF file.

Action: This is a warning only.

SQL3053N

There are more than 8191 rows to be exported to a worksheet formatted file.

Cause: The maximum number of rows that can be placed in a worksheet format file (WSF) is 8191.

The EXPORT utility stops processing after placing 8191 rows in the file.

Action: To prevent this error, use a SELECT statement that results in fewer rows to export, and resubmit the command.

SQL3054N

The input file is not a valid PC/IXF file. The file is too short to contain a valid H record.

Cause: The end of the file was reached before the expected end of the first record. The file may not be a PC/IXF file.

The IMPORT utility stops processing. No data is imported.

Action: Verify that the input file is correct.

SQL3055N

The input file is not a valid PC/IXF file. The length field in the first record cannot be converted to a numeric value.

Cause: The value in the length field of the first record is not the ASCII representation of a number. The file may not be a PC/IXF file.

The utility stops processing. No data is loaded.

Action: Verify that the input file is correct.

SQL3056N

The input file is not a valid PC/IXF file. The value in the length field of the H record is too small.

Cause: The value in the length field of the H record is not large enough for a valid H record. The file may not be a PC/IXF file.

The utility stops processing. No data is loaded.

Action: Verify that the input file is correct.

SQL3057N

The input file is not a valid PC/IXF file. The type field in the first record is not H.

Cause: The type field in the first record is not H. The first record is not a valid H record. The file may not be a PC/IXF file.

The utility stops processing. No data is loaded.

Action: Verify that the input file is correct.

SQL3058N

The identifier field in the H record is not IXF.

Cause: The identifier field in the H record does not identify the file as a PC/IXF file.

The utility stops processing. No data is loaded.

Action: Examine the identifier field in the H record.

SQL3059N

The version field in the H record is not valid.

Cause: The version field in the H record contains a value that is not valid.

The utility stops processing. No data is loaded.

Action: Examine the version field in the H record.

SQL3060N

The HCNT field in the H record cannot be converted to a numeric value or the value is out of range.

Cause: The Heading-record-count field in the H record is not an ASCII representation of a number or is a number that is not valid for this field.

The utility stops processing. No data is loaded.

Action: Examine the HCNT field in the H record.

SQL3061N

The single-byte and/or double-byte code page field in the H record cannot be converted to a numeric value or the value is out of range.

Cause: The single-byte and/or double-byte code page field in the H record is not an ASCII representation of a number or is a number that is not valid for this field.

The utility stops processing. No data is loaded.

Action: Examine the single-byte and double-byte code page field in the H record, change them to appropriate values as specified in the *API Reference* and resubmit the command.

SQL3062N   SQL3068N

SQL3062N

The double-byte code page field in the H record cannot be converted to a numeric value or the value is out of range.

Cause: The double-byte code page field in the H record is not an ASCII representation of a number or is a number that is not valid for this field.

The utility stops processing. No data is loaded.

Action: Examine the double-byte code page field in the H record, change it to an appropriate value as specified in the *API Reference* and resubmit the command.

SQL3063N

The single-byte code page value "<value 1>" in the H record is not compatible with the single-byte code page value "<value 2>" for the application. The forcein option was not specified.

Cause: The single-byte code page value in the H record is not compatible with the application code page value. When the forcein option is not used, the data cannot be loaded unless conversion from value 1 to value 2 is supported.

The utility stops processing. No data is loaded.

Action: To load this data, resubmit the command with the forcein option.

SQL3064N

The double-byte code page value "<value 1>" in the H record is not compatible with the double-byte code page value "<value 2>" for the application. The forcein option was not specified.

Cause: The double-byte code page value in the H record is not compatible with the application code page value. When the forcein option is not used, the data cannot be loaded unless value 1 and value 2 are the same.

The utility stops processing. No data is loaded.

Action: To load data where the double-byte code page values do not match, resubmit the command with the forcein option.

SQL3065C

The value for the application code page cannot be determined.

Cause: The system encountered an error while determining the code page of the application.

The utility stops processing. No data is loaded or unloaded.

Action: Contact your technical service representative.

SQL3066N

While reading or looking for the T record, the end of the file was reached.

Cause: The end of the file was reached while the system was looking for the T record or while it was reading the T record.

The utility stops processing. No data is loaded.

Action: Examine the T record of the input file. If the PC/IXF file was copied from one medium to another, compare the copy with the original or repeat the copy process.

SQL3067N

The length field in the T record cannot be converted to a numeric value.

Cause: The length field in the T record is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

Action: Examine the length field in the T record.

SQL3068N

The value in the length field of the T record is too small.

Cause: The value in the length field of the T record is not large enough so the T record is not valid.

The utility stops processing. No data is loaded.

Action: Examine the length field in the T record.

SQL3069N

The first non-A record following the H record is not a T record.

Cause: After the H record, the first record not an A record is also not a T record. The H record must be immediately followed by a T record but A records may exist before the T record.

The utility stops processing. No data is loaded.

Action: Examine the records following the H record.

SQL3070N

The length field in an A record is out of range.

Cause: The length field in an A record is a number that is not valid for this field.

The utility stops processing. No data is loaded.

Action: Examine the length field in the A records.

SQL3071N

The data-convention field in the T record is not C.

Cause: The data-convention field in the T record is a value other than C.

The utility stops processing. No data is loaded.

Action: Examine the data-convention field in the T record.

SQL3072N

The data-format field in the T record is not M.

Cause: The data-format field in the T record is a value other than M.

The utility stops processing. No data is loaded.

Action: Examine the data-format field in the T record.

SQL3073N

The machine-form field in the T record is not PCbbb (where b = blank).

Cause: The machine-form field in the T record is a value other than PC*bbb*, where each *b* is a blank.

The utility stops processing. No data is loaded.

Action: Examine the machine-form field in the T record.

SQL3074N

The data-location field in the T record is not I.

Cause: The data-location field in the T record is a value other than I.

The utility stops processing. No data is loaded.

Action: Examine the data-location field in the T record.

SQL3075N

The CCNT in the T record cannot be converted to a numeric value or the value is out of range.

Cause: The C-record count field in the T record is not an ASCII representation of a number or is a number that is not valid for this field.

The utility stops processing. No data is loaded.

Action: Examine the CCNT field in the T record.

SQL3076N

The name length field in the T record cannot be converted to a numeric value or the value is out of range.

Cause: The name length field in the T record is not an ASCII representation of a number or is a number that is not valid for this field.

The utility stops processing. No data is loaded.

Action: Examine the name length field in the T record.

SQL3077N

The number of C records specified in the CCNT field in the T record "<value>" exceeds the maximum allowed "<maximum>".

Cause: The value in the CCNT field of the T record is larger than the maximum allowed for the indicated release.

The utility stops processing. No data is loaded.

Action: Examine the CCNT field in the T record.

SQL3078N   SQL3085N

> **SQL3078N**
>
> **The length field in an A record cannot be converted to a numeric value.**

Cause: The length field in an A record is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

Action: Examine the length fields in the A records.

> **SQL3079N**
>
> **The length field in a C record cannot be converted to a numeric value.**

Cause: The length field in a C record is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

Action: Examine the length field in the C records.

> **SQL3080N**
>
> **The value in the length field of the C record is too small.**

Cause: The value in the length field of a C record is not large enough so the C record is not valid.

The utility stops processing. No data is loaded.

Action: Examine the length field in the C records.

> **SQL3081N**
>
> **Not enough C records were found.**

Cause: The number of C records found (in the correct position) is less than the number expected from the C-record-count (CCNT) in the T record.

The utility stops processing. No data is loaded.

Action: Examine the T and C records.

> **SQL3082N**
>
> **While reading or looking for a C record, the end of the file was reached.**

Cause: The end of the file was reached while the system was looking for a C record or while it was still reading a C record.

The utility stops processing. No data is loaded.

Action: Examine the C records in the input file. If the PC/IXF file was copied from one medium to another, compare the copy with the original or repeat the copy process.

> **SQL3083N**
>
> **The D-record-id field in the C record for column "<name>" cannot be converted to a numeric value.**

Cause: The D-record-id field in the C record for the indicated column is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

Action: Examine the D-record-id field in the C record.

> **SQL3084N**
>
> **The D-record-position field in the C record for column "<name>" cannot be converted to a numeric value.**

Cause: The D-record-position field in the C record for the indicated column is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

Action: Examine the D-record-position field in the C record.

> **SQL3085N**
>
> **The D-record-id and D-record-position fields in the C record for column "<name>" are out of range or inconsistent with the previous C record.**

Cause: The D-record-id or the D-record-position fields in the C record for the indicated column contain a value that is either out of the value range or is not correct relative to the previous C record.

The utility stops processing. No data is loaded.

Action: Examine the D-record-id and D-record-position fields in the C record.

SQL3086N

There was no source column specified to be loaded into database column "<name>" or the specified source column does not exist, but the database column is not nullable.

Cause: Either there was not a PC/IXF column specified to be exported into the indicated column or the specified PC/IXF source column does not exist. Nulls cannot be inserted because the database column is not nullable.

The utility stops processing. No data is loaded.

Action: Check the *dcoldata* parameter for erroneous names or positions or for fewer entries in the *dcoldata* parameter than the *tcolstrg* parameter states or implies.

SQL3087N

The source column specified to be loaded into database column "<name>" is not valid, but the database column is not nullable.

Cause: The PC/IXF column cannot be loaded into the indicated database column and the reason was given in a previous message in the message log. Nulls cannot be inserted because the database column is not nullable.

The utility stops processing. No data is loaded.

Action: Read the previous messages to understand why the column is not valid.

SQL3088N

The source column specified to be loaded into database column "<name>" is not compatible with the database column, but the database column is not nullable.

Cause: The source PC/IXF column is not compatible with the target database column. The column types or lengths may be incompatible. Nulls cannot be inserted because the database column is not nullable.

The utility stops processing. No data is loaded.

Action: Compare the columns in the source PC/IXF file with the columns in the database.

SQL3089N

A non-D record was found where a D record was expected.

Cause: Some record other than a D record was found where a D record was expected.

The utility stops processing. No data is loaded.

Action: Examine the D records in the file.

SQL3090N

The length field in a D record cannot be converted to a numeric value.

Cause: The length field in a D record is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

Action: Examine the length field in the D records.

SQL3091N

The length field in a D record is out of range.

Cause: The length field in a D record is a number that is not valid for this field.

The utility stops processing. No data is loaded.

Action: Examine the length field in the D records.

SQL3092N

The identifier field in a D record does not contain the expected value.

Cause: The identifier field in a D record is not valid. One or more D records may have been written out of sequence.

The utility stops processing. No data is loaded.

Action: Examine the identifier field in the D records.

SQL3093N

The input file is not a valid WSF file.

Cause: The first record in the worksheet format (WSF) file was not a beginning-of-file (BOF) record, or the version of the WSF file is not supported.

The IMPORT utility stops processing. No data is imported.

Action: Verify that the file is a valid WSF file and that the name was entered correctly.

SQL3094N   SQL3099N

SQL3094N

The input column "<name>" was not found, but the corresponding database column is not nullable.

Cause: The specified column was not in the input file. Because the corresponding database column is not nullable, data cannot be loaded into the column.

The utility stops processing. No data is loaded. The column names processed before the error occurred are in the database

Action: Verify that the input file contains the specified column names.

SQL3095N

The specified column position "<position>" is not in the valid range of 1 to 256.

Cause: A column position was specified that was not within the range of 1 to 256.

The utility stops processing. No data is loaded. The column names processed before the error occurred are in the database.

Action: Verify that the specified column position is within the range 1 to 256.

SQL3096N

The data type "<type>" of the database column "<name>" is not compatible with any WSF column type, but the database column is not nullable.

Cause: There is not any worksheet format (WSF) column type that is compatible with the indicated database column. The database column is not nullable so the IMPORT utility stops processing.

No data is imported.

Action: Redefine the database table so the columns are compatible with the columns being imported from the WSF file. Resubmit the command.

SQL3097N

The record length field in the WSF record is not valid for its record type.

Cause: The worksheet format (WSF) records have an expected fixed length or a variable-length range. The record does not contain the fixed length, or the variable length is out of range. Some damage occurred to the WSF file, or it was generated incorrectly, possibly with a level of the Lotus product not supported by the database manager.

The IMPORT utility stops processing.

Action: Regenerate the WSF file with a supported level of the Lotus product.

SQL3098N

The row number from the input file is not in the valid range of 1 to 8192.

Cause: The maximum number of rows that a worksheet format (WSF) spreadsheet can contain is 8192. The cell coordinate contains a value outside the valid range. Some damage occurred to the WSF file, or it was generated incorrectly, possibly with a level of the Lotus product not supported by the database manager.

The IMPORT utility stops processing.

Action: Regenerate the WSF file with a supported level of the Lotus product.

SQL3099N

The column number from the input file is not in the valid range of 1 to 256.

Cause: The maximum number of columns that a worksheet format (WSF) spreadsheet can contain is 256. The cell coordinate contains a value outside the valid range. Some damage has occurred to the WSF file, or it was generated incorrectly, possibly with a level of the Lotus product that is not supported by the database manager.

The IMPORT utility stops processing.

Action: Regenerate the WSF file with a supported level of the Lotus product.

SQL3100W

Column number "<column-number>" (identified as "<name>") in the output DEL format file is longer than 254 bytes.

Cause: The length or maximum length of the specified output column is greater than 254 bytes. Columns larger than 254 are not supported by some other products.

The entire field is exported without truncation.

Action: If the output file cannot be processed by another product, export only a substring of the incorrect column, redefine the table, or manually truncate the data in the DEL column.

SQL3101W

There is a character string delimiter in column "<column-number>" in row "<row-number>".

Cause: The system puts character string delimiters around the character strings being exported but it found a character string that already has the delimiter in the character string.

Delimiters will be placed around the character string. Future use of that string leads to an appearance of truncation. Processing can continue.

Action: Review the data for the given column and row in the output table or file. To avoid loss of data, change the character delimiter to some character that is not in the data.

SQL3102W

The number of columns in the dcoldata parameter is greater than the number of columns in the tcolstrg parameter.

Cause: The number of columns taken from the input file or table is greater than the number to be put in the output table or file.

Only data for the columns indicated in the output table or file will be processed. Data in the excess input columns is not processed.

Action: Review the data in the output table or file.

SQL3103W

The number of columns in the dcoldata parameter is less than the number of columns in the tcolstrg parameter.

Cause: The number of columns taken from the input file or table is less than the number to be put in the output table or file.

Only data for the columns indicated in the output table or file will be processed. Data in the excess output columns is not processed.

Action: Review the data in the output table or file.

SQL3104N

The Export/Unload utility is beginning to export/unload data to file "<name>".

Cause: This is the normal beginning message.

Action: No action is required.

SQL3105N

The Export/Unload utility has finished exporting/unloading "<number>" rows.

Cause: This is the summary message for the export/unload utilities which is printed at the end of the message file. This message will state how many rows resulting from the SELECT statement were processed before the export utility terminated or how many rows from the table where processed before the unload utility terminated.

Action: If an sqlcode of 0 was returned from the utility, then no response is required. If an sqlcode of 3107 was returned, check the message file for warnings that occurred during the export/unload and resubmit the command if desired. If a negative sqlcode was returned, an error occurred during the export/unload and the data file may not contain all the data requested. The command should be resubmitted with the error corrected.

SQL3106N

An error occurred while formatting a message for the message file.

Cause: The error message may be incomplete or formatted incorrectly.

Action: No action is required.

SQL3107W SQL3115W

SQL3107W

There is at least one warning message in the message file.

Cause: At least one warning was received during processing.

This warning does not affect processing.

Action: Review the message file warning.

SQL3109N

The utility is beginning to load data from file "<name>".

Cause: This is the normal beginning message. The message may indicate the name of a temporary file created on the server instead of the source file.

Action: No action is required.

SQL3110N

The utility has completed processing. "<number>" rows were read from the input file.

Cause: This is the normal ending message.

Action: No action is required.

SQL3111C

An I/O error occurred while closing the input data file.

Cause: A system I/O error occurred while closing the input data file. This error can refer to a problem on either the client or the server.

The file is not closed.

Action: Examine the input file for an I/O error.

SQL3112W

There are fewer input file columns specified than database columns.

Cause: There are fewer columns specified in the input file than in the output table. Because the extra columns in the table are defined as nullable, the values in those columns are filled with nulls.

Action: No action is required.

SQL3113W

The data type "<type>" of the database column "<name>" is not compatible with this format file. Null values are inserted for the column.

Cause: The database column type is not valid for this file. Because the column is nullable, nulls are inserted.

Null values are loaded for the indicated column.

Action: If nulls are not acceptable for the column, do one of the following:

- Edit the data in the table.
- If possible, use another compatible column in the database table as the target column and resubmit the command.
- Redefine the database table so the columns are compatible with the columns being loaded and resubmit the command.

SQL3114W

Some data following "<text>" in row "<row-number>" and column "<column-number>" was not loaded.

Cause: Some data in the specified row and column is not loaded, probably because the column contains data as follows:

- A closed string delimiter
- An isolated carriage return or line feed control character
- A non-delimited string value.

The text that was loaded is shown in the "<text>" token.

The contents of the field may be incomplete.

Action: Compare the value in the output table with the input file. If necessary, correct the input file and resubmit the command, or edit the data in the table.

SQL3115W

The field value beginning "<text>" in row "<row-number>" and column "<column-number>" is longer than the longest allowable table column. The value was truncated.

Cause: The field value is longer than 32700 bytes.

The value was truncated after 32700 bytes.

Action: Compare the value in the output table with the input file. If necessary, correct the input file and

resubmit the command or edit the data in the table. Check for unmatched character string delimiters.

SQL3116W

The field value in row "<row-number>" and column "<column-number>" is missing, but the target column is not nullable.

Cause: A null field value was encountered in the input file. Because the target column in the table is not nullable, it cannot be loaded.

For files other than ASCII files, the value of the column number specifies the field within the row of the missing data. For ASCII files, the value of the column number specifies the byte position within the row of the missing data.

The row is not loaded.

Action: If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3117W

The field value in row "<row-number>" and column "<column-number>" cannot be converted to a SMALLINT value. A null was loaded.

Cause: The value in the specified field cannot be converted to a SMALLINT value. There may be a data type mismatch. The value may be larger than a 2-byte integer.

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

A null value is loaded.

Action: Examine the input value. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3118W

The field value in row "<row-number>" and column "<column-number>" cannot be converted to a SMALLINT value, but the target column is not nullable. The row was not loaded.

Cause: The value in the specified field cannot be converted to a SMALLINT value. There may be a data type mismatch. The value may be larger than a 2-byte integer. A null cannot be loaded because the output column in the table is not nullable.

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

The row is not loaded.

Action: Correct the input file and resubmit the command or edit the data in the table.

SQL3119W

The field value in row "<row-number>" and column "<column-number>" cannot be converted to an INTEGER value. A null was loaded.

Cause: There may be a data type mismatch because the value in the specified field cannot be converted to an INTEGER value.

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

A null value is loaded.

Action: Examine the input value. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3120W

The field value in row "<row-number>" and column "<column-number>" cannot be converted to an INTEGER value, but the target column is not nullable. The row was not loaded.

Cause: There may be a data type mismatch because the value in the specified field cannot be converted to an INTEGER value. A null cannot be loaded because the output column in the table is not nullable.

SQL3121W • SQL3125W

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

The row is not loaded.

Action: Correct the input file and resubmit the command or edit the data in the table.

SQL3121W

The field value in row "<row-number>" and column "<column-number>" cannot be converted to a FLOAT value. A null was loaded.

Cause: The value in the specified field cannot be converted to a FLOAT value. There may be a data type mismatch.

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

A null value is loaded.

Action: Examine the input value. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3122W

The field value in row "<row-number>" and column "<column-number>" cannot be converted to a FLOAT value, but the target column is not nullable. The row was not loaded.

Cause: The value in the specified field cannot be converted to a FLOAT value. There may be a data type mismatch. A null cannot be loaded because the output column in the table is not nullable.

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

The row is not loaded.

Action: Correct the input file and resubmit the command or edit the data in the table.

SQL3123W

The field value in row "<row-number>" and column "<column-number>" cannot be converted to a PACKED DECIMAL value. A null was loaded.

Cause: The value in the specified field cannot be converted to a PACKED DECIMAL value. There may be a data type mismatch.

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

A null value is loaded.

Action: Examine the input value. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3124W

The field value in row "<row-number>" and column "<column-number>" cannot be converted to a PACKED DECIMAL value, but the target column is not nullable. The row was not loaded.

Cause: The value in the specified field cannot be converted to a PACKED DECIMAL value. There may be a data type mismatch. A null cannot be loaded because the output column in the table is not nullable.

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

The row is not loaded.

Action: Correct the input file and resubmit the command or edit the data in the table.

SQL3125W

The character data in row "<row-number>" and column "<column-number>" was truncated because the data is longer than the target database column.

Cause: The length of the field data in the input file was longer than the length of the database field where it is being loaded.

The character data was truncated.

Action: Compare the value in the output table with the input file. If necessary, correct the input file and resubmit the command or edit the data in the table. The width of the database column cannot be increased. If necessary, define a new table with wider columns and repeat the process.

SQL3128W

The field containing "<data>" in row "<row-number>" and column "<column-number>". was truncated into a DATE field because the data is longer than the database column.

Cause: The date value in the specified field is longer than the length of the string representation of a date.

The date value is truncated to fit into the table.

Action: Compare the value in the output table with the input file. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3129W

The date, time, or timestamp field containing "<text>" in row "<row-number>" and column "<column-number>" was padded with blanks.

Cause: The field data in the input file was shorter than the database column.

The data on the right is padded with blanks.

Action: Compare the value in the output table with the input file. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3130W

The field containing "<text>" in row "<row-number>" and column "<column-number>" was truncated into a TIME field because the data is longer than the database column.

Cause: The time value in the specified field is longer than the length of the string representation of a time.

The time value is truncated to fit into the table.

Action: Compare the value in the output table with the input file. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3131W

The field containing "<text>" in row "<row-number>" and column "<column-number>" was truncated into a TIMESTAMP field because the data is longer than the database column.

Cause: The timestamp value in the specified field is longer than the length of the string representation of a timestamp.

The timestamp value is truncated to fit into the table.

Action: Compare the value in the output table with the input file. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3132W

The character data in column "<column>" will be truncated to size "<size>".

Cause: The character data column has a defined size that is longer than the longest default character column which can be exported; each value will be truncated to the specified size.

For example, by default, only the first SQL_LONGMAX bytes of a LOB column will be exported. If the entire LOB column is desired, the *LOBSINFILE* keyword must be specified in the filetype modifier and each LOB column will be stored into a separate file.

Action: This is a warning only. No action is required.

SQL3137W

Row "<row-number>" is too short. At least one input value being loaded to a non-nullable database column is missing. The row was not loaded.

Cause: If loading from a delimited ASCII file, the row contains too few fields. If loading from a non-delimited ASCII file, the row contains too few bytes of data. There is no input value for at least one non-nullable target column.

The row is not loaded.

Action: Examine the input file and the contents of the target table. Correct the input file and resubmit the command, or edit the data in the table.

SQL3138W   SQL3145W

SQL3138W

The ending character string delimiter was not found before the end of the input data file was reached.

Cause: The end of the input data file was reached before the terminating character string delimiter was found.

A closing character string delimiter is assumed at the end of the data.

Action: Compare the value in the output table with the input file. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3139W

An error "<error>" occurred while the utility was disconnecting from the database.

Cause: The IMPORT or EXPORT utility could not disconnect from the database.

The output data may be incomplete.

Action: Use the error number in the message to determine the exact error.

SQL3142W

The column heading for column "<column-number>" is being truncated to 240 bytes.

Cause: The LOTUS 1-2-3** and Symphony** programs have a limit of 240 bytes for label records. If a column heading greater than 240 bytes is specified for export, it is be truncated to 240 bytes.

The column heading is truncated. Processing continues.

Action: Verify that the column headings are 240 bytes or less. Check for a possible error in specifying the names for the columns in the output worksheet format (WSF) file.

SQL3143W

A maximum length of variable length column "<column-number>" exceeds the limitation of 240 bytes. Data from the column may be truncated.

Cause: The LOTUS 1-2-3** and Symphony** programs have a limit of 240 bytes for label records. Whenever a character field longer than 240 bytes is written to a worksheet format (WSF) file, the data will be truncated to 240 bytes.

Continue processing. Subsequent data entries for the column may be truncated.

Action: Verify the output. If significant data from the column is lost because of truncation, investigate selecting the data from the column in several fields by substringing, or redesign the database.

SQL3144W

The length of fixed length column "<column-number>" exceeds the 240-byte limitation. Data from the column may be truncated.

Cause: The Lotus 1-2-3** and Symphony** programs have a limit of 240 bytes for label records. Whenever a character field longer than 240 bytes is written to a worksheet format (WSF) file, the data will be truncated to 240 bytes.

All data entries for the column are truncated with no additional messages written to the message log.

Continue processing.

Action: Verify output. If significant data from the column is lost because of truncation, investigate selecting the data from the column in several fields by substringing, or redesign the database.

SQL3145W

The data for row "<row-number>", column "<column-number>" is being truncated to 240 bytes.

Cause: The Lotus 1-2-3** and Symphony** programs have a limit of 240 bytes for label records. Whenever a character field longer than 240 bytes is written to a worksheet format (WSF) file, the data is truncated to 240 bytes. This message is preceded by message SQL3143 associated with the column.

Processing continues. The data is truncated.

Action: Verify the output. If significant data from the column is lost because of truncation, investigate selecting the data from the column in several fields by substringing, or redesign the database.

SQL3146N

The DATE or TIMESTAMP value for row "<row-number>" and column "<column-number>" is out of range.

Cause: The value for a date or timestamp is not valid. A date value from 01-01-1900 to 12-31-2099 is valid for worksheet format (WSF) files.

The cell record is not created.

Action: Compare the value in the output file with the input table. If necessary, correct the input value and resubmit the command or edit the data in the table.

SQL3147W

More than 2048 rows were exported into a worksheet formatted file.

Cause: The number of rows exported exceeds 2048. The first generation products cannot support more than 2048 rows.

Continue processing.

Action: The rows beyond 2048 can be read only by second and third generation products.

SQL3148W

A row from the input file was not inserted into the table. SQLCODE "<sqlcode>" was returned.

Cause: The database operation to insert a row of data read from the input file failed. One or more of the fields in the input file is not compatible with the database field where the field is being inserted.

Processing continues with the next row of input data.

Action: See the next message in the message file for the row number not inserted. Examine the input file and the database contents. If desired, modify the database or modify the input file and rerun the operation.

SQL3149N

"<number-1>" rows were processed from the input file. "<number-2>" rows were successfully inserted into the table. "<number-3>" rows were rejected.

Cause: This summary message tells how many rows of data were read from the input file, how many rows were successfully inserted into the database table, and how many rows were rejected. If using the INSERT_UPDATE option, the number of rows updated is the number of rows processed minus the number inserted and rejected.

Action: None, because this is a summary message. The detail messages may suggest corrective action.

SQL3150N

The H record in the PC/IXF file has product "<product>", date "<date>", and time "<time>".

Cause: Information is given about the product that created the PC/IXF file and when it was created.

Action: No action is required.

SQL3151N

No data conversion will be done from the single-byte code page value "<code-page>" in the H record to the application single-byte code page value "<code-page>" because the forcein option was specified.

Cause: No data conversion will be performed from the IXF code page to the application code page because the forcein option was specified.

Action: No action is required. If the conversion from the IXF file code page to the application code page is supported by the database manager, the operation can be resubmitted without the forcein option and the data will be converted.

SQL3152N

The double-byte code page value "<value>" in the H record is not compatible with the double-byte code page value "<value>" for the application. Data will be imported because the forcein option was specified.

Cause: The double-byte code page values in the record and the application are not compatible. Because the *forcein* option was used, the data is imported.

SQL3153N SQL3160W

Action: No action is required.

SQL3153N

The T record in the PC/IXF file has name "\<name>", qualifier "\<qualifier>", and source "\<source>".

Cause: Optional information is given about the name of the table where data was extracted, the product that created the table, and the original source of the data.

Action: No action is required.

SQL3154W

The HCNT value in the H record and the CCNT value in the T record are not compatible. The CCNT value in the T record will be used.

Cause: The HCNT value in the H record and the CCNT value in the T record do not agree.

The CCNT value in the T record is used.

Action: Verify that the CCNT value is correct. If not, make the necessary changes to the HCNT or CCNT values and resubmit the command.

SQL3155W

The name length field in the C record for column "\<name>" is not valid. Data from the column will not be loaded.

Cause: The value in the name length field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

Action: Change the name length field in the C record and resubmit the command.

SQL3156W

The null field in the C record for column "\<name>" is not valid. Data from the column will not be loaded.

Cause: The null field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

Action: Change the null field in the C record and resubmit the command.

SQL3157W

The type field in the C record for column "\<name>" is not valid. Data from the column will not be loaded.

Cause: The type field in the C record for the indicated column is not valid. The code page values may not be compatible with the column type.

Data from the indicated column is not loaded.

Action: Change the type field in the C record and resubmit the command.

SQL3158W

The single-byte code page field in the C record for column "\<name>" is not valid. Data from the column will not be loaded.

Cause: The single-byte code page field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

Action: Change the single-byte code page field in the C record and resubmit the command.

SQL3159W

The double-byte code page field in the C record for column "\<name>" is not valid. Data from the column will not be loaded.

Cause: The double-byte code page field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

Action: Change the double-byte code page field in the C record and resubmit the command.

SQL3160W

The column length field in the C record for column "\<name>" is not valid. Data from the column will not be loaded.

Cause: The column length field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

Action: Change the column length field in the C record and resubmit the command.

SQL3161W

The precision field in the C record for column "<name>" is not valid. Data from the column will not be loaded.

Cause: The precision field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

Action: Change the precision field in the C record and resubmit the command.

SQL3162W

The scale field in the C record for column "<name>" is not valid. Data from the column will not be loaded.

Cause: The scale field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

Action: Change the scale field in the C record and resubmit the command.

SQL3163W

The column length field in the C record for floating point column "<name>" is blank. A value of 00008 will be used.

Cause: The column length field in the C record for the indicated column is blank.

A column length of 00008 is used.

Action: No action is required.

SQL3164W

The column length field in the C record for floating point column "<name>" is not valid. Data from the column will not be loaded.

Cause: The column length field in the C record for the indicated column is not valid. The indicated column is a floating point column.

Data from the indicated column is not loaded.

Action: Change the column length field in the C record and resubmit the command.

SQL3165W

The column type field "<type>" in the C record for column "<name>" is not valid. Data from the column will not be loaded.

Cause: The column type in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

Action: Change the column type field in the C record and resubmit the command.

SQL3166W

No PC/IXF column was specified to be loaded into database column "<name>", or the specified PC/IXF column does not exist. Nulls will be inserted.

Cause: Either there was not a PC/IXF column specified to be loaded into the indicated column, or the specified PC/IXF source column does not exist.

Null values are loaded for the indicated column.

Action: No action is required. If nulls are not acceptable for this column, check the *dcoldata* parameter for erroneous names or positions or for fewer entries in the *dcoldata* parameter than columns stated in or implied by the *tcolstrg* parameter.

SQL3167W

The PC/IXF column specified to be loaded into database column "<name>" is not valid. Nulls will be inserted.

Cause: Values in the PC/IXF column cannot be loaded into the indicated database column and the reason is given in a previous message in the log.

Null values are loaded for the indicated column.

Action: Read the previous messages to understand why the column is not valid.

SQL3168W

The PC/IXF column specified to be loaded into database column "<name>" is not compatible with the database column. Nulls will be inserted.

Cause: The source PC/IXF and the target database column types or lengths may not be compatible.

Null values are loaded for the indicated column.

SQL3169N   SQL3175W

Action: Compare the columns in the source PC/IXF file and the database.

SQL3169N

The forcein option may be used to make the PC/IXF column "<name>" acceptable for loading into database column "<name>".

Cause: This is for information only about the optional use of the *forcein* option.

Action: No action required.

SQL3170W

The end of the file was reached within a row of data. The partial row of data was not loaded.

Cause: The end of the file was reached before the end of the current row of data. The file may contain only part of the expected data.

The partial row of data is not loaded.

Action: If the PC/IXF file was copied from one medium to another, compare the copy with the original or repeat the copy process.

SQL3171W

A non-label record was found in the column header row. The record was not processed.

Cause: The IMPORT utility expects only label records in the column header row (row one) of the worksheet format (WSF) file.

The system does not process the record and continues to the next record.

Action: Remove all data and information except the column headers from the first row of the spreadsheet file. Resubmit the command.

SQL3172W

The specified input column "<name>" was not found. The corresponding database column will contain null values.

Cause: The specified input column was not found in the input spreadsheet file. The database column is nullable and contains null values.

Action: Verify the specified input column name.

SQL3173N

The imported data for column "<name>" will always contain fewer characters than the column width.

Cause: The database column width is larger than the maximum worksheet format (WSF) label record.

Continue processing.

Action: No action is required.

SQL3174W

The data type "<type>" of the database column "<name>" is not compatible with any WSF column type. Null values will be inserted for this column.

Cause: The database column type is not valid for a worksheet format (WSF) file. Because the column is nullable, nulls are imported for the column.

Action: If nulls are not acceptable for the column, do one of the following:

* Edit the data in the table.
* If possible, use another compatible column in the database table as the target column and resubmit the command.
* Redefine the database table so that the columns are compatible with the columns imported from the WSF file, and resubmit the command.

SQL3175W

The input record for database row "<row>", column "<column>" is not valid.

Cause: The record in the spreadsheet file is not compatible with the data type of the database column. If the database column is a graphic data type, the input data may contain an odd number of bytes.

If the column is nullable, a null is inserted. If the column is not nullable, the row is not imported.

Action: Either edit the data in the table or verify that the data in the spreadsheet file is valid for importing into a database manager database and resubmit the command.

SQL3176W

The value for row "<row>", column "<column>" in the WSF file is out of range for a date value.

Cause: The record in the spreadsheet file contains a value that is too large or too small to represent a valid worksheet format (WSF) date. A valid WSF date is between 1 and 73050, inclusive.

If the column is nullable, a null is inserted. If the column is not nullable, the row is not imported.

Action: Either edit the data in the table or verify that the data in the spreadsheet file is valid for import into a database manager database and resubmit the command.

SQL3177W

The value in the WSF file for row "<row>", column "<column>" is out of range for a time value.

Cause: The record in the spreadsheet file contains a value that is either too large or too small to represent a valid worksheet format (WSF) time. A WSF time is a value equal to or greater than zero, but less than one.

If the column is nullable, a null is inserted. If the column is not nullable, the row is not imported.

Action: Either edit the data in the table or verify that the value to be imported is a time value in the input spreadsheet file and resubmit the command.

SQL3178W

The type of record in the WSF file for database row "<row-number>", column "<column-number>" is not valid for representing a time value.

Cause: The input value is an integer value. A time value must be represented by a floating point number or a fraction of a day in a worksheet format (WSF) spreadsheet file.

If the column is nullable, a null is inserted. If the column is not nullable, the row is not imported.

Action: Either edit the data in the table or verify that the value to be imported is a time value in the input spreadsheet file and resubmit the command.

SQL3179W

Row "<row>" in the input file is missing data for importing into a non-nullable column in the database. The row was not imported.

Cause: The row of data from the input file has missing or not valid data for a non-nullable column. The values in the remaining database columns in that row are not imported.

Processing continues with the next row. The row is not imported.

Action: Either edit the data in the table or verify that the data in the spreadsheet file is valid for importing into a database manager database.

SQL3180W

Insert diskette "<number>" into drive "<drive>".

Cause: This is a request for the application to prompt the user to place the specified diskette in the specified drive.

The utility is waiting to be called again after the user puts the diskette in the drive.

Action: Prompt the user for the diskette and then return to the utility with the *callerac* parameter indicating whether processing continues or ends.

SQL3181W

The end of the file was reached before the expected ending record was found.

Cause: During the load of a PC/IXF file created by the database manager, the A record of subtype E that is expected as the last A record was not found.

The input file probably is damaged.

Continue processing.

Action: Check for data that was not loaded. If data is missing, edit the table or change the input file and resubmit the command. If the PC/IXF file was copied from one medium to another, compare the copy with the original or repeat the copy process.

SQL3182W   SQL3187W

SQL3182W

Insert diskette "<number>" into drive "<drive>". The diskette that is currently inserted is not the right diskette or the continuation diskette is not valid.

Cause: During the load of a PC/IXF file that is contained on more than one diskette, a request to insert a diskette was sent to the application, a confirmation that the diskette is in the drive was returned but the continuation file is not there or is not valid. This action does not apply to the first diskette.

The utility waits for a response from the application to either continue processing or stop processing.

Action: Have the user verify that the correct diskette is in the drive. If the correct diskette is in the drive, call the utility again with the *callerac* parameter set to stop processing.

SQL3183W

Multiple delimiter overrides in the filetmod parameter are not separated by blanks.

Cause: At least one COLDEL, CHARDEL or DECPT keyword in the *filetmod* parameter is not at the beginning of the *filetmod* parameter and does not follow a blank (space). This condition may arise during the import or export of delimited ASCII (DEL) files.

The utility stops processing. The erroneous delimiter overrides are ignored.

Action: Resubmit the command with a correct *filetmod* parameter.

SQL3185W

The previous error occurred while processing data from row "<row-number>" of the input file.

Cause: This message provides the identification of the row where the error occurred for the previous message listed in the message file (for example, SQL3306).

Action: No action is required.

SQL3186W

Data was not loaded into the database, because the log was full. SQLCODE "<sqlcode>" was returned. A commit will be attempted and the operation will continue if the commit is successful.

Cause: The utility could not insert a row of data into the database because the database transaction log is full.

The completed database transactions are committed and the insert is tried again. If the repeated insert continues to indicate that the log is full, the utility stops processing.

Action: Be aware that a subsequent failure of the utility causes the database to roll back to the state after the last commit, not to the state before the utility was initially called.

SQL3187W

An error occurred while creating an index. SQLCODE "<sqlcode>" was returned.

Cause: The IMPORT utility was creating an index when an error occurred. An index may already exist for some table.

This error occurs only during the import of a PC/IXF file.

The utility continues processing. The file was imported, but the index was not created on the table.

DataJoiner users: in addition to the previously listed cause, this error can occur when a CREATE NICKNAME statement was issued at DataJoiner and the table at the data source had an index or indexes with too many columns, or the total index row size could not be represented in the DataJoiner catalog. The "<sqlcode>" in the message provides further information about the problem encountered.

Action: Read the next message in the message log (SQL3189) for the name of the index that was not created. Create the index with the CREATE INDEX command.

DataJoiner users: select from the data source catalog and the DataJoiner catalog to determine which index or indexes were not created in the DataJoiner catalog. Either:

- Create the indexes using the CREATE INDEX command. Remember to truncate columns as appropriate in order to not violate limits which caused the index to not be created in the first place.

- Do nothing and allow DataJoiner to function without knowledge about the index.

Both of the previously listed options have potential performance implications.

SQL3188N

An error occurred while erasing the contents of the table.

Cause: When running IMPORT with the REPLACE option, the specified database table is truncated before inserting the data back into the table. An error occurred during the truncation processing.

The utility ends with an error.

Action: Resubmit the command.

SQL3189N

The previous message refers to index "<name>" with columns "<column list>".

Cause: This message always follows message SQL3187 when an error occurs while creating an index. "<name>" is the name of the index where the create failed. "<column list>" is a string of index column names. Each column name in the string is preceded by a plus (+) or a minus (–) sign indicating ascending or descending order.

The utility continues processing.

Action: Manually create the index with the CREATE INDEX command.

SQL3190N

The indexixf option is not valid for this Import operation.

Cause: If INDEXIXF is used in the *filetmod* parameter of the IMPORT command, each of the following is also required:

- The IMPORT must be replacing the contents of a table.
- The *dcoldata* parameter must be null.
- Each IXF column must be targeted to a database column with the same name.

The utility stops processing. No data is imported.

Action: Resubmit the command either without the INDEXIXF option or with the other parameters that are valid with the INDEXIXF option.

SQL3193N

The specified view cannot be updated. You cannot import into this view.

Cause: The IMPORT utility can be run against a view only if the view can be updated. The specified view is defined such that data in it may not be changed.

The IMPORT utility stops processing. No data is imported.

Action: Resubmit the command with the name of a table or a view that can be updated.

SQL3194N

The specified table is a system table. You cannot load a system table.

Cause: The utility cannot be run against a system table.

The utility stops processing. No data is loaded.

Action: Resubmit the command with a valid table name.

SQL3195W

The diskette "<number>" in drive "<drive>" cannot be used for the output file. Insert a formatted diskette with writable free space.

Cause: The EXPORT utility cannot use the current diskette for exporting to a PC/IXF file for one of the following reasons:

- The output file cannot be opened on the diskette. The diskette may be unformatted.
- Not enough free space is available on the diskette.

This warning code is a request for the application to prompt the user to place another diskette in the specified drive.

The utility is waiting to be called again after the user puts the diskette in the drive.

Action: Prompt the user for the diskette and then return to the utility with the *callerac* parameter indicating whether processing continues or ends.

SQL3196N   SQL3220W

SQL3196N

The input file was not found.

Cause: The source file to be loaded into a database could not be found in the path indicated by the *datafile* parameter.

The command cannot be processed.

Action: Verify that the input file does exist and that the path to it is correct.

SQL3197N

An attempt was made to execute multiple copies of import or export.

Cause: An attempt was made to execute more than one instance of the import or export utility on a system where this is not supported.

The command cannot be processed.

Action: Resubmit the second operation when no other processes are attempting to execute the same utility.

SQL3201N

The specified table cannot be replaced because another table is dependent on it.

Cause: A table that is a parent in a relationship with another table cannot be replaced.

The command cannot be processed.

Action: Select a different option, such as INSERT, or select a different target for the utility operation.

SQL3203N

The INSERT_UPDATE option is not permitted for the specified target because it has no primary key or all columns are in the primary key.

Cause: The INSERT_UPDATE option is only valid when the target table has a primary key and the target columns include all columns of the primary key. In addition, the target table should contain at least one column that is not part of the primary key.

The command cannot be processed.

Action: Select a different option such as INSERT, or choose a different target for the utility operation.

SQL3204N

The INSERT_UPDATE option may not be applied to a view.

Cause: The INSERT_UPDATE option is not valid for views and a view has been selected as the target of the utility operation.

The command cannot be processed.

Action: Select a different option such as INSERT, or choose a different target for the utility operation.

SQL3205N

The specified view cannot be replaced because the base table has a dependent.

Cause: A view whose base table is a parent table in a referential relationship with any table (including itself) cannot be replaced.

The command cannot be processed.

Action: Select a different option such as INSERT, or choose a different target for the utility operation.

SQL3206N

The specified view cannot be replaced because its definition involves a subquery.

Cause: A view whose definition involves a subquery cannot be replaced. If the view definition depends on the definition of other views, the other views cannot contain subqueries. The use of a subquery anywhere in the definition of any view on which the target view is based prohibits the REPLACE option.

The command cannot be processed.

Action: Select a different option, such as INSERT, or choose a different target for the utility operation.

SQL3220W

Volume <volume-name> was not found in the <directory-name> directory. Copy the volume into this directory and continue the import.

Cause: An attempt to import a multiple IXF file was made, but one of the files is missing from the directory specified. Import tries to find the parts in the same directory as the first part.

The import will terminate.

Action:

- Find the part and put it in the same directory as the first part. Then call import again with *callerac* of SQLU_CONTINUE. The import will continue processing the file.

- Terminate the import by calling import with a callerac of SQLU_TERMINATE.

SQL3221W

...Begin COMMIT WORK. Input Record Count = <count>

Cause: The import is about to attempt a COMMIT of the work performed.

Action: If the SQL3222W message is not printed immediately after this message, then the COMMIT failed and you will need to check your table or view to see which records have been imported. You can then start the import again with a RESTARTCOUNT set to the number of records successfully imported to skip those rows and import the rest of the file. (If you had been doing a CREATE or REPLACE_CREATE or REPLACE, call the second import with INSERT option.)

SQL3222W

...COMMIT of any database changes was successful.

Cause: The COMMIT was successful.

Action: None necessary if you get this message.

SQL3223N

The parameter of type pointer to <parameter> is not specified properly.

Cause: The parameter of type <parameter> is not specified properly. The types are one of "struct sqluimpt_in", "struct sqluimpt_out", "struct sqluexpt_out", "struct sqluload_in", "struct sqluload_out", "struct sqluunld_in", or "struct sqluunld_out". The pointer should be either a NULL pointer or should point to the appropriate structure and its sizeOfStruct field should be initialized to one of SQLUIMPT_IN_SIZE for struct sqluimpt_in, SQLUIMPT_OUT_SIZE for struct sqluimpt_out, SQLUEXPT_OUT_SIZE for struct sqluexpt_out, SQLULOAD_IN_SIZE for struct sqluload_in, SQLULOAD_OUT_SIZE for struct sqluload_out, SQLUUNLD_IN_SIZE for struct sqluunld_in or SQLUUNLD_OUT_SIZE for struct sqluunld_out.

The command cannot be processed.

Action: Call the utility again with a proper parameter.

SQL3225N

The RESTARTCOUNT value is greater than the number of rows in the file. No rows were loaded.

Cause: The utility was called with a RESTARTCOUNT value which was greater than the number of rows in the input file resulting in no rows being loaded into the table/view.

Action: Ensure that the RESTARTCOUNT value is correct or call the utility again with the REPLACE or INSERT option and a correct RESTARTCOUNT value.

SQL3250N

COMPOUND="<value>" is an invalid value or is not allowed in conjunction with other import parameters.

Cause: The COMPOUND=x option was specified for the import utility and cannot be processed due to one of the following reasons:

- It is invalid when the INSERT_UPDATE option is used.

- The database being imported to is accessed through a server or gateway of a previous release.

- The value is not within the allowed range of 1 to 100. (On DOS or Windows, the maximum value is 7).

Action: Change the filetmod option to correct the usage of COMPOUND=x.

SQL3251N

"number" more errors occurred during the import.

Cause: The utility encountered more errors than can be relayed in the sqlca (maximum number is 7) while using the COMPOUND option. Messages for these errors will not be printed to the message file.

The utility continues processing.

Action: If you desire all the error messages for each row inserted during an import, do not use the COMPOUND option or use a COMPOUND value less than or equal to 7.

SQL3300N SQL3306N

SQL3300N

The records in the input file are not in the correct sequence.

Cause: The records in the worksheet format (WSF) file are expected to be in ascending order (row 1, col 1 ... row 1, col 256; row 2, col 1 ... row 2, col 256, and so on). Some damage has occurred to the WSF file or it was generated incorrectly, possibly with a level of the Lotus product not supported by the database manager.

The IMPORT utility stops processing.

Action: Regenerate the WSF file with a supported level of the Lotus product.

SQL3301N

A BOF record was found in the middle of the input file.

Cause: A beginning-of-file (BOF) record must be the first record of a worksheet format (WSF) file. It cannot occur at any other location in the file. Some damage has occurred to the WSF file or it was generated incorrectly, possibly with a level of the Lotus product not supported by the database manager.

The IMPORT utility stops processing.

Action: Regenerate the WSF file with a supported level of the Lotus product.

SQL3302N

An EOF record was found before any data was imported.

Cause: The input file is valid, but it did not contain any usable data to be imported. The first worksheet row is reserved for title information. The second worksheet row is used for column labels. Data begins in the third row.

The IMPORT utility stops processing.

Action: Regenerate the worksheet format (WSF) file with valid data in the rows that the database manager uses for data.

SQL3303N

The file type must be IXF when using the CREATE or REPLACE_CREATE keywords in the tcolstrg parameter.

Cause: File types other than IXF are not allowed with the CREATE or REPLACE_CREATE keywords in the *tcolstrg* parameter.

The IMPORT utility stops processing. No data is imported.

Action: Change the file type to IXF, or use INSERT, INSERT_UPDATE or REPLACE.

SQL3304N

The table does not exist.

Cause: The parameters specified on the command require that the table exists.

The utility stops processing. No data is loaded.

Action: Do one of the following:

- Resubmit the command with a table name of an existing table.
- If the input file is a PC/IXF file, resubmit the command with the CREATE option.

SQL3305N

The table cannot be created because it already exists.

Cause: The CREATE keyword indicates that a new table must be created but a table with the specified name already exists.

The IMPORT utility stops processing. No data is imported.

Action: Either erase the existing table or use a keyword other than CREATE and resubmit the command.

SQL3306N

An SQL error "<sqlcode>" occurred while inserting a row into the table.

Cause: An SQL error occurred while inserting a row into the table.

If the SQL error is not severe the row will be rejected and the utility continues processing; otherwise the utility stops processing.

Action: Examine other messages in the message file for details about the actual error and resubmit the command, if necessary.

SQL3307N

The number of columns in the dcoldata parameter does not match the number of entries in the tcolstrg parameter, or a column specified in the dcoldata parameter does not exist.

Cause: The CREATE or REPLACE_CREATE option was specified on the IMPORT command. Either:

- If the NAMES or POSITIONS method indicator was specified in the *dcoldata* parameter, the number of columns explicitly specified in *dcoldata* is not equal to the number of columns explicitly specified in the *tcolstrg* parameter.
- If the DEFAULT method indicator was specified in the *dcoldata* parameter, the number of columns in the PC/IXF file is less than the number of columns specified in the *tcolstrg* parameter.
- One column specified in the *dcoldata* parameter does not exist in the PC/IXF file.

The IMPORT utility stops processing. No table is created.

Action: Either correct the columns specified in the *dcoldata* and *tcolstrg* parameters or correct the columns specified in the *dcoldata* parameter.

SQL3308N

The code page values of the PC/IXF column "<name>" are not compatible with the code page values for the application. The forcein parameter was not specified.

Cause: The code page values for the column and the application are not compatible. When the forcein parameter is not specified, the data cannot be loaded if the conversion from the IXF file code pate to the application code page is not supported.

The utility stops processing. No data is loaded.

Action: To load data with these code pages, resubmit the command with the forcein option.

SQL3309N

Column "<name>" in the PC/IXF file is defined as a graphic column. The forcein parameter was not specified.

Cause: During the load of a PC/IXF file, a graphic data column was encountered. Because the *forcein* parameter was not used, the data cannot be loaded.

The utility stops processing. No data is loaded.

Action: If you want to load the data with the graphic data, resubmit the command with the *forcein* parameter.

SQL3310N

Column "<name>" in the PC/IXF file is not valid.

Cause: The CREATE or REPLACE_CREATE option was specified on the IMPORT command. A column with an invalid C record was found while importing the PC/IXF file.

The IMPORT utility stops processing. The table is not created.

Action: Verify the column definition information in the input file.

SQL3313N

The disk is full. Processing was ended.

Cause: The disk or diskette is full. During the export to a PC/IXF file, either the PC/IXF data file is on a fixed disk, or the PC/IXF data file and the database are on the same drive, or the PC/IXF data file and the message file are on the same drive.

The EXPORT utility stops processing. The exported data is not complete.

Action: Provide more space on the disk or diskette or specify that the data file be located on a different drive than the database or the message file and resubmit the command.

SQL3314N   SQL3320N

SQL3314N

The date and time fields in an A record do not match the date and time fields in the H record.

Cause: During the load of a PC/IXF file, an A record was found in the PC/IXF file, that contained run identification information (in the date and time fields) that differs from the run identification information in the header (H) record. This action does not apply to an A record at the start of a continuation file.

The input file has probably been damaged.

The utility stops processing.

Action: Re-create the damaged file, or repair the damaged file to recover as much data as possible. Resubmit the command.

SQL3315N

The volume fields in an A record of subtype C are not valid.

Cause: During the load of a PC/IXF file that was created by Database Services, an A record was found in the PC/IXF file, that contained volume information (in the volume fields) that is not valid.

The input file has probably been damaged.

The utility stops processing.

Action: Re-create the damaged file, or repair the damaged file to recover as much data as possible. Resubmit the command.

SQL3316N

An I/O error occurred while closing part of the input file.

Cause: During the load of a multiple part PC/IXF file, an I/O error occurred while the system was closing one of the files that make up the input PC/IXF file. This action does not apply to the last file in the group of files that make up the PC/IXF file.

The utility stops processing.

Action: Resubmit the command.

SQL3317N

The string pointed to by the filetmod parameter contains conflicting information.

Cause: The *filetmod* string defines the generation and product family for the output WSF file. More than one generation or product family was defined in the string.

The utility stops processing. The output file is not created.

Action: Change the *filetmod* string to define only one generation and product family. Resubmit the command.

SQL3318N

There are duplicate keywords in the filetmod parameter.

Cause: A COLDEL, CHARDEL, or DECPT keyword appears more than once in the *filetmod* parameter. This condition may arise while using delimited ASCII (DEL) files.

The utility stops processing. No data is loaded or exported.

Action: Resubmit the command with a correct *filetmod* parameter.

SQL3319N

An SQL error "<sqlcode>" occurred while creating the table.

Cause: An SQL error occurred while creating the table.

The IMPORT utility stops processing. The table was not created. No data was imported.

Action: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL3320N

There is no delimiter or decimal point character after a keyword in the filetmod parameter.

Cause: A COLDEL, CHARDEL, or DECPT keyword is located at the very end of the *filetmod* parameter. There is no delimiter or decimal point character following the keyword. This condition may arise while using delimited ASCII (DEL) files.

The utility stops processing. No data is loaded or exported.

Action: Resubmit the command with a correct *filetmod* parameter.

SQL3321C

Data was not imported into the database because the log was full. Recovery was not successful. SQLCODE "<sqlcode>" was returned.

Cause: The IMPORT utility could not insert a row of data into the database because the database transaction log was full. All work was committed, but the utility could not insert the row, because the database transaction log remains full.

The utility stops processing. All previous changes were committed, but the current row was not imported.

Action: Check the amount of space remaining on the filesystem containing your database files. Consider increasing the maximum log size in the database configuration file.

SQL3322N

An operating system semaphore error occurred.

Cause: An error occurred in the wait/post semaphores.

The utility stops processing. For the EXPORT utility, data may be incomplete on the media. For the IMPORT utility, data not already committed is rolled back.

Action: Stop and restart DB2 and resubmit the utility.

SQL3324N

Column "<name>" has a type of "<type>" which is not recognized.

Cause: A column of data returned by the SQL statement is not supported.

Action: Correct the error and retry the command.

SQL3325W

All the columns in row "<row-number>" have NULL values; the row will not be included in the WSF data file.

Cause: When you are doing an export of a WSF file and the SELECT statement results in a row of all NULL values, the row will not be added to the WSF file. The total number of rows specified in message SQL3105N is the number of rows resulting from the SELECT statement and not the number of rows in the WSF file.

The command continues processing.

Action: No action required. This is an informational message.

SQL3326N

The column list following the table name in the tcolstrg parameter is invalid.

Cause: When you call IMPORT or LOAD with a tcolstrg parameter which contains a column list after the table name, this message will result if it is invalid. For example, the following tcolstrg parameters will fail:

insert into tablea()   no columns within the parenthesis

insert into tablea(two words) invalid column name

insert into tablea(grant.col1) column name should not be qualified

insert into tablea(x1234567890123456789) column name too long

insert into tablea(,col1,col2) missing column name

The command cannot continue.

Action: Change the tcolstrg parameter with a valid column list and call the utility again.

SQL3327N

A system error occurred (reason code1 = " <reason-code-1>" and reason code2 = "<reason-code-2>").

Cause: A system error occurred during processing.

The utility stops processing.

Action: Record all error information from the SQLCA, if possible. Retain the message file. Terminate all applications using the database. Reboot the system. Restart the database. Try the command again.

If sufficient memory resources exist and the problem continues, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

SQL3330W SQL3337N

SQL3330W

In row "<row-number>" a character field has an odd length but its target database column is a graphic column. The row was not loaded.

Cause: Only even length character fields can be loaded into graphic database columns.

The row is not loaded.

Action: Either load the data to a new table using the CREATE option in the IMPORT command or do not load this column into this table.

SQL3331C

The permission setting of a file (or directory) does not allow the specified access.

Cause: This may accompany another error message. It indicates an attempt was made to get access to a file or directory in a way that is incompatible with the attributes of the file. One of the following may be the source of the difficulty:

- Opening a file for writing on a read-only device
- Opening a file for writing on a read-only file
- Opening a directory instead of a file
- Encountering a locking or sharing violation.

The command cannot be processed.

Action: Retry the utility when the file is not in use, or retry the utility directing the output to a path and file that allows writing.

SQL3332C

The maximum number of open files has been reached.

Cause: This message may accompany another error message. It indicates that the maximum number of open files has been reached.

The command cannot be processed.

Action: Terminate other applications to reduce the number of files open and retry the utility.

SQL3333C

The file or directory does not exist.

Cause: This message may accompany another error message. It indicates that a file or directory to be accessed does not exist or cannot be found.

The command cannot be processed.

Action: Resubmit the command with the correct file name, including a valid path.

SQL3334C

There is not enough storage available.

Cause: This message may accompany another error message. It indicates there is not enough storage available to open the file.

The command cannot be processed.

Action: Stop the application. Possible solutions include:

- Verify that your system has sufficient real and virtual memory.
- Remove background processes.

SQL3335C

The file system is full.

Cause: This message may accompany another error message. It indicates there is no available space for writing on the device.

The command cannot be processed.

Action: To make space available on the device, delete unneeded files or direct the output data to a device with space available.

SQL3337N

An I/O error occurred while writing data to the server.

Cause: An I/O error has occurred while attempting to write data to a temporary file on the server (temporary files are created in the tmp directory under the sqllib directory for the instance of the database manager). The file system on the server may be full.

The utility stops processing. The database is unchanged.

Action: Contact the System Administrator for the server to make space available on the server, and resubmit the command.

SQL3338N

An I/O error occurred reading a temporary message file on the server.

Cause: A system I/O error has occurred while attempting to read the temporary message file on the server.

The IMPORT operation is complete; however, the message file on the database client system may be empty or incomplete.

Action: Query the remote database to determine if the utility operation was successful.

SQL3400N

The method specified in dcoldata is not valid for Non-Delimited ASCII files. It must be 'L' for locations.

Cause: When loading from a non-delimited ASCII file, columns must be selected by locations in the file.

The command cannot be processed.

Action: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3401N

The method specified in dcoldata is not valid for any filetype.

Cause: The method for selecting columns in the file is not a value allowed for any *filetype*. Choose one of the following method indicators:

- P for positions
- N for names
- L for locations
- D for default

Further restrictions are based on *filetype*.

The command cannot be processed.

Action: Resubmit the command with a valid method indicator.

SQL3402N

A beginning-ending location pair with both values zero was specified for non-nullable column "<name>".

Cause: A location pair with both the beginning and the ending location set to zero was specified for the indicated column, but the column is not nullable.

The command cannot be processed.

Action: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3403N

The beginning-ending location pair for loading into column "<name>" is not valid.

Cause: The field specification for locating the input data for the indicated database column within the input non-delimited ASCII file is not valid. The field specification has one of the following errors:

- The beginning location is less than zero.
- The ending location is less than zero.
- The ending location is less than the beginning location.

The command cannot be processed.

Action: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3404N

The beginning-ending location pair for loading into column "<name>" is not valid for a number.

Cause: The field specification for locating the data for the indicated database column within the input non-delimited ASCII file is not valid. The location pair defines a field longer than 50 bytes.

The command cannot be processed.

Action: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3405N   SQL3410N

SQL3405N

The beginning-ending location pair for loading into column "<name>" is not valid for a date.

Cause: The field specification for locating data for the indicated database column within the non-delimited ASCII file is not valid. The location pair defines a field length that is not valid for an external representation of a date.

The command cannot be processed.

Action: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3406N

The beginning-ending location pair for loading into column "<name>" is not valid for a time.

Cause: The field specification for locating the data in the indicated database column within the input non-delimited ASCII file is not valid. The location pair defines a field length that is not valid for an external representation of a time.

The command cannot be processed.

Action: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3407N

The beginning-ending location pair for loading into column "<name>" is not valid for a timestamp.

Cause: The field specification for locating the data for the indicated database column within the input non-delimited ASCII file is not valid. The location pair defines a field length that is not valid for an external representation of a timestamp.

The command cannot be processed.

Action: Resubmit the command with a valid set of locations for the columns in the file.

SQL3408W

The beginning-ending location pair for loading into column "<name>" defines a field larger than the target column. Data may be truncated.

Cause: The field specification for obtaining data from the input non-delimited ASCII file defines a field larger than the size (or maximum size) of the target database column.

The utility continues processing. Truncation occurs when necessary.

Action: No action is required.

SQL3409W

The beginning-ending location pair for loading into column "<name>" defines a field shorter than the target fixed-length column. Data will be padded.

Cause: The indicated database column is a fixed-length column. The field specification for obtaining data from the input non-delimited ASCII file defines a field smaller than the size of the target database column.

The utility continues processing. Values input to the indicated database column are padded with spaces as necessary, on the right.

Action: No action is required.

SQL3410N

The beginning-ending location pair for loading into column "<name>" is not valid for a graphic column.

Cause: The field specification for locating the input data in the ASCII file that is to be loaded into the indicated database column defines a field with an odd number of bytes.

The command cannot be processed.

Action: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3411W

The field value in row "<row-number>" and column "<column-number>" is not valid for a graphic column. A null was loaded.

Cause: The value in the field indicated is not an acceptable graphic column value. The value may contain an odd number of bytes. For DEL files, the value of the column number specifies a field in the indicated row. For ASCII files, the value of the column number specifies the byte location within the row where the value begins.

The row is not loaded.

Action: If a null is not acceptable, correct the input file and resubmit the command, or edit the data in the table.

SQL3412W

The field value in row "<row-number>" and column "<column-number>" is not valid for a graphic column but the target column is not nullable. The row was not loaded.

Cause: The value in the field indicated is not an acceptable graphic column value. The value may contain an odd number of bytes. A null cannot be loaded because the target column is not nullable. For DEL files, the value of the column number specifies a field in the indicated row. For ASCII files, the value of the column number specifies the byte location within the row where the value begins.

The row is not loaded.

Action: No action is required. If the row is necessary, correct the input file and resubmit the command, or edit the data in the table.

SQL3413W

The field value in row "<row-number>" and column "<column-number>" is too short for the target column. A null was loaded.

Cause: The value in the specified field is not acceptable because it is too short for the target column. The value of the column number specifies the byte location within the row where the field begins.

A null value is loaded.

Action: No action is required. If a null is not acceptable, correct the inner field and resubmit the command, or edit the data in the table.

SQL3414N

The temporary file "<filename>" could not be found.

Cause: At the end of the load phase a temporary file is created with information necessary to restart the load. If the load is interrupted before this point, this file is not created.

This message indicates that this file could not be found during load restart.

The utility stops.

Action: Depending on where the interrupt occured, it may be possible to restart the load at the build phase.

SQL3500W

The utility is beginning the "<phase>" phase at time "<timestamp>".

Cause: This is an informational message indicating that a phase is about to begin and that the previous phase has ended. The phases are (in order of appearance):

- LOAD
- BUILD
- DELETE

During the LOAD phase, data is loaded into your table. If there are any indexes to be built, BUILD phase will follow the LOAD phase. If there were any duplicate keys found for a unique index, the DELETE phase will follow the BUILD phase.

If the LOAD is terminated before the end of the LOAD, you need to know the phase at which you should restart the LOAD.

Action: No action is required.

SQL3501W

The table space(s) in which the table resides will not be placed in backup pending state since forward recovery is disabled for the database.

Cause: COPY NO was specified on the invocation of LOAD which results in the table space(s) in which the table resides being placed into backup pending state unless forward recovery is disabled for the database.

The utility continues.

Action: No action is required.

SQL3502W   SQL3508N

SQL3502W

The utility has encountered "<number>" warnings which exceeds the total number of warnings allowed.

Cause: The number of warnings issued during the command has exceeded the total number of warnings that the user specified at invocation of the utility.

The utility terminates.

Action: Ensure that the correct data is being loaded with the appropriate options or increase the number of warnings allowed. Resubmit the command.

SQL3503W

The utility has loaded "<number>" rows which equals the total count specified by the user.

Cause: The number of rows loaded has equaled the total number of rows that the user specified at invocation of the utility.

The utility has successfully completed.

Action: No response required.

SQL3504W

A consistency point is being established.

Cause: Consistency points will be established at times other than the regular interval specified in the SAVECOUNT parameter at invocation when:

- the threshold for the amount of meta-data kept in memory or in temporary files has been reached.
- A device error occurred for the copy image and the load must terminate.

Action: If the LOAD continues after this message, no action is required. If the LOAD terminates, it can be restarted after all errors are corrected (such as activating a device or decreasing the interval at which consistency points are to be established).

SQL3505N

The length specified in the filetmod for the RECLEN option is not in the valid range of 1 to 32767.

Cause: In the *filetmod* parameter, the RECLEN option was specified for an ASC file. The length specified is not valid.

Action: Correct the length specified and resubmit the command.

SQL3506W

The value specified in the null indicator in row "<row-number>" and column "<column-number>" is not valid. A value of 'N' will be assumed.

Cause: For ASC files, a null indicator column can be specified for each data column and should contain either a 'Y' or a 'N'. A 'Y' indicates that the column is a null value and a 'N' indicates that the column contains data. If neither of these values are in the null indicator column, it is assumed to be a 'N' and data will be loaded into the column.

Action: If the data or the null indicator are incorrect, correct the input file and resubmit the command.

SQL3507N

The column number specified for a null indicator is not in the valid range of 0 to 32767 or the null indicator parameter is not valid.

Cause: In the *null_ind* parameter, columns for null indicators were specified for an ASC file but one of the columns is not valid or the pointer passed in for the null indicators is not valid.

Action: Correct the parameter and resubmit the command.

SQL3508N

Error in accessing a file of type "<file-type>" during load or load query. Reason code: "<reason-code>". Path: "<path/file>".

Cause: An error occurred while trying to access a file during load or load query processing. The utility stops processing.

Action: If doing a load and the tablespace is not in load pending state, correct the problem and invoke the

---

load utility again. If the tablespace is in load pending state, then invoke the load utility in RESTART or REPLACE mode, or restore a backup of the tablespace(s). The state of the tablespace can be determined using the LIST TABLESPACES command.

The following is a list of reason codes:

1   Unable to open the file.

This could be caused by an incorrect file name or insufficient authority to access the file/directory Correct the problem and either restart or rerun the load.

2   Unable to read/scan the file.

This could be the result of a hardware error. If the error is a hardware error, take the appropriate action as specified in the *Problem Determination Guide* for hardware problems and restart or rerun the load.

3   Unable to write to or change size of the file.

This could be the result of a disk full condition or a hardware error. Refer to the file type list below and either ensure there is enough space to run the load or specify a different location to be used. Restart or rerun the load. If the error is a hardware error, take the appropriate action as specified in the *Problem Determination Guide* for hardware problems and restart or rerun the load.

4   The file contains invalid data.

A file required by the load contains incorrect data. See the action described for REMOTE_FILE.

5   Unable to close the file.

If the load cannot be restarted or rerun, contact your IBM service representative.

6   Unable to delete the file.

If the load cannot be restarted or rerun, contact your IBM service representative.

7   Parameter specified incorrectly. Refer to the list of file types to determine the parameter in error and rerun the load with a valid parameter.

The following is a list of file types:

SORTDIRECTORY Ensure that the workdirectory parameter is specified properly. There must be enough combined space in all the directories to hold twice the size of the index keys for the loaded data. For load insert and load restart there must also be room for twice the size of the index keys of the existing data in the table.

MSGFILE Ensure that the messagefile parameter is specified properly. There must be enough disk space to write out the messages that occur during the load.

If this is a load query, ensure that the local message file parameter is NOT the same as the messagefile parameter used for the load whose status is being queried.

REMOTE_FILE Ensure that the remotefile parameter is specified properly. The default name for this parameter is db2utmp. See the *Admin Guide* for more details on this parameter.

One reason for getting this message could be that two loads are running using the same remote file (or defaulting to the same value). The remotefile parameter must be specified when running load in order to ensure that each load uses a different set of files.

If the file in the third token has the extension ".msg", make sure there is enough disk space to write out the message information for messages that occur during the load. If this is a load query, the remotefile parameter must be the same as the remotefile parameter used for the load whose status is being queried.

If the file in the third token has the extension ".rid", and this is a restart of a load in the delete phase, ensure that the remotefile parameter is the same as the first load. This file created from the original load is required by a load being restarted in the delete phase. If the correct remotefile parameter was used, it is possible that the file was removed or corrupted and the load must be restarted in the build phase.

If the file in the third token has the extension ".log", ensure that there is enough disk space to write temporary meta-data in this directory. If this is a RESTART, the remotefile parameter must be the same as the remotefile parameter used for the original load since this file which is created during the original load is required by load restart. If the correct remotefile parameter was used and sufficient disk space is available, it is possible that the logfile was removed or corrupted, and either a load replace must be done or the tablespace must be restored from an existing backup.

SQL3509W SQL3514N

SQL3509W

The utility has deleted "<number>" rows from the table.

Cause: When a table with a unique index is loaded, rows causing violation of uniqueness of an index will be deleted from the table during the delete phase. This message provides information on how many rows have been deleted.

Action: No action is required.

SQL3510N

 The working directory for the sort phase cannot be accessed.

Cause: One or more of the specified working directories for the sort phase either does not exist or has no read/write permission.

Action: Ensure that the specified working directories exist and the read/write permission is setup correctly and resubmit the command.

SQL3511W

The file named in row "<row-number>" and column "<column-number>" cannot be found. A null was loaded.

Cause: The filename in the specified field cannot be found. There may be a data type mismatch.

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

A null value is loaded.

Action: Examine the input value. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3512W

The file named in row "<row-number>" and column "<column-number>" cannot be found, but the target column is not nullable. The row was not loaded.

Cause: The filename in the specified field cannot be found. A null cannot be loaded because the output column in the table is not nullable.

For delimited ASCII (DEL) files, the value of the column number specifies the field within the row that contains the value in question. For ASCII files, the value of the column number specifies the byte location within the row where the value in question begins.

The row is not loaded.

Action: Correct the input file and resubmit the command or edit the data in the table.

SQL3513N

The codepage of the file does not match the codepage of the database. The file cannot be loaded.

Cause: DB2CS files cannot be loaded into a database with a codepage that differs from the original database.

Action: Change the codepage of the database and resubmit the command or use another filetype (such as PC/IXF) to move your data from the original database to the new database.

SQL3514N

A utility system error occurred. Function code: "<function>". Reason code: "<reason-code>". Error code: "<error-code>".

Cause: A system error occurred during database utility processing.

Action: Different actions are required depending on the value of "<function>".

The possible function codes are:

- 1 - An error occurred while Load was sorting.

  Try to restart the load. If the error persists, provide the function, reason code and error code to your technical service representative.

- 2 - An error occurred using the SMARTsort utility.

  Try the load again using the DB2 Client/Server sort utility instead of SmartSort. To do this, reset the

DB2SORT environment variable at the server to blank. You may have to restart the Database Manager in order to pick up the new value of DB2SORT. If the error does not recur, provide the function, reason code and error code to your IBM SmartSort technical service representative.

SQL3515W

The utility has finished the "<phase>" phase at time "<timestamp>".

Cause: This is an informational message indicating that a phase has finished. The phases are (in order of appearance):

- LOAD
- BUILD
- DELETE

Action: No action is required.

SQL3516N

The utility could not restart the load as specified.

Cause: The Load Utility detected discrepancies in the last consistency point performed by Load before failure occurred. This condition might be caused either by a system error or an invalid log file.

Action: Restart the Load at the Build phase, to bring the table to a consistent state and create indexes (if applicable), or perform Load with the REPLACE option.

SQL3517N

An unexpected record was read from the input source.

Cause: The utility has encountered a record that is in a format that is not valid. The source may have been corrupted when copied from the original source.

Processing terminates.

Action: Either regenerate the source using the UNLOAD that originally created it or recopy it from the original source in binary and restart the LOAD.

SQL3518N

The data in the source is not compatible to the table to be loaded.

Cause: The source cannot be used to load this table because one of the following reasons:

- the table definition does not match the table definition of the source.
- the source was created on a different platform than the table that is being loaded.
- the source was created from a table that had a different codepage from the table that is being loaded.

Action: Ensure that both the table and source are correctly specified. If you want to load data from a table with a different definition or from a different platform or codepage, use another filetype such as IXF or DEL.

SQL3519W

Begin Load Consistency Point. Input record count = "<count>".

Cause: The Load Utility is about to attempt to perform a Consistency Point to commit the table data that has already been loaded.

Action: If message SQL3520W is not printed immediately after this message, then the Consistency Point failed. The Load will have to be restarted at the Build phase, to ensure that the table is brought back to a consistent state and all indexes (if any) created. It will then be possible to check which records have been loaded. Start the Load again with a RESTARTCOUNT set to the number of records successfully loaded to skip those rows and load the rest of the file.

If this message is followed by messages SQL3520W, then this message is informational only and no response is required.

SQL3520W

Load Consistency Point was successful.

Cause: The Consistency Point performed by Load was successful.

Action: This is an informational message only. No response is required.

SQL3521N   SQL3600N

SQL3521N

The input source file "<sequence-num>" was not provided.

Cause: A load was invoked with multiple input files but not all the files were provided. For the DB2CS filetype, all the input source files that were created originally must be provided. For the IXF filetype, all the input source files must be provided in the correct order.

The utility terminates.

Action: Restart the utility providing all the input source files and setting the RESTARTCOUNT appropriately for the data that was already loaded.

SQL3522N

A copy target cannot be provided when both log retain and user exits are disabled.

Cause: A copy target has been specified on invocation of load for a database that has both log retain and user exits disabled. Copy targets are not valid for such databases.

The utility stops processing.

Action: Verify that the database should have log retain and user exits disabled and invoke the load without specifying a copy target.

SQL3523W

There are no messages to be retrieved from the message file. Reason code: "<rc>".

Cause: No messages are returned from a query of the remote file. The possible reason codes are:

1   the remote file does not exist.
2   there are no messages in the remote file.

Action: Ensure that the correct remote file has been specified. If the remote file is specified correctly and messages are expected, check the database monitor to verify that the utility is active and it is not waiting for resources such as locks. Note that the remote file will not be created until the utility is in progress and it will be deleted after the completion of the utility.

SQL3601W

Setting the constraints for a parent table to OFF in a referential structure or adding a foreign key when the parent table in check pending state has resulted in one or more descendent tables being automatically placed into a Check Pending state.

Cause: Setting a parent table in a referential structure requires dependent and descendent tables to be set in check pending. This is necessary to enforce referential integrity constraints. Adding a foreign key when the parent table is in a check pending state requires all new dependents and descendents of the parent table to be placed automatically in check pending.

Action: This is a warning message. The user must execute the SET CONSTRAINTS statement with the IMMEDIATE CHECKED option in order to check constraints on all the dependent and descendent tables as well.

sqlcode: +3601

sqlstate: 01586

SQL3602W

Check data processing found constraint violations and moved them to exception tables.

Cause: There are rows that violate constraints that were specified to be checked by the SET CONSTRAINTS statement execution. These were moved to exceptions tables.

Action: Check the exception tables for rows that violate constraints. The rows have been deleted from the original table but can be corrected and transferred back from the exception tables.

sqlcode: +3602

sqlstate: 01603

SQL3603N

Check data processing through the SET CONSTRAINTS statement has found that constraint "<constraint-name>" has been violated.

Cause: A row has been found to violate a constraint defined on a table specified to be checked by the SET CONSTRAINTS statement.

Action: The row has not been deleted from the table since the FOR EXCEPTION option was not used.

It is advisable to execute the SET CONSTRAINTS statement using the FOR EXCEPTION option when checking the data. The data may be corrected with the information from the exception table(s).

sqlcode: -3603

sqlstate: 23514

SQL3604N

Exception table "<excp-table-name>" corresponding to table "<table-name>" in the SET CONSTRAINTS statement does not have the proper structure, has been defined with unique indexes, constraints or triggers, or is in the check pending state itself.

Cause: The exception table corresponding to a table must have its definition similar to that of the original table. Optional columns for the utility are as specified in the relevant section of the documentation describing the exception tables. There must not be any constraints or triggers defined on the exception table. The exception table itself should not be in the check pending state.

Action: Create the exception table as indicated in the relevant section of the documentation and re-run the utility.

sqlcode: -3604

sqlstate: 428A5

SQL3605N

Table "<table-name>" named in the SET CONSTRAINTS statement is either not listed to be checked or is an exception table specified more than once.

Cause: When the FOR EXCEPTION clause is specified in the SET CONSTRAINTS statement, this error could be caused by one of the following :

- The table is missing from the list of tables to be checked

- The table to be checked is the same as an exception table

- The exception table is specified for more than one table to be checked

Action: Correct the table names and execute the command again.

sqlcode: -3605

sqlstate: 428A6

SQL3606N   SQL3702W

SQL3606N

There is a mismatch in the number of tables being checked and in the number of exception tables specified in the SET CONSTRAINTS statement.

Cause: There must be a one-to-one correspondence between the original tables and the exception tables provided in the list.

Action: Create the missing exception table if not already done and specify it in the list in order to execute the command again.

sqlcode: -3606

sqlstate: 428A7

SQL3608N

Cannot check or reset the Check Pending state on dependent table "<dep-table-name>" using the SET CONSTRAINTS statement while the parent table "<par-table-name>" is in the check pending state.

Cause: The parent table must be clean (not in the check pending state) or be included in the invocation list in order to

- reset the check pending state of a dependent table
- check a dependent table.

Action: Ensure that the parent table is not in check pending by executing the SET CONSTRAINT statement to check the parent table.

It is recommended to check the parent table first. It is also possible to check the dependent table and include the parent table in the invocation list. In this case, the command could still fail if there are constraint violations in the parent table and they are not deleted (ie., if the FOR EXCEPTION option is not used).

In the case of a referential cycle, all tables must be included in the invocation list.

sqlcode: -3608

sqlstate: 428A8

SQL3700W

Device "<device>" is full. There are "<active-devices>" other active devices. Please mount new media or take appropriate action.

Cause: The media on the specified device is full. This device is one of the "<active-devices>" + 1 target devices to which unloaded data is being directed.

Action: Take ONE of the following actions:

- Mount a new target media on the specified device and continue unloading by invoking unload with caller action 1 (SQLU_CONTINUE).

  OR

- If "<active-devices>" is not zero, continue the unload without this device by invoking the UNLOAD utility with the caller action 4 (SQLU_DEVICE_TERMINATE).

  OR

- Cancel the unload by invoking the UNLOAD utility with caller action 2 (SQLU_TERMINATE).

SQL3701W

The lobpaths parameter was specified, but the table does not contain any LOB or Long data. This parameter will be ignored.

Cause: The lobpaths parameter specifies a separate target for LOB and Long data. The table does not contain any LOB or Long data, so the targets specified by the lobpaths parameter will not be used.

Action: No action required.

SQL3702W

Warning. SQLCODE "<sqlcode>" was received for device "<device>". Unload will attempt to continue without this device.

Cause: SQLCODE "<sqlcode>" was detected for the specified device which is one of the targets for the unload. Unload will continue but will ignore this device.

Action: The media loaded on the specified device will not contain any unloaded data and should not be included with the media supplied to the LOAD utility when the unloaded data is being loaded. In order to fix the problem with the device, look up the returned SQLCODE in the Messages Reference.

SQL3703W

"<xxx>" of "<yyy>" number of type "<type>" pages have been unloaded and sent to be written out to the target media.

Cause: The table being unloaded consists of "<yyy>" pages of the specified type of data. "<xxx>" of these have been processed by the UNLOAD utility and have been sent to a media writer which will write the data to the target media.

"<type>" is one of:

- 0 for regular data
- 2 for Long data and allocation information
- 3 for LOB data
- 4 for LOB allocation information

Note that for Long and LOB data, "<xxx>" may be less than "<yyy>" when unload is complete, because unused space is not unloaded, but will be recreated when the data is reloaded.

Even for regular data, a final message where "<xxx>" = "<yyy>" may not be issued. Instead message 3105 is used to indicate that the unload completed successfully.

Action: This is an informational message. No action is required.

SQL3704N

The num_buffers parameter specified is invalid.

Cause: The num_buffers parameter determines the number of buffers that the utility will use. The minimum is 2 if the lobpaths parameter is not specified and 3 if the lobpaths parameter is specified. This is the minimum required to allow the utility to work. There is, however, an optimal number of buffers that the utility will use if this parameter is not specified. This optimal number is based on the number of internal processes that the utility will have running and whether or not the lobpaths parameter was specified. If the number of buffers specified is less than the optimal number, some processes will be waiting for buffers to use. Therefore, it is recommended to specify 0 for this parameter and let the utility choose the number of buffers. Only specify this parameter if, due to the size of the utility storage heap, it is necessary to limit the amount of memory that the utility uses.

Action: Resubmit the command using a valid num_buffers parameter.

SQL3705N

The buffer size parameter specified is not valid. The buffer size must be specified as 0 or be between 8 and 16384 inclusive. For multiple buffers, the total buffer size must not exceed 16384.

Cause: The application calling the utility has supplied a buffer size parameter that is not valid. The buffer size is used to determine the internal buffer size. The value is the number of 4K pages that are obtained for this buffer. The value must be specified as 0 or be between 8 and 16384 inclusive. If there are multiple buffers, the number of buffers multiplied by the buffer size must not exceed 16834.

When 0 is specified:

- for a table whose regular data is in a database managed storage table space, the default buffer size chosen is the extent size for the table space or 8, whichever is larger.
- for a table whose regular data is in a system managed storage table space, the default buffer size chosen is 8.

Action: Reissue the command with a valid buffer size.

SQL3706N

A disk full error was encountered on "<path/file>".

Cause: A disk full error was encountered during the processing of a database utility. The utility stops.

Action: Ensure enough disk space is available for the utility or direct the output to other media such as tape.

SQL3707N

The sort memory size parameter specified "<size1>" is not valid. The minimum size required is "<size2>".

Cause: The sort memory size is not large enough for sorting the keys of the index(es).

Action: Reissue the command with a valid sort memory size.

To ensure that only the minimum amount of storage is used, specify a value of 0 (to let it default). However, using more than the minimum should improve the performance of the sort.

SQL3783N SQL3805N

SQL3783N

An error was encountered while opening the copy location file. Open file error code is "<errcode>".

Cause: During load recovery, an error was encountered while opening the copy location file. The operating system file open return code is returned.

Action: Look at the error code in the message for more information. Correct the error, if possible, and resubmit the command.

SQL3784W

Invalid data was encountered while reading from the copy location file. Failure occurred at line "<line-no>" with error type "<errtype>"

Cause: During load recovery, invalid data was encountered while reading from the copy location file. The line number and error type are returned. The utility waits for a response to continue.

Action: Correct the data in the copy location file and return to the utility with the correct caller action parameter to indicate if processing should continue or terminate.

SQL3785N

Load recovery for table "<schema.tablename>" at time "<timestamp>" failed due to error "<sqlcode>" with additional information "<additional-info>".

Cause: A severe error occurred during load recovery. The utility stops processing.

Action: Examine the error code in the message for more information. Take corrective action and resubmit the command.

SQL3798W

An invalid value was used for parameter "<parameter>" to invoke the roll forward recovery API to continue load recovery.

Cause: Load recovery is in progress and one of the parameters passed in is not valid for the current state of load recovery.

Action: Correct the value in error and return to the utility with the correct caller action parameter to indicate if processing should continue or terminate.

SQL3799W

Load recovery for table "<schema.tablename>" at time "<timestamp>" is pending due to warning "<sqlcode>" with additional information "<additional-info>".

Cause: A warning condition is encountered during load recovery. The utility waits for a response to continue.

Action: Examine the error code in the message for more information. Take corrective action and return to the utility with the correct caller action parameter to indicate if processing should continue or terminate.

SQL3802N

An invalid quiesce mode "<quiesce-mode>" was encountered.

Cause: An invalid quiesce mode was passed to the quiesce API.

Action: Resubmit the command with the correct parameters.

SQL3804N

The index is invalid.

Cause: An invalid index was encountered during the processing of a utility command.

Action: Check in the *Administration Guide* to determine the appropriate way to revalidate the index and resubmit the command after the condition is corrected.

SQL3805N

The state of the application or of one or more table spaces for the table specified prohibits the loadapi action or quiescemode "<action>" Reason code = "<reason-code>".

Cause: The loadapi action (either quiescemode or callerac) passed to the load API is in conflict with either the state of the application or the state of one or more table spaces for the table.

The possible reason codes are as follows:

01   the state of one of the table spaces for the table specified prohibits the loadapi action or quiescemode.

02   the application is not at the start of its logical unit of work. This condition prohibits the load action specified.

03   the state of the application prohibits the load or unload action specified.

04   one or more of the table spaces for the table is already quiesced by the maximum number of quiescers.

05   system catalog tablespace cannot be quiesced.

06   load copy not allowed when tablespace is in backup pending state.

Action: Corresponding to each of the above reasons:

- Resubmit the command with the correct loadapi action or quiescemode, or correct the state of the table spaces for the table.
- Resubmit the command with the correct load action or complete the current logical unit of work by issuing either a COMMIT or ROLLBACK.
- Resubmit the command with the correct load or unload action.
- Determine which table spaces for the table have reached the maximum number of quiescers. QUIESCE RESET these table spaces.
- Resubmit the command with a table which is not in the system catalog tablespace
- Resubmit the command omitting the copy parameter

SQL3806N

Not all table constraints are off for the table to be loaded.

Cause: One or more constraints for the table to be loaded are still on when the load API is invoked.

Action: Resubmit the command after all table constraints are turned off.

SQL4001N

The character "<character>" in column "<column>" on line "<line>" is not valid.

Cause: The specified character is not a valid character in SQL statements.

The statement cannot be processed.

Action: Remove or replace the character that is invalid.

SQL4002N

"<token 1>" and "<token 2>" are undeclared host variables that cannot both be used as descriptor names in a single SQL statement.

Cause: The specified identifiers have not been declared as host variables. Descriptor names are not declared before using. Because more than one descriptor name in a single statement is not valid, at least one of the host variables is not valid.

The statement cannot be processed.

Action: Correct the use of descriptor names or undeclared host variables in the statement. Note that the statement can contain additional undeclared host variables.

SQL4003N

The SQL statement on line "<line>" is not supported by this version of the precompiler.

Cause: The release numbers of the precompiler and the installed version of the database manager are not compatible. The specified statement is supported by the database manager but not the precompiler.

The statement cannot be processed.

Action: Repeat the precompile process with the current version of the precompiler.

SQL4004N

The package name is not valid.

Cause: The package name contains characters that are not valid. Either the name is too long or no name was specified with the PACKAGE option.

No package is created.

Action: Resubmit the command with a valid package name or without the PACKAGE option.

SQL4005N

An invalid token "<token>" was found starting in position "<position>" on line "<line>".

Cause: A syntax error in the SQL statement was detected at the specified token "<token>".

The statement cannot be processed.

Action: Examine the statement, especially around the specified token. Correct the syntax.

SQL4010N SQL4102W

SQL4010N

Illegal nesting of Compound SQL statements.

Cause: This error is returned when a BEGIN COMPOUND clause is detected as a sub-statement of a Compound SQL statement.

Action: Resubmit the precompilation without the nested BEGIN COMPOUND.

SQL4011N

Invalid SQL sub-statement in a Compound SQL statement.

Cause: This error is returned when an invalid sub-statement is detected in a Compound SQL statement. Valid statements are:

- ALTER TABLE
- COMMENT ON
- CREATE INDEX
- CREATE TABLE
- CREATE VIEW
- Positioned DELETE
- Searched DELETE
- DROP
- GRANT
- INSERT
- LOCK TABLE
- REVOKE
- SELECT INTO
- Positioned UPDATE
- Searched UPDATE
- COMMIT; only when specified as the last sub-statement.

Action: Resubmit the precompilation without the invalid sub-statement.

SQL4012N

Invalid use of COMMIT in a Compound SQL statement.

Cause: This error is returned when a sub-statement is detected after a COMMIT has been found in a COMPOUND SQL statement.

Action: Make the COMMIT sub-statement the last sub-statement, and resubmit the precompilation.

SQL4013N

An END COMPOUND statement was found without a previous BEGIN COMPOUND statement.

Cause: This error is returned when an END COMPOUND statement has been found without a preceding BEGIN COMPOUND.

Action: Either remove the END COMPOUND, or add a BEGIN COMPOUND, and resubmit the precompilation.

SQL4014N

Invalid Compound SQL syntax.

Cause: This error is returned when a Compound SQL statement contains a syntax error. Possible causes include:

- END COMPOUND is missing

- one of the substatements is empty (zero length or blank).

Action: Fix the syntax error and re-precompile.

SQL4140W

An error has occurred causing the flagger to stop. Module name = "<module-name>". Internal error code = "<error code>".

Cause: The flagger has encountered an internal error. If the syntax, bindfile or package option is also specified, processing continues but flagger operation is discontinued. Otherwise, processing is terminated.

Action: Record this message number (SQLCODE), module name and error code in the message. Contact your technical service representative with the information.

SQL4142W

There is not enough memory for flagger operation. Internal error code = "<error code>".

Cause: There is not enough memory for the flagger to operate. If the syntax, bindfile or package option is also specified, processing continues but flagger operation is discontinued. Otherwise, processing is terminated.

Action: Ensure that your system has sufficient real and virtual memory, and remove any unnecessary background processes.

SQL4143W

An error has occurred when releasing flagger memory. Internal error code = "<error code>".

Cause: The flagger cannot release allocated memory. If the syntax, bindfile or package option is also specified, processing continues but flagger operation is discontinued. Otherwise, processing is terminated.

Action: Restart precompilation if flagging is required.

SQL4147W

Invalid flagger version number.

Cause: An invalid flagger version number is passed to the Precompiler Services COMPILE SQL STATEMENT API. If the syntax, bindfile or package option is also specified, processing continues but flagger operation is discontinued. Otherwise, processing is terminated.

Action: Provide a valid flagger version number. Refer to the *Command Reference* manual.

SQL4300N

Java support is not installed on this platform.

Cause: The Java stored procedure / user defined function enablement is not installed on this server.

Action: If Java support is available on this platform, install the Java stored procedure / user defined function enablement module.

sqlcode: -4300

sqlstate: 42884

SQL4301N

Java interpreter startup, communication, or shutdown failed.

Cause: A severe error occurred while attempting to start, stop or communicate with a Java interpreter.

Action: Check the First Failure Service Log (db2diag.log) for additional information.

sqlcode: -4301

sqlstate: 58004

SQL4302N

Java stored procedure or user defined function "<name>": unclassified exception: "<message>"

Cause: Java stored procedure or user defined function "<name>" encountered unclassified exception "<message>".

Action: Check the First Failure Service Log (db2diag.log) for additional information. Debug the Java routine causing the exception.

sqlcode: -4302

sqlstate: 58004

SQL4303N

Java stored procedure or user defined function "<name>" could not parse class!method in "<string>".

Cause: A stored procedure or user defined function was declared as being implemented by a Java class and method whose names were not properly formatted in the catalogs. The external name must be formatted as follows: "package.subpackage.class!method".

Action: Check the DB2CLI.PROCEDURES or SYSIBM.SYSFUNCTIONS table for proper syntax.

sqlcode: -4303

sqlstate: 42884

SQL4304N

Java stored procedure or user defined function "<name>" could not load or instantiate class "<class>".

Cause: The Java class that implements the given stored procedure or user defined function was not found in the sqllib/function directory, or for user defined functions only, did not have a public default constructor.

Action: Install the Java ".class" file into the appropriate directory under sqllib/function. Ensure that if it is a user defined function, it has a "public" default constructor. Ensure that the Java class is declared "public".

sqlcode: -4304

sqlstate: 42884

SQL4305N

Java stored procedure or user defined function "<name>" internal error code "<code>".

Cause: A serious internal error occurred while evaluating the named stored procedure or user defined function.

Action: Check the First Failure Service Log (db2diag.log) for additional information.

sqlcode: -4305

sqlstate: 38503

SQL4306N

Java stored procedure or user defined function "<name>" cannot call matching method "<method-name>".

Cause: A Java method could not be located in the class that implements the named stored procedure or user defined function. The declared argument list may be incorrect or the Java access flags were incorrect.

Action: Check the First Failure Service Log (db2diag.log) for the full class and method name. Ensure that the Java method is declared to take the proper argument types. For stored procedures, the method must be "static public" and for user defined functions, non-static "public".

sqlcode: -4306

sqlstate: 42884

SQL4307N

Java stored procedure or user defined function "<name>" method call problems, reason code "<reason-code>".

Cause: An error occurred in building a Java method's argument list, or accepting its result value. The possible reason codes are as follows:

1   more than 90 arguments for a stored procedure
2   stored procedure result slot number was out of range
3   attempted to write Java null to non-Null result slot
4   attempted to write incompatible Java type into slot
5   attempted to write out-of-range value into slot
6   encountered error while converting Java value to slot
7   encountered error while computing Java data type mapping for user defined argument or result
8   attempted to write incompatible Java type into slot
9   attempted to write out-of-range value into slot

Action: Check the First Failure Service Log (db2diag.log) for additional information. Debug the Java program to return the appropriate results.

sqlcode: -4307

sqlstate: 38501

SQL4308N

DB2CLI.PROCEDURES pseudo-catalog table does not have the required schema, reason code "<reason-code>".

Cause: The DB2CLI.PROCEDURES pseudo-catalog table must be of the form documented for Java stored procedures to operate. The table does not have that form. Possible reason codes are as follows:

1   the number of expected columns is incorrect
2   the PROCNAME column is in error
3   the PROC_LOCATION column is in error
4   the LANGUAGE column is in error
5   the FENCED column is in error

Action: Recreate the DB2CLI.PROCEDURES table according to the pseudo DDL as given in the documentation.

sqlcode: -4308

sqlstate: 42884

SQL4901N

Precompiler Services requires reinitialization because of a previous error.

Cause: On a previous function call, an error occurred. The requested function call cannot be processed until Precompiler Services is reinitialized.

The function cannot be completed.

Action: Call the sqlainit function to reinitialize Precompiler Services

---

SQL4902N

At least one of the characters in parameter "<n>" of function "<function>" is not valid.

Cause: The specified parameter in the specified function contains at least one character that is not valid.

The function cannot be completed.

Action: Correct the specified parameter and call the function again.

SQL4903N

The length of parameter "<n>" of function "<name>" is not valid.

Cause: The length of the specified parameter in the specified function is not valid.

The function cannot be completed.

Action: Correct the specified parameter and call the function again.

SQL4904N

The pointer to parameter "<n>" of function "<function>" is not valid.

Cause: The pointer to the specified parameter in the specified function is not valid.

The function cannot be completed.

Action: Correct the specified parameter and call the function again.

SQL4905N

The value of parameter "<n>" of function "<function>" is not within the valid range.

Cause: The value of the specified parameter in the specified function is outside the valid range for that parameter. If the specified parameter is a structure, it may contain values that are within the valid range but are not valid when considered together. Some structures contain headers that specify the allocated size and how much of that size is being used. It is not valid for the allocated size to be less than the used size.

The function cannot be completed.

Action: Correct the specified parameter and call the function again.

SQL4911N

The host variable data type is not valid.

Cause: The data type of the host variable is not valid.

The function cannot be completed.

Action: Correct the data type of the host variable and call the function again.

SQL4912N

The host variable data length is out of range.

Cause: The length of the host variable is not valid.

The function cannot be completed.

Action: Correct the length of the host variable and call the function again.

SQL4913N

The host variable token ID has already been used.

Cause: The token ID of the host variable has already been used. The token ID must be unique within the module.

The function cannot be completed.

Action: Correct the token ID of the host variable and call the function again.

SQL4914N   SQL4940N

SQL4914N

The host variable token ID is not valid.

Cause: The token ID of the host variable is not valid.

The function cannot be completed.

Action: Correct the token ID of the host variable and call the function again.

SQL4915N

The "sqlainit" function has already been called.

Cause: Precompiler Services has already been initialized.

Action: No action is required. Continue processing.

SQL4916N

The "sqlainit" function has not been called.

Cause: Precompiler Services must be initialized before the requested function call can be processed.

The function cannot be completed.

Action: Issue an sqlainit function call to initialize Precompiler Services.

SQL4917N

Element "<number>" in the option array is not valid.

Cause: The option array contains an element with an *option.type* or *option.value* that is not valid. The element number in the message is the *n*th element in the option portion of the option array.

The function cannot be completed.

Action: Correct the values stored in the option array. Call the function again.

SQL4918N

The term_option parameter of function "sqlainit" is not valid.

Cause: The *term_option* parameter is not valid.

The function cannot be completed.

Action: Correct the *term_option* parameter and call the function again.

SQL4919N

The task_array parameter of function "sqlacmpl" is too small.

Cause: The task array structure passed to Precompiler Services on the sqlacmpl function call is too short.

The function did not complete successfully.

Action: Increase the size of the precompiler task array structure allocated by the precompiler. Recompile the application program.

SQL4920N

The token_id_array parameter of function "sqlacmpl" is too small.

Cause: The token id array structure passed to Precompiler Services on the sqlacmpl() function call is too small.

The function did not complete successfully.

Action: Increase the size of the precompiler token id array structure allocated by the precompiler. Recompile the application program.

SQL4930N

Bind or precompile option or option value "<option-name>" is invalid.

Cause: Either "<option-name>" is an invalid bind or precompile option or the value specified for this option is invalid. The bind or precompile cannot continue.

Action: Correct the bind or precompile option or option value and retry the bind or precompile command.

SQL4940N

The "<clause>" clause is not permitted or is required.

Cause: The indicated clause is either not allowed in the context where it appears in the SQL statement or it is required in the statement.

A subquery, an INSERT statement, or a CREATE VIEW statement cannot have INTO, ORDER BY, or FOR UPDATE clauses. An embedded SELECT statement cannot have ORDER BY or FOR UPDATE clauses. An embedded SELECT statement cannot contain a set operator except in a subquery. SELECT statements used in cursor declarations cannot have an INTO clause.

An embedded SELECT statement must have an INTO clause.

The function cannot be completed.

Action: Remove or add the clause to correct the statement.

SQL4941N

The SQL statement is blank or empty.

Cause: The text following EXEC SQL was blank or empty.

The function cannot be completed.

Action: Ensure that a valid SQL statement is provided following the word EXEC SQL.

SQL4942N

The statement selects an incompatible data type into host variable "<name>".

Cause: An embedded SELECT statement selects into a host variable "<name>" but the data type of the variable and the corresponding SELECT list element are not compatible. If the data type of the column is date and time, the data type of the variable must be character with an appropriate minimum length. Both must either be numeric, character, or graphic.

The function cannot be completed.

Action: Verify that the table definitions are current and that the host variable has the proper data type.

SQL4943W

The number of host variables in the INTO clause is not the same as the number of items in the SELECT clause.

Cause: The number of host variables specified in both the INTO clause and the SELECT clause must be the same.

The function is processed.

Action: Correct the application program to specify the same number of host variables as SELECT list expressions.

SQL4944N

An update or insert value is NULL but the object column cannot contain NULL values.

Cause: One of the following occurred:

- The update or insert value was NULL but the object column is declared as NOT NULL in the table definition. Therefore, null values cannot be inserted into that column, and values in that column cannot be set to NULL by an update.
- The column name list for the INSERT statement omits a column that was declared as NOT NULL in the table definition.
- The view for the INSERT statement omits a column that was declared as NOT NULL in the base table definition.

The function cannot be completed.

Action: Examine the definition of the object table to determine which columns of the table have the NOT NULL attribute, and correct the SQL statement.

SQL4945N

The use of a parameter marker is not valid.

Cause: Parameter markers can be used only in dynamic SQL statements.

The function cannot be completed.

Action: Use host variables instead of parameter markers for static SQL statements.

SQL4946N

The cursor or statement name "<name>" is not defined.

Cause: The cursor or statement name "<name>" specified in the statement is not defined.

The function cannot be completed.

Action: Check the application program for completeness and possible spelling errors in the cursor or statement names.

SQL4947W · SQL4997N

SQL4947W

An INCLUDE SQLDA statement was encountered and ignored.

Cause: The FORTRAN precompiler shipped by the database manager does not support the INCLUDE SQLDA statement.

The statement is ignored. Processing continues.

Action: No action is required. To prevent this message, remove the INCLUDE SQLDA statement from the program.

SQL4950N

Compound SQL statements containing user-defined SQLDAs are not supported in this environment.

Cause: Compound SQL statements containing user-defined SQLDAs are not supported in a 16-bit application.

Action: Move the statement out of the compound SQL block or replace the statement with one that uses host variables instead of an SQLDA.

SQL4951N

The sqlda_id parameter of function "<name>" is not valid.

Cause: The *sqlda_id* parameter of the specified function in the application program is not valid. The *sqlda_id* parameter cannot be null.

The function cannot be completed.

Action: Correct the *sqlda_id* parameter in the application program.

SQL4952N

The sqlvar_index parameter of function "<name>" is not valid.

Cause: The *sqlvar_index* parameter of the specified function in the application program is not valid. The *sqlvar_index* may be larger than the number of sqlvar elements in the SQLDA.

The function cannot be completed.

Action: Correct the *sqlvar_index* parameter in the application program.

SQL4953N

The call_type parameter of function "<name>" is not valid.

Cause: The *call_type* parameter of the specified function in the application is not valid.

The function cannot be completed.

Action: Correct the *call_type* parameter in the application program.

SQL4954N

The section_number parameter of function "<name>" is not valid.

Cause: The *section_number* parameter of the specified function in the application program is not valid. Note that for the SQL CONNECT statements, the section_number parameter of the function sqlacall() is used to pass in the statement type.

The function cannot be completed.

Action: Correct the *section_number* parameter in the application program.

SQL4994N

Precompilation was ended because of a user interrupt request.

Cause: Precompilation was terminated because of an interrupt. The user may have pressed an interrupt key sequence.

Processing is terminated. No package was created.

Action: Resubmit the precompilation if needed.

SQL4997N

The authorization ID is not valid.

Cause: The authorization ID for the application was specified previously but it was defined with more than 8 characters or with characters that are not valid for an authorization ID.

The authorization ID must not be PUBLIC (public) and it must not begin with SYS (sys), IBM (ibm), or SQL (sql). Also, it must not contain the underscore character or any characters outside the database manager base character set.

The function cannot be processed.

Action: Retry the application with a valid authorization ID.

SQL4998C

The application state is in error; the database connection has been lost.

Cause: The connection to the database has been lost.

The function cannot be processed.

Action: Reconnect to the database.

SQL4999N

A Precompiler Services or Run-Time Services error occurred.

Cause: A database manager error occurred that prevents Precompiler Services or Run-Time Services from processing function calls.

No Precompiler Services or Run-Time Services function calls can be processed.

Action: Record the message number (SQLCODE) and all error information form the SQLCA if possible.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

- Environment: Outer Precompiler Using Precompiler Services API
- Required information:
  - Problem description
  - SQLCODE
  - SQLCA contents if possible
  - Trace file if possible.

SQL5001N

"<authorization-ID>" does not have the authority to change the database manager configuration file.

Cause: The user attempted to Update or Reset the database manager configuration file without having SYSADM authority.

The requested change is not made.

Action: Do not attempt to change the database manager configuration file without appropriate authorization. Contact a user with SYSADM authority if a change is required.

SQL5005C

System Error.

Cause: An operating system error, probably an I/O error, occurred.

The command cannot be processed.

Action: Resubmit the command. If the error continues while accessing the database manager configuration file, either use the backup version of the file or reinstall the database manager. If the error continues while accessing a database configuration file, drop the database and restore it from a backup version.

SQL5010N

The path name to the database manager configuration file is not valid.

Cause: An error occurred while determining the path for the database manager configuration file. The database manager directory structure may have been altered.

The command cannot be processed.

Action: Resubmit the command. If the error continues, reinstall the database manager.

SQL5012N

The entry in the database manager configuration file for the maximum number of segments allowed for the DB2 kernel is not in the valid range.

Cause: The value for the maximum number of segments allowed for the DB2 kernel must be between 8 and 8192.

The requested change is not made.

Action: Resubmit the command with a valid value for the maximum number of DB2 segments allowed.

SQL5018N

The entry in the database manager configuration file for the maximum number of remote connections to the workstation (numrc) is not in the valid range.

Cause: The value for the maximum number of remote connections to the workstation must be between 1 and 255.

The requested change is not made.

SQL5020N · SQL5030C

Action: Resubmit the command with a valid value for the maximum number of remote connections to the workstation.

SQL5020N

The entry in the database manager configuration file for the node name for the workstation (nname) is not valid.

Cause: The node name specified in the configuration command is not valid. The node name may be 1 to 8 characters. All the characters must be chosen from the database manager base character set.

The command cannot be processed.

Action: Resubmit the command with a valid node name.

SQL5021N

The entry in the database manager configuration file for the index recreation time (indexrec) is not valid. Possible values are 1 (during index access) and 2 (during database restart).

Cause: An invalid value was passed to Configuration Services for the Index Recreation Time flag (indexrec) in the database manager configuration file. The invalid value may have been entered through the Command Line Processor or through a program API call. The only valid values for the API call are 1 (during database access), and 2 (during database restart). The valid values for the command line processor are ACCESS and RESTART.

The update to the database manager configuration file is rejected.

Action: Repeat the update request, entering one of the possible values.

SQL5022N

The entry in the database configuration file for the index recreation time (indexrec) is not valid. Possible values are 0 (use system setting), 1 (during index access), and 2 (during database restart).

Cause: An invalid value was passed to Configuration Services for the Index Recreation Time flag (indexrec) in a database configuration file. The invalid value may have been entered through the Command Line Processor or through a program API call. The only valid

values for the API call are 0 (use system setting), 1 (during index access), and 2 (during database restart).

The valid values for the command line processor are SYSTEM, ACCESS, and RESTART.

The update to the database configuration file is rejected.

Action: Repeat the update request, entering one of the possible values.

SQL5025C

The database manager configuration file is not current.

Cause: The database manager configuration file has been updated since you connected to the database. It is not compatible with the configuration for the database to which you are connected.

No access is allowed to the database manager configuration file.

Action: Wait until all applications disconnect from the databases. Issue the stop database manager command, and then issue the start database manager command.

If installing the sample database, drop it and install the sample database again.

SQL5028N

The value of sysadm_group must be the primary group of the instance owner.

Cause: An attempt was made to update sysadm_group in the database manager configuration file. For Version 2 of DB2 on Unix-based platforms, this value is only allowed to be the primary group of the instance owner.

Action: Use the primary group of the instance owner to update this field in the database manager configuration file on Unix-based platforms.

SQL5030C

The release number is not valid.

Cause: The release number in the database manager configuration file or database configuration file is not valid.

The command cannot be processed.

Action: Verify that the release levels of your database and DB2 match.

If installing the sample database, drop it and install the sample database again.

sqlcode: -5030

sqlstate: 58031

SQL5035N

The database requires migration to the current release.

Cause: The database was created on a lower level system release. The Migrate Database command must be used to convert the database to the current release level.

The command cannot be processed.

Action: Perform the Migrate Database command before attempting to use the database with the current system release.

If this message is received during a restore, drop the existing database before proceeding.

sqlcode: -5035

sqlstate: 55001

SQL5043N

Start database manager processing was successful. However, support for one or more communication protocols failed to start successfully.

Cause: Communication protocol support did not start successfully for one or more protocols. Possible reasons can include the following:

- Communication subsystem configuration error.
- Communication subsystem call failure.
- Database manager configuration error.
- System call failure.
- Database manager licensing error.

You can still connect to the server with communication protocols that did start successfully. Local clients can also connect to the server.

Action: The database manager attempts to start all communication protocols specified by the DB2COMM environment variable. Examine the database manager service log. The log will contain more details on the cause of the error and also which communication protocol(s) did not start successfully.

This error only affects communication protocols specified by the DB2COMM environment variable.

SQL5047C

Insufficient memory to execute this function.

Cause: There is not enough memory available to execute this function.

The function cannot be completed.

Action: Stop the application. Possible solutions include:

- Terminate other processes.
- Decrease the values of the configuration parameters that define allocation of memory.
- Ensure that your system has sufficient real and virtual memory.

SQL5048N

The release level of the database client is not supported by the release level of the database server.

Cause: Database clients can only access database servers with a release level which is in the range of one level lower than the client to two levels higher than the client.

Action: One or more of the following:

- Upgrade the client release level to fall within the range of releases supported by the current server release level.
- Upgrade the server release level to a level supported by the current client release level.

SQL5050C

The content of the database manager configuration file is not valid.

Cause: The database manager configuration file is not valid. The file may have been altered by a text editor or a program other than the database manager.

The command cannot be processed.

Action: Reinstall the database manager.

SQL5055C SQL5075N

SQL5055C

The content of the database configuration file is not valid.

Cause: The database configuration file for the database is not valid. The file may have been altered by a text editor or a program other than the database manager.

Action: Recreate the database or restore it from a backup version.

sqlcode: -5055

sqlstate: 58031

SQL5060N

The configuration parameter token specified is invalid.

Cause: A token number specified in the sqlfupd structure passed to a Configuration Services API is not valid. It does not represent any of the supported configuration parameters. Alternatively, if an UPDATE was attempted, the token specified may be that of an unmodifiable configuration parameter.

Action: Choose a valid token number from those specified in the description of the Configuration Services API in the *API Reference*. Recode the call to the API, and rerun the program.

SQL5061N

An invalid pointer to structure sqlfupd was passed to Configuration Services.

Cause: The pointer to structure sqlfupd passed as a parameter to one of the Configuration Services APIs was invalid. It is either NULL, or does not point to an allocated block of memory of the size indicated by the count parameter. See the *API Reference* for more information.

Action: Correct the code which invokes the Configuration Services API, and retry the API call.

SQL5062N

An invalid pointer was passed within the sqlfupd structure to Configuration Services.

Cause: The structure sqlfupd passed as a parameter to one of the Configuration Services APIs contained an invalid pointer. The pointer is either NULL, or does not point to an allocated block of memory. Each token passed in the structure must have a corresponding pointer to a field that is either passed in or out of the API. See the *API Reference* for more information.

Action: Correct the code which invokes Configuration Services, and rerun the program.

SQL5065C

The nodetype value in database manager configuration file is not valid.

Cause: The *nodetype* parameter in the database manager configuration file is not valid.

The command cannot be processed.

Action: Reinstall the database manager.

SQL5070N

The count parameter in the configuration command is not valid. It must be greater than 0.

Cause: The count value passed as a parameter to a Configuration Services API must be greater than 0.

The command cannot be processed.

Action: Correct the code which invokes Configuration Services, and rerun the program.

SQL5075N

The configuration utility was interrupted.

Cause: An interrupt was received by the configuration utility. The user may have pressed the interrupt key sequence.

The command was not processed. Any requested change is not made.

Action: Resubmit the command.

SQL5081N

The entry in the database configuration file for the size of the buffer pool (buffpage) is not in the valid range.

Cause: The minimum value for the size of the buffer pool is two times the maximum number of active processes (maxappls). The maximum value for the size of the buffer pool is 524288 (number of 4KB pages) and is operating system dependent. The maximum value under AIX is 51000 (4KB pages). Under HP-UX, the value must be between 16 and 150000 (4KB pages).

The requested change is not made.

Action: Resubmit the command with a valid value for the size of the buffer pool.

SQL5083N

The entry in the database configuration file for the initial log file size (logfile) is not in the valid range.

Cause: The value for the initial log file size must be between 12 and (2**32 – 1).

The requested change is not made.

Action: Resubmit the command with a valid value for the initial log file size.

SQL5091N

The entry in the database configuration file for the size of one log file extension (logext) is not in the valid range.

Cause: The value for the size of one log file extension must be between 4 and 256.

The requested change is not made.

Action: Resubmit the command with a valid value for the size of one log file extension.

SQL5092N

The entry in the database configuration file for the maximum allowable number of log file extensions (logmaxext) is not in the valid range.

Cause: The value for the maximum allowable number of log file extensions must be between 0 and (2 * 10**6).

The requested change is not made.

Action: Resubmit the command with a valid value for the maximum allowable number of log file extensions.

SQL5093N

The entry for the database configuration parameter for the size of the agent heap is not in the valid range.

Cause: The value for the size of the agent heap must be between 2 and 85.

The requested change is not made.

Action: Resubmit the command with a valid value for the agent heap size.

SQL5099N

The new path to the log (newlogpath) in the database configuration file is not valid.

Cause: The path to the log file is not valid for one of the following reasons:

- The path string is longer than 242 bytes.
- The path does not exist.
- The first directory in the path has a name in the form SQLNNNNN, where NNNNN is a value from 00001 to 99999.
- A file with the correct name was found in the specified path, but is not a log file for this database.

The requested change is not made.

Action: Resubmit the command with a valid value for the new log path.

SQL5100N

The entry in the database manager configuration file for the number of allowed concurrent databases is too large.

Cause: The requested change would cause (1) the number of allowed concurrent databases to be too large or (2) the number of segments allowed for DB2 to be too small.

The number of allowed concurrent databases is restricted by the maximum number of segments allowed for DB2. The following condition must always be true:

```
segments >= ((number of databases * 5) + 1)
```

The requested change is not made.

Action: Do one of the following:

- Increase the maximum number of segments allowed for DB2.
- Decrease the number of allowed concurrent databases.

SQL5101N

The entries in the database configuration file define log file parameters (logprimary and logsecond) that are not in the valid range.

Cause: The requested change would cause the total number of logfiles to be out of range. The following condition must always be true:

```
logprimary + logsecond <= 128
```

The requested change is not made.

Action: Do one or both of the following:

- Decrease the number of primary log files.
- Decrease the number of secondary log files.

SQL5103N

The entry in the database configuration file for the size of the buffer pool (buffpage) is too small for the maximum number of active applications (maxappls).

Cause: The requested change would cause the maximum number of active applications to be too large for the size of the buffer pool. The following condition must always be true:

```
bufferpool_size > (number of active_processes * 2)
```

The requested change is not made.

Action: Do one or both of the following:

- Increase the size of the buffer pool.
- Decrease the maximum number of active processes allowed.

SQL5112N

The entry in the database configuration file for one or more database attribute flags must be either 0 or 1.

Cause: A request was made to modify one of the database attribute flags (SQLF_DETS) with an invalid value. The valid values are 1 to enable and 0 to disable the flag.

The command cannot be processed.

Action: Resubmit the command with a valid value.

SQL5120N

Old and new log parameters cannot both be modified at the same time.

Cause: The user is attempting to modify both the previous log parameters and the new parameters. The application should only support the parameters of the current release.

The request is denied.

Action: Modify only the parameters of the current release and retry the command.

SQL5121N

The entry in the database configuration file for configuration options is not valid.

Cause: The value set for database options (SQLF_DETS) is not valid. Valid settings are between 0 and 15. The requested change is not made.

The requested change is not made.

Action: Resubmit the command with a valid database options value.

SQL5122N

Access to the database was invalid because of a machine dependent check.

Cause: The database and database configuration file cannot be accessed because of copy protection.

The user request is denied.

Action: Return to the original database and modify the configuration file to turn the copy protection off, then create a new backup which may be used to restore the database. This must be done by an individual with SYSADM authority. If the original database is no longer available, contact your service representative.

SQL5123N

Database "<name>" cannot be configured because an I/O error occurred while accessing the log control file.

Cause: An error occurred while accessing SQLOGCTL.LFH for the specified database.

The requested change is not made.

Action: Restore the database from a backup copy or recreate the database.

SQL5126N

An attempt was made to modify the database manager configuration parameter "<parm>" which is not valid for a node type of "<node-type-code>".

Cause: The user attempted to modify a database manager configuration parameter which is invalid for the given node type. "<node-type-code">" is defined as follows:

0   Standalone (valid on DB2 for OS/2 only)
1   Database server with local and remote clients
2   Client
3   Database server with local clients

The requested change is not made.

Action: Resubmit the request with a valid parameter for the given nodetype.

SQL5130N

The value specified for the configuration parameter "<parameter>" is not in the valid range of "<start-of-range>" to "<end-of-range>".

Cause: The request is not done because the value of "<parameter>" is not within the valid range.

Action: Ensure that the value specified for "<parameter>" is within the valid range, then try the request again.

SQL5131N

The value specified for the configuration parameter "<parameter>" is not in the valid range. The valid range is "-1", or from "<start-of-range>" to "<end-of-range>".

Cause: The request is not done because the value of "<parameter>" is not within the valid range.

Action: Ensure that the value specified for "<parameter>" is within the valid range, then try the request again.

SQL5132N

The configuration parameter is null or too long. The maximum length is <maximum length>.

Cause: The configuration parameter is not set or too long.

Action: Change the value of the configuration parameter to within the maximum length given.

SQL5133N

The configuration parameter is invalid. The valid set of values is "<value 1>" or "<value 2>".

Cause: The configuration parameter is not one of the two values given.

Action: Change the value of the configuration parameter to one of the two values given.

SQL5134N

The configuration parameter tpname contains invalid characters.

Cause: One or more characters in the tpname is not in the valid range. The characters in the tpname must be one of the following:

- A - Z
- a - z
- 0 - 9
- $
- #
- @
- . (period)

Action: Change the tpname and retry the command or function call.

SQL5135N

The settings of the maxlocks and maxappls configuration parameters do not use all of the locklist space.

Cause: The number of active processes (maxappls) times the maximum percentage of lock list space for each application (maxlocks) must be greater than or equal to 100. That is:

```
maxappls * maxlocks >= 100
```

This ensures that all of the allocated locklist space can be used.

SQL5136N SQL5144N

Action: Increase the setting for maxappls, maxlocks, or both.

SQL5136N

The entry in the database manager configuration file for the default database path (dftdbpath) is invalid.

Cause: An invalid value was specified by dftdbpath. The rules for the default database path on UNIX-based systems are:

1. the path must follow the operating system's naming conventions
2. the path must exist
3. the path must be less than or equal to 215 characters

The rules for other platforms (such as OS/2, WindowsNT, Windows 95) are:

1. the path must be a drive letter
2. the drive must exist

Action: Change the dftdbpath and retry the command or function call.

SQL5137N

The entry in the database manager configuration file for the diagnostic directory path (diagpath) is invalid.

Cause: An invalid value was specified for *diagpath*. The rules for the diagnostic directory path are:

1. the path must follow the operating system's naming conventions

2. the path must exist

3. the path must be less than or equal to 215 characters

Action: Change the *diagpath* and retry the command or function call.

SQL5140N

The entry in the database manager configuration parameter authentication must be one of SERVER, CLIENT, or DCS.

Cause: The allowed values for the configuration parameter *authentication* are SERVER, CLIENT, or DCS.

The requested change is not made.

Action: Resubmit the command with a valid value for authentication.

SQL5141N

The configuration parameter avg_appls is out of range. The valid range is from 1 to the value of maxappls.

Cause: The allowable range for *avg_appls* is from 1 to the value of *maxappls*.

The requested change is not made.

Action: One or more of the following:

- Change the value of *avg_appls* to a value in the valid range.
- Increase the value of *maxappls* to a higher value, then set *avg_appls* again.

SQL5142N

The configuration parameter agentpri is not in the valid range.

Cause: The valid values for *agentpri* are -1, or a three digit number of which the first digit is a priority class ranging from 2 to 4 and the last two digits are a priority level within the class in the range of 00 to 31. The priority class is defined as follows:

2   REGULAR
3   TIMECRITICAL
4   FOREGROUNDSERVER

For example, the number 304 would correspond to a priority class of 3 (TIMECRITICAL), with a priority level of 4.

Action: Change the value of the configuration parameter to a value within the valid range.

SQL5144N

The update cannot be completed because the configuration parameter "<parameter>" would be out of range. The valid range is from "<start-of-range>" to the value of "<expression>".

Cause: The request is not completed because the value of "<parameter>" would not be within the valid range. The update either attempted to set the value of "<parameter>" outside its valid range or attempted to reduce the value of "<expression>".

Action: One or more of the following as required:

- Ensure that the value of "<parameter>" is within the valid range.

- Increase the value of "<expression>".

SQL5145N

The update cannot be completed because the configuration parameter "<parameter>" would be out of range. The valid range is "-1", or from "<start-of-range>" to the value of "<expression>".

Cause: The request is not completed because the value of "<parameter>" would not be within the valid range. The update either attempted to set the value of "<parameter>" outside its valid range or attempted to reduce the value of "<expression>".

Action: One or more of the following as required:

- Ensure that the value of "<parameter>" is within the valid range.

- Increase the value of "<expression>".

SQL5146N

The update cannot be completed because the configuration parameter "<parameter>" would be out of range. The valid range is from the value of "<expression>" to "<end-of-range>".

Cause: The request is not completed because the value of "<parameter>" would not be within the valid range. The update either attempted to set the value of "<parameter>" outside its valid range or attempted to increase the value of "<expression>".

Action: One or more of the following as required:

- Ensure that the value of "<parameter>" is within the valid range.

- Decrease the value of "<expression>".

SQL5147N

The update cannot be completed because the configuration parameter "<parameter>" would be out of range. The valid range is "-1", or from the value of "<expression>" to "<end-of-range>".

Cause: The request is not completed because the value of "<parameter>" would not be within the valid range. The update either attempted to set the value of "<parameter>" outside its valid range or attempted to increase the value of "<expression>".

Action: One or more of the following as required:

- Ensure that the value of "<parameter>" is within the valid range.

- Decrease the value of "<expression>".

SQL6000N

DB2 Conversion of QMF Data.

Cause: This is the normal termination message.

Action: No action is required.

SQL6001N

The prefix for the filename was not specified.

Cause: The user executed a module of the SQLQMF facility directly rather than using the SQLQMF facility command.

The command cannot be processed.

Action: Use the provided SQLQMF facility command.

SQL6002N

Communications Manager encountered a download error.

Cause: Communications Manager encountered an error during the download of the host file.

The command cannot be processed.

Action: Review the Communications Manager message log.

SQL6003N

The file exported from QMF contains rows that are too long. The row length is "<number>".

Cause: The computed row size (the sum of the computed column sizes) exceeds the maximum of 7000 bytes.

The command cannot be processed.

Action: Return to your QMF host session and run a query with fewer data columns selected. EXPORT the data again and then try the SQLQMF facility command again.

SQL6004N SQL6009N

---SQL6004N----------------

**"<function>" returned the unexpected return
code "<code>".**

Cause: An unexpected error occurred during
processing. The Communications Manager or DB2 may
not be installed or configured correctly.

The command cannot be processed.

Action: Ensure that Communications Manager has
been installed and the appropriate host communications
session is active. Retry the command after checking for
other errors. If the problem persists, contact your
Communications Manager system administrator.

---SQL6005N----------------

**An error occurred while reading the downloaded
QMF file.**

Cause: One of the following conditions was
encountered:

- The file could not be opened.
- The end of file was encountered prematurely.
- There was an I/O error while reading the file.

The command cannot be processed.

Action: Review the Communications Manager
message log. Retry the command. If the error
continues, contact your Communications Manager
system administrator.

---SQL6006N----------------

An error occurred while writing to an output file.

Cause: One of the following conditions was
encountered:

- Not enough space exists on the C: drive to write the
  data.
- An output file could not be opened.
- An I/O error occurred while writing the file.
- An I/O error occurred while closing the file.
- The file is in use by another OS/2 process.

DataJoiner users: there isn't enough space to hold the
djxlink output. DataJoiner is installed in the DataJoiner
directory Additional space is required for the temporary
files that djxlink creates and uses.

The command cannot be processed.

Action: Check for sufficient disk space on the C: drive.
Retry the command.

DataJoiner users: increase the size of the file system
where DataJoiner is installed. To estimate the amount of
free space djxlink needs on an AIX operating system,
enter:

```
ls -e /usr/lpp/djx*/lib/libdb2euni.a
```

This command returns the number of bytes used by the
listed file. Multiply that number by 3. The result is an
estimate of how much free space is needed on the file
system. Increase the size of the file system as
necessary and retry the command.

---SQL6007N----------------

**The decimal number in row "<row>", column
"<column>" cannot be converted to ASCII.**

Cause: The indicated decimal field could not be
converted.

The command cannot be processed.

Action: Verify that the host column data type is
DECIMAL. Retry the QMF EXPORT and then try the
SQLQMF facility command again. If the error continues,
rerun the QMF query without the named column.

---SQL6008N----------------

**The file specified in the command is not in QMF
data format.**

Cause: The file specified by the *filename* parameter is
not in the expected QMF format.

The command cannot be processed.

Action: Verify that you have typed the name correctly.
If correct, return to your QMF host session and issue the
command EXPORT DATA TO *filename* again. Be sure
to export using the QMF data format.

---SQL6009N----------------

**The file exported from QMF contains Column
"<name>" with a width "<width>" that is too
long. The maximum column width is 4000 bytes.**

Cause: The downloaded QMF file has a column width
greater than 4000 bytes.

The command cannot be processed.

Action: Return to your QMF host session, rerun the
QMF query without the named column, and export the
data again. Then run the SQLQMF facility command
again.

SQL6010N

The downloaded QMF file has more than 255 data columns.

Cause: The file being processed has more than 255 data columns.

The command cannot be processed.

Action: Return to your QMF host session and rerun a query, selecting 255 or fewer data columns. EXPORT the data again and then run the SQLQMF facility command again.

SQL6011N

The data type "<number>" ("<type-text>") of column "<name>" (positioned at column "<number>") cannot be processed.

Cause: The QMF file contains a column with data type that is not supported.

The SQLQMF facility does NOT support the following data types:

- LONG VARCHAR
- LONG VARGRAPHIC.

Only the SQLQMF facility SQLQMFDB supports graphic data types.

The command cannot be processed.

Action: Return to your QMF host session and rerun the query without selecting the named column. Then run the SQLQMF facility command again.

SQL6012N

Too many parameters were specified for the command.

Cause: Too many parameters were specified with the command.

The command cannot be processed.

Action: Retry the command with the correct number of parameters.

SQL6013N

Host filename "<host filename>" is either too long or does not begin with an alphabetic character.

Cause: The *host filename* does not begin with an alphabetic character; or, if the host is a VM System, the *host filename*, *filetype*, or *filemode* are too long.

The command cannot be processed.

Action: Retry the command with a correct *host filename* syntax.

SQL6014N

Invalid command syntax, colon (':') should follow a keyword.

Cause: Keyword parameters with operands require the keyword to be immediately followed by a ":" character and then the operand.

The command cannot be processed.

Action: Verify the command syntax and retry the command.

SQL6015N

The keyword is not recognized.

Cause: The keyword parameter indicator ("/") was followed by a value that is not a keyword.

The command cannot be processed.

Action: Retry the command using a different keyword value.

SQL6016N

Too many operands for the System/370 filename "<name>".

Cause: If the host is a VM System, the host file name contains more than three space-separate tokens. If the host is an MVS System, the host filename contains embedded blanks.

The command cannot be processed.

Action: Retry the command using a different keyword value.

SQL6017N SQL7004N

SQL6017N

More information may be contained in the Import message log "<name>".

Cause: The database IMPORT operation terminated with warning or error messages.

The command has left its work files allocated.

Action: Use the message preceding this message and the IMPORT message log, if it exists, to decide if the IMPORT was successful and to determine corrective action. If the import was successful, erase the DEL, CRE, COL, and IML files.

SQL6018N

No S/370 filename was specified.

Cause: The S/370 filename is a required parameter.

The command cannot be processed.

Action: Retry the command with a host filename.

SQL6019N

Communications short session ID "<ID>" is either too long or is invalid.

Cause: The value provided for the communications short session ID is either longer than one byte or is not alphabetic.

The command cannot be processed.

Action: Retry the command with a valid value.

SQL6020N

An import option was specified without also providing a database name.

Cause: No database name was provided and an import option was specified.

The command has terminated.

Action: Retry the command and include a database name.

SQL6021N

Data was successfully imported.

Cause: This is a normal ending message when the SQLQMF facility imports data to a database.

Action: No action is required.

SQL7001N

Unknown command "<command>" was requested.

Cause: The command submitted to REXX could not be recognized.

The command cannot be processed.

Action: Verify that the command is a valid SQL statement and rerun the procedure. Note that all commands must be in uppercase.

SQL7002N

The cursor name is invalid.

Cause: The user specified an incorrect cursor name.

The command cannot be processed.

Action: Verify that the cursor name is one of the form "c1" through "c100".

SQL7003N

The statement name is invalid.

Cause: The user has specified an incorrect statement name.

The command cannot be processed.

Action: Verify that the statement name is of the form "s1" through "s100".

SQL7004N

The syntax of the request is invalid.

Cause: REXX was unable to parse the command string submitted.

The command cannot be processed.

Action: Use proper command syntax.

SQL7005W

The cursor used by this OPEN statement has not been declared.

Cause: An OPEN statement was attempted without a declaring a cursor.

The command cannot be processed.

Action: Insert a DECLARE statement before the OPEN statement and rerun the procedure.

SQL7006N

Invalid keyword "<keyword>" given on "<request>".

Cause: The statement contains a not valid keyword "<keyword>".

The command cannot be processed.

Action: Specify the correct keyword format.

SQL7007N

REXX variable "<variable>" does not exist.

Cause: A REXX variable that did not exist in the REXX variable pool was passed.

The command cannot be processed.

Action: Verify that all variable names in the host-variable list are assigned before the command that failed. Then run the procedure again.

SQL7008N

REXX variable "<variable>" contains inconsistent data.

Cause: A variable that contained inconsistent data was passed to REXX.

The command cannot be processed.

Action: If the variable is an SQLDA, verify that the data and length fields have been assigned correctly. If it is a REXX variable, verify that the type of data is appropriate to the command where it is being used.

SQL7009N

Truncated REXX variable "<variable>".

Cause: The variable "<variable>" passed to REXX contained inconsistent data. The data string from "<variable>" was truncated.

The command cannot be processed.

Action: Verify that the data length is consistent with the length given in the input SQLDA, and run the procedure again.

SQL7010N

Invalid scan ID "<ID>".

Cause: The scan ID "<variable>" passed to REXX did not exist or contained inconsistent or missing data.

The command cannot be processed.

Action: Verify that the data contained in the scan ID has been assigned correctly, and run the procedure again.

SQL7011N

Required parameter "<parameter>" not given.

Cause: The parameter "<parameter>" is required by REXX command syntax but was not specified.

The command cannot be processed.

Action: Specify the required parameter value, and run the procedure again.

SQL7012N

Attempted ISL change while connected to a database.

Cause: Isolation level (ISL) may not be changed while connected to a database.

The command is ignored.

Action: If a change in isolation level is necessary, disconnect from the current database, then set the isolation level and connect to the database.

SQL7013N SQL8001N

SQL7013N

Mismatched cursor and statement name or hold attribute.

Cause: In REXX, the cursor and statement name form must be 'cnn' and 'snn,' where 'nn' is a number from 1 to 100. The numbers must be the same for a cursor and statement pair. Also, c1 through c50 are declared without hold and c51 through c100 with hold.

The command is ignored.

Action: Verify that the cursor and statement numbers match and rerun the procedure.

SQL7014N

Incorrect number of host variable components.

Cause: In REXX, the first component of a compound host variable lists a number which is not equal to the actual number of components defined.

The command is ignored.

Action: Verify that the number in the first component matches the number of components actually defined and rerun the procedure.

SQL7015N

The variable name "<variable>" is not valid in REXX.

Cause: The variable name specified is not valid in REXX. The name must conform to the language requirements.

The command cannot be processed.

Action: Change the name to one that conforms to the REXX requirements and resubmit the command.

SQL7099N

Invalid error "<error>" has occurred.

Cause: A REXX internal error has occurred.

The command cannot be processed.

Action: Verify that REXX is installed properly and rerun the procedure. If the error continues, note the error number and call the dealer.

SQL8000N

A DB2 server license was not found.

Cause: This message is displayed at db2start time, when the customer node does not have a DB2 server license.

Action: Check if the current node has a DB2 NetLS nodelocked license with the correct version. If a license does not exist, then contact an IBM representative to obtain the license, and follow the steps required to install the license as explained in the *Installation and Operation Guide*.

SQL8001N

A DB2 concurrent connect license was not found. Reason code "nnnn".

Action: Check if the current node has a DB2 NetLS nodelocked license with the correct version.

If a license does not exist, then contact an IBM representative to obtain the license, and follow the steps required to install the license as explained in the *Installation and Operation Guide*.

If a license does exist, then you might have reached the limit as to the number of allowed concurrent database client connections to the DB2 server. The following is a list of relevant reason codes:

```
-100   Encountered an unknown error
-101   Internal DB2 license parameter error
-102   Format of vendor ID in the license
       information was not correct
-103   NetLS license server error
-105   NetLS calls were not initialized
-106   The NetLS product encountered
       self-license error
-107   Unsupported machine model
-108   License information is for an unknown
       machine
-109   License is for a lower machine model
-110   License did not exist
-111   License has not started or is expired
-112   License has wrong product version text
-114   No more licenses available
-115   The user is not authorized by the NetLS
       user file to use the product
-205   No more concurrent licenses available
```

Note: Save the reason code for future reference by an IBM representative.

sqlcode: -8001

sqlstate: 42968

SQL8002N

A DDCS concurrent Host database connect license was not found. Reason code "nnnn".

Cause: This message is returned when attempting to connect to a host database, and no more concurrent DDCS client licenses are available.

Action: Check if the current node has a DDCS NetLS nodelocked license with the correct version.

If a license does not exist, then contact an IBM representative to obtain the license, and follow the steps required to install the license as explained in the *Installation and Operation Guide.*

If a license does exist, then you might have reached the limit as to the number of allowed concurrent host database client connections.

The following is a list of relevant reason codes:

```
-100    Encountered an unknown error
-101    Internal DDCS license parameter error
-102    Format of vendor ID in the license
        information was not correct
-103    NetLS license server error
-105    NetLS calls were not initialized
-106    The NetLS product encountered
        self-license error
-107    Unsupported machine model
-108    License information is for an unknown
        machine
-109    License is for a lower machine model
-110    License did not exist
-111    License has not started or is expired
-112    License has wrong product version text
-114    No more licenses available
-115    The user is not authorized by the NetLS
        user file to use the product
-205    No more concurrent licenses available
```

Note: Save the reason code for future reference by an IBM representative.

sqlcode: -8002

sqlstate: 42968

SQL8005N

A DB2 SDK license was not found.

Cause: This message is displayed when the customer node does not have a DB2 SDK nor a DB2 license.

Action: Check if the current node has a DB2 or a DB2 SDK NetLS nodelocked license with the correct version. If a license does not exist, then contact an IBM representative to obtain the license, and follow the steps required to install the license as explained in the *Installation and Operation Guide.* If a license does exist, make sure that all the license information has been entered correctly, and check the license expiry date.

sqlcode: -8005

sqlstate: 42968

SQL8006W

This product does not have a valid license key installed. If you have licensed this product, please ensure the license key is properly installed.

If a license key is not installed, this product will be enabled for a "<number>" day evaluation period. USE OF THE PRODUCT FOR THE EVALUATION PERIOD CONSTITUTES ACCEPTANCE OF THE TERMS OF THE IBM EVALUATION AGREEMENT IN THE EVALUATE.AGR FILE, LOCATED IN THE PRODUCT INSTALL PATH.

Cause: No license key for the product has been installed. The product will be enabled to operate for a limited period of time for evaluation purposes. THE EVALUATION PERIOD IS CONTROLLED BY A TIME DISABLING DEVICE.

Action: If you have purchased a fully entitled version of this product, please install your license key as described in the product installation documentation. If you have already installed the license key, please verify your license file for accuracy.

USE OF THE PRODUCT FOR THE EVALUATION PERIOD IS SUBJECT TO THE TERMS OF THE IBM EVALUATION AGREEMENT IN THE EVALUATE.AGR FILE, LOCATED IN THE PRODUCT INSTALL PATH. YOUR USE OF THE PRODUCT DURING THE EVALUATION PERIOD CONSTITUTES YOUR ACCEPTANCE OF THESE TERMS. If you do not accept the terms, you are not authorized to use the product and you must de-install it. You can obtain a license key for the fully authorized program with the

product by contacting your IBM representative or authorized dealer.

SQL8007W

There are "<number>" day(s) left in the product evaluation period. Refer to the IBM Evaluation Agreement in the EVALUATE.AGR file, located in the product install path, for evaluation license terms and conditions.

Cause: A permanent license key has not been entered for this product. The evaluation period will expire after the specified number of days.

Action: The product is operating in evaluation mode, and is enabled for a limited period of time. When the evaluation period has expired it will not execute until a license key for the fully entitled version of the product is installed. You can obtain a license key for the product by contacting your IBM representative or authorized dealer.

SQL8100N

The table is full.

Cause: When creating a database the following parameters may be specified:

- maximum number of pages per segment per file

- number of segments

Each table part may now have multiple files within the database segments. The file can grow until it reaches its maximum size in a segment (maximum number of pages per segment) and then it moves to the next segment to add more data. This occurs until it has used the maximum number of pages per segment in all configured segments, and the table becomes full.

Therefore, the total amount of space per database part is the product of the maximum number of pages and the number of segments. The table will be full if any of the table parts use all of their configured space.

DataJoiner users: this situation can be detected by DataJoiner or by the data source.

Action:

- Delete rows from the table

- Reorganize the table to make more space

- Create a new database with larger values for maximum pages per segment and number of segments, back up the original database and restore to the new database.

DataJoiner users: isolate the problem to the data source failing the request (refer to the *Problem Determination Guide* to determine which data source is failing to process the SQL statement) and:

- Take the necessary steps to increase the space in the table at the data source.

- If the table is on DataJoiner, create a new database with larger values for maximum pages per segment and number of segments. Back up the original database and restore to the new database.

SQL8101N

A database segment may be incorrect.

Cause: This error could occur in two ways:

1. Every database segment has an identifier file. The file may be missing or the contents of the file may be incorrect.

2. One, or more, previously allocated database segments are missing.

Action:

- Check to ensure that the file systems are mounted correctly

- Restore the database from a backup

- Call your IBM service representative

SQL10002N

The password supplied is too long.

Cause: The length of the password is expected to be 18 characters or less. However, if the password is to be verified for an APPC conversation, it must be 8 characters or less.

Action: Ensure that the password is not longer than the limit allowed.

sqlcode: -10002

sqlstate: 28000

SQL10003C

There are not enough system resources to process the request.

The request cannot be processed.

Cause: The database manager could not process the request due to insufficient system resources. The resources that can cause this error include:

- The amount of memory in the system

- The number of message queue identifiers available in the system.

Action: Stop the application. Possible solutions include:

- Remove background processes.

- Terminate other applications using the above resources.

- If you are using Remote Data Services, increase the Remote Data Services heap size (rsheapsz) in the server and client configuration because at least one block is used per application.

  NOTE: This is applicable only for releases of DB2 prior to Version 2.

- Decrease the values of the configuration parameters that define allocation of memory, including *udf_mem_sz* if UDFs are involved in the failing statement.

sqlcode: -10003

sqlstate: 57011

SQL10004C

An I/O error occurred while accessing the database directory.

Cause: Either the system database directory or the local database directory cannot be accessed. This error may occur not only when the system is cataloging or uncataloging a database but also when the system is accessing a database that is cataloged in the directory.

The command cannot be processed.

Action: Possible solutions include:

- If the local database directory is damaged, restore the databases that were cataloged there from backup versions and catalog them.

If installing the sample database, drop it and install the sample database again.

sqlcode: -10004

sqlstate: 58031

SQL10005N

The mode parameter "<parameter>" in the CONNECT TO statement is not valid. It must be SHARE for shared access or EXCLUSIVE for exclusive use. For DDCS connections, only SHARE mode is supported.

Cause: The *mode* parameter of the CONNECT TO statement must be either SHARE for shared or EXCLUSIVE for exclusive use. If connecting to a database using Distributed Database Connection Services (DDCS), only shared access is allowed.

The command cannot be processed.

Action: Resubmit the command with a valid *mode* parameter.

SQL10007N

Message "<msgno>" could not be retrieved. Reason code: "<code>".

Cause: The requested message <msgno> could not be retrieved from the message file. Reason code <code> is one of the following:

1. The environment variable "DB2INSTANCE" is either not set, or is set to an invalid instance. Correct it and try again.
2. The message file was found, but the file could not be opened because of permissions. Check the file permissions of the files under the message directory.
3. The message file could not be found. Either the file does not exist, or the directory the message file should be in does not exist. Check that a either a 'prime' directory (the default) or a directory with the same name as the 'LANG' environment variable exists under the message directory.
4. The requested message does not exist in the message file. Either the message file is outdated, or it is the wrong one.
5. LC_CTYPE is set to a locale which the database does not support. Correct it and try again.
6. An unexpected system error occurred. Try execution again. If problem persists, contact your IBM representative.
7. Not enough memory. An attempt to get private memory failed. Try again.

Action: Reissue the command after verifying the following:

- ensure that the DB2INSTANCE environment variable is set to the correct literal string for the username attempting this command
- ensure that correct home directory is specified for the username attempting this command (i.e. in the /etc/passwd file)
- ensure that the LANG environment variable is set to the correct value for the installed language, or is set to 'C' (defaults to whatever is in the 'prime' directory), on the username attempting this command

If all of the above are correct and the error still occurs, reinstall DB2.

SQL10009N

The codeset "<codeset>" and/or territory "<territory>" that has been specified is not valid.

Cause: This version of the database manager does not support the active codeset or territory or both, as specified by the Create Database command.

The command cannot be processed.

Action: Refer to the Create Database command in the *Administration Guide* for details of valid codeset and territory pairs supported by the database manager.

SQL10010N

The specified library, "<name>", was loaded, but the function "<function>" could not be executed.

Cause: The function routine within the library cannot be found.

The command cannot be processed.

Action:

1. If it is a DARI library, ensure the DARI library is built correctly. Make sure that the function routine is exported through the 'export file'.

2. For other libraries, user may have to reinstall the database manager instance or the database manager product.

sqlcode: -10010

sqlstate: 42724

SQL10011N

An interrupt was received while loading the specified DARI library "<name>".

Cause: When loading the DARI library, the command was interrupted, perhaps by the interrupt key (usually Ctrl+Break or Ctrl+C).

Processing is stopped.

Action: Resubmit the command.

sqlcode: -10011

sqlstate: 42724

SQL10012N

An unexpected operating system error was received while loading the specified library "<name>".

Cause: An unexpected error occurred when trying to load the library module specified in the program name field.

Action: Resubmit the current command. If the error continues, stop the database manager and start it again. If the error continues, reinstall the database manager.

If reinstallation does not correct the error, record the message number (SQLCODE) and all information in the SQLCA if possible.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Then contact IBM as specified in this guide.

sqlcode: -10012

sqlstate: 42724

SQL10013N

The specified library "<name>" could not be loaded.

Cause: The library module could not be found.

DataJoiner users: the specified library module could be one of the following:

- a DataJoiner system library
- a DARI library
- a DataJoiner data access module

Action: For DARI users, ensure that the specified library is available.

- If the client application uses a fully qualified path name to specify the DARI library, the DARI library must be stored in the given directory path. If the client application does not use the path name, the DARI library must be stored in the default directory (<InstanceHomeDir>/sqllib/function). Where <InstanceHomeDir> is the home directory of the database manager instance.

- If the error message occurs during database manager start time then the user will have to reinstall the DB2 instance or the database manager product.

DataJoiner users: if the error message occurs as a result of issuing db2start on the DataJoiner system and the "<name>" is "from DJXCOMM" then DataJoiner experienced a problem loading one of the data access modules identified by the DJXCOMM environment variable. The DJXCOMM environment variable must be updated to contain only valid data access modules.

If the error message occurs as a result of issuing db2start on the DataJoiner system and the "<name>" is not "from DJXCOMM", the user must re-install the DataJoiner instance or the DataJoiner product.

If the library identifies a data access module, that module must be installed, link-edited (as required), and
tcctoLe¢è¢´èàthe i$nmuing tur pr'u–em <na°e "<namet)è¢èàuretified   (as not val.du: s.ãèÃ.èThe tur library $module or the tur pr'u–ae¢¢èà<na°e sont¨d (as
mnd prccoduŸs.tcctCLe¢è¢èT–m ujt as da. prcceuui¾n $u: s.ãèÃ.èT–m ujt as da.T–m   Øe e¢¢$fluc,m T»°e The t as ns u,s °e Tieèè/t tiis T–m
n°e v"rŸs.ãèÃ.

SQL10019N

The database cannot be accessed in the specified path.

Cause: The database cannot be accessed for one of the following reasons:

- The path does not contain a database image.

- Permissions on the path are incorrect.

Action: Ensure that the path references a valid database and that the permissions are correct.

sqlcode: -10019

sqlstate: 58031

SQL10021N

The file permissions do not allow write access to the database.

Cause: A write operation was attempted to a database residing on a file system which does not allow write access.

The statement cannot be processed.

Action: Verify that the file permissions for the file system on which the database resides allow write access.

SQL30000N

Execution failed because of a Distributed Protocol Error that will not affect the successful execution of subsequent commands or SQL statements: Reason Code "<reason-code(subcode)>".

Cause: A system error occurred that prevented successful processing of the current environment command or SQL statement. This message (SQLCODE) is produced at statement compilation or execution time.

The command or statement cannot be processed. The current transaction is not rolled back and the application remains connected to the remote database.

Action: Record the message number and the reason code. Record all error information from the SQLCA, if possible. Attempt to rerun the application.

If sufficient memory resources exist and the problem continues, invoke the Independent Trace Facility at the operating system command prompt. Refer to the

Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

Required information:

- Problem description
- SQLCODE and reason code
- SQLCA contents if possible
- Trace file if possible.

sqlcode: -30000

sqlstate: 58008

SQL30020N

Execution failed because of a Distributed Protocol Error that will affect the successful execution of subsequent commands and SQL statements: Reason Code "<reason-code>".

Cause: A system error occurred that prevented successful execution of the current environment command or SQL statement, as well as any subsequent commands or SQL statements.

The command or statement cannot be processed. The current transaction is rolled back and the application is disconnected from the remote database.

Action: Record the message number (SQLCODE) and the reason code. Record all error information from the SQLCA, if possible. Attempt to connect the remote database and rerun the application.

If sufficient memory resources exist and the problem continues, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

Required information:

- Problem description
- SQLCODE and reason code
- SQLCA contents if possible
- Trace file if possible.

Some possible reason codes include:

121C Indicates that the user is not authorized to perform the requested command.
1232 The command could not be completed because of a permanent error. In most cases, the server will be in the process of an abend.
220A The target server has received an invalid data description. If a user SQLDA is specified, ensure that the fields are initialized correctly. Also, ensure that the length does not exceed the maximum allowed length for the data type being used.

If you are using the DDCS product in a gateway server environment with downlevel clients, this error may occur if your application's host variables and the queried table's column descriptions do not match.

sqlcode: -30020

sqlstate: 58009

SQL30021N

Execution failed because of a Distributed Protocol Error that will affect the successful execution of subsequent commands and SQL statements: Manager "<manager>" at Level "<level>" not supported.

Cause: A system error occurred that prevented successful connection of the application to the remote database. This message (SQLCODE) is produced for SQL CONNECT statement. "<manager>" and "<level>" are numeric values that identify the incompatibility between the client and the server.

The command cannot be processed.

Action: Record the message number, the "<manager>", and "<level>" values. Record all error information from the SQLCA, if possible. Attempt to connect to the remote database again.

If the problem persists, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Then, contact your service representative with the following information:

- Problem description
- SQLCODE and reason code
- SQLCA contents if possible
- Trace file if possible.

sqlcode: -30021

sqlstate: 58010

SQL30040N

Execution failed because of unavailable resources that will not affect the successful execution of subsequent commands and SQL statements: Reason "<reason>". Type of Resource "<resource-type>". Resource name "<resource-name>". Product ID "<product-ID>".

Cause: The application cannot process the command or SQL statement because the specified resource has

run out. The current transaction is not rolled back and the application remains connected to the remote database.

The command cannot be processed.

Action: Increase the size of the specified resource and resubmit the command.

sqlcode: -30040

sqlstate: 57012

SQL30041N

Execution failed because of unavailable resources that will affect the successful execution of subsequent commands and SQL statements: Reason "<reason>". Type of Resource "<resource-type>". Resource name "<resource-name>". Product ID "<product-ID>".

Cause: The application cannot process the command or SQL statement because resources are not available at the remote database.

DataJoiner users: this situation can be detected by DataJoiner or by the data source. Possible causes are:

- If the "<reason>" is "stubOnly", "<resouce type>" identifies a data source type where the client libraries have been either not link-edited or incorrectly link-edited with DataJoiner. "<resource name>" identifies the data source function DataJoiner attempted to invoke but was unable to find.
- If the "<resource name>" is "Encrypt" or "Decrypt", the user exits for password encryption or decryption are unavailable or had an error.

This SQLCODE is produced at statement compilation or processing time.

The command or statement cannot be processed.

Action: Investigate the remote database system environment.

DataJoiner users:

- If the "<reason>" is "stubOnly", the data source client libraries must be link-edited with DataJoiner using djxlink as described in the *IBM DataJoiner Planning, Installation, and Configuration Guide*.
- If the password encryption or decryption user exit failed, "<reason>" is the integer value that DataJoiner received when invoking the user exit. Ensure that no errors occurred when the encryption and decryption user exits were link-edited with DataJoiner.

SQL30050N • SQL30052N

If using a user-supplied user exit, please check the user exit source code to determine why the "<reason>" was returned. If the error is found in the user exit source code, fix the error, link-edit the object code with DataJoiner, and re-issue the failing command or statement.

- For all other cases, isolate the problem to the data source failing the request (refer to the *Problem Determination Guide* to determine which data source is failing in processing an SQL statement), correct the problem at the data source, and re-issue the failing command or statement.

sqlcode: -30041

sqlstate: 57013

SQL30050N

"<number>" command or SQL statement invalid while bind is in progress.

Cause: An application attempted to issue a command or SQL statement that is not valid while a precompile/bind operation is in progress. <number> is a numeric value that identifies the erroneous command or SQL statement.

The command or statement is not processed.

Action: If the application is not the database manager precompiler/binder, verify that bind is not active before issuing the command or SQL statement.

If the application is the database manager precompiler/binder, record the message number (SQLCODE) and the <number> value. Record all error information from the SQLCA, if possible. Attempt to perform the bind operation again.

If sufficient memory resources exist and the problem continues, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE and reason code
- SQLCA contents if possible
- Trace file if possible.

sqlcode: -30050

sqlstate: 58011

SQL30051N

Bind process with the specified package name and consistency token not active.

Cause: An attempt has been made to issue a precompile/bind operation when precompile/bind is not active or an attempt was made to use an invalid package name and/or consistency token during an active precompile/bind operation.

The command or statement cannot be processed.

Action: If the application is not the database manager. precompiler/binder, verify that precompile/bind is active before issuing the bind operation and that correct information is being passed on the bind operation.

If the application is the database manager precompiler/binder, record the message number (SQLCODE) and all error information from the SQLCA, if possible. Attempt to perform the operation again.

If sufficient memory resources exist and the problem continues, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE and reason code
- SQLCA contents if possible
- Trace file if possible.

sqlcode: -30051

sqlstate: 58012

SQL30052N

Program preparation assumptions are incorrect.

Cause: The SQL statement being compiled is not recognized by the precompiler and cannot be processed by the database.

The statement cannot be processed.

Action: Ensure that the statement is correct and attempt to perform the operation again. If the problem continues, remove the statement from the program.

sqlcode: -30052

sqlstate: 42932

SQL30053N

The value for OWNER failed the authorization check at the remote database.

Cause: The value specified for the OWNER option on the precompile/bind failed the authorization check at the remote database. This SQLCODE is produced during precompile/bind. It is not produced by the database manager precompiler/binder.

The precompile/bind operation cannot be processed.

Action: Ensure that you are authorized to use the ID specified for the OWNER option, or do not use the OWNER option.

sqlcode: -30053

sqlstate: 42506

SQL30060N

"<authorization-ID>" does not have the privilege to perform operation "<operation>".

Cause: Authorization ID <authorization-ID> attempted to perform the specified <operation> without having been granted the proper authorization to do so. This SQLCODE is produced at statement compilation or processing time.

The statement cannot be processed.

Action: Ensure that the <authorization-ID> has been granted the authorization necessary to perform the desired operation.

sqlcode: -30060

sqlstate: 08004

SQL30061N

The database alias or database name "<name>" was not found at the remote node.

Cause: The database name is not an existing database at the remote database node.

The statement cannot be processed.

Action: Resubmit the command with the correct database name or alias.

DataJoiner users: check to ensure that the entry in SYSCAT.SERVERS correctly specifies the database name of the data source.

sqlcode: -30061

sqlstate: 08004

SQL30070N

"<command-identifier>" Command is not supported.

Cause: The remote database received a command that it does not recognize. The current environment command or SQL statement cannot be processed successfully, nor can any subsequent commands or SQL statements.

The current transaction is rolled back and the application is disconnected from the remote database. The statement cannot be processed.

Action: Record the message number (SQLCODE) and the command identifier. Record all error information from the SQLCA, if possible. Attempt to connect to the remote database and rerun the application.

If sufficient memory resources exist and the problem continues, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE and command identifier
- SQLCA contents if possible
- Trace file if possible.

sqlcode: -30070

sqlstate: 58014

SQL30071N

"<object-identifier>" Object is not supported.

Cause: The remote database received data it does not recognize. The current environment command or SQL statement cannot be processed successfully, nor can any subsequent commands or SQL statements.

The current transaction is rolled back and the application is disconnected from the remote database. The command cannot be processed.

Action: Record the message number (SQLCODE) and the object identifier. Record all error information from the SQLCA, if possible. Attempt to connect to the remote database and rerun the application.

If sufficient memory resources exist and the problem continues,

SQL30072N   SQL30073N

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE and object identifier
- SQLCA contents if possible
- Trace file if possible.

sqlcode: -30071

sqlstate: 58015

SQL30072N

"<parameter-identifier>" Parameter is not supported.

Cause: The remote database received data it does not recognize. The current environment command or SQL statement cannot be processed successfully, nor can any subsequent commands or SQL statements.

The command cannot be processed.. The current transaction is rolled back and the application is disconnected from the remote database.

Action: Record the message number (SQLCODE) and the parameter identifier. Record all error information from the SQLCA, if possible. Attempt to connect to the remote database and rerun the application.

If sufficient memory resources exist and the problem continues, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE and parameter identifier
- SQLCA contents if possible
- Trace file if possible.

sqlcode: -30072

sqlstate: 58016

SQL30073N

"<parameter-identifier>" Parameter value "<value>" is not supported.

Cause: The remote database received data it does not recognize. The current environment command or SQL statement cannot be processed successfully, nor can any subsequent commands or SQL statements.

The current transaction is rolled back and the application is disconnected from the remote database. The command cannot be processed.

Action: Record the message number (SQLCODE) and the parameter identifier. Record all error information from the SQLCA, if possible. Attempt to connect to the remote database and rerun the application.

If sufficient memory resources exist and the problem continues,

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility. Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE, parameter identifier, and value
- SQLCA contents if possible
- Trace file if possible.

Some possible parameter identifiers include:

002F The target server does not support the data type requested by the application requester. For example, when DDCS is used to connect to DB2 2.3, this error will be returned unless the proper PTF is applied to DB2 2.3. Ensure that the level of the server is supported by the requester.

119C, 119D, 119E The target server does not support the CCSID requested by the application requester. Ensure the CCSID used by the requester is supported by the server.

- 119C - Verify the single-byte CCSID.
- 119D - Verify the double-byte CCSID.
- 119E - Verify the mixed-byte CCSID.

sqlcode: -30073

sqlstate: 58017

SQL30074N

"<reply-identifier>" Reply is not supported.

Cause: The client received a reply it does not recognize. The current environment command or SQL statement cannot be processed successfully, nor can any subsequent commands or SQL statements.

The current transaction is rolled back and the application is disconnected from the remote database. The statement cannot be processed.

Action: Record the message number (SQLCODE) and the reply identifier. Record all error information from the SQLCA, if possible. Attempt to connect to the remote database and rerun the application.

If sufficient memory resources exist and the problem continues, invoke the Independent Trace Facility at the operating system command prompt. Refer to the Independent Trace Facility in the *Problem Determination Guide* for information on how to use this facility.

Contact your technical service representative with the following information:

Required information:

- Problem description
- SQLCODE and reply identifier
- SQLCA contents if possible
- Trace file if possible.

sqlcode: -30074

sqlstate: 58018

SQL30080N

A communication error "<reason-code>" occurred sending or receiving data from the remote database.

Cause:

The "<reason-code>" contains the original error codes reported by Communications Manager.

For APPC error codes, the format is *pppp-ssssssss-dddddddd* where *pppp* is the primary return code, *ssssssss* is the secondary return code and *dddddddd* is the sense data. Each of these error code values is represented in hexadecimal notation. Possible primary and secondary error codes include 0003-00000004 and 0003-00000005, which indicate that a conversation with the Database Server could not be allocated. Sense data will only be displayed for an APPC ALLOCATE error.

The command cannot be processed. The connection attempted to the database is unsuccessful or the current transaction to the connected database is rolled back and the application is disconnected from the remote database.

Refer to the document *IBM Communications Manager 1.0 APPC Programming Guide and Reference (SC31-6160)* for explanation of the APPC primary and secondary return codes. For details of APPC sense data, refer to the *IBM Communications Manager 1.0 Problem Determination Reference (SC31-6157)*.

Action: Ensure that both the database manager and Communications Manager are started at the Database Server and that all Communications Manager configuration parameters are correct. Also, consult the *Communications Manager/2 APPC Programming Guide & Reference* for further information.

Note: If the reason code embedded in your message is 0003-084C0001 one possible cause for this message is that you are trying to make a remote connection to a workstation that is not a server. Verify that the remote workstation is a server. If it is not a server, install the database manager as a server.

sqlcode: -30080

sqlstate: 08001

SQL30081N

A communication error has been detected.

> **Communication protocol being used: <protocol>**
> **Communication API being used: <interface>**
> **Location where the error was detected: <location>**
> **Communication function detecting the error: <function>**
> **Protocol specific error code(s): <rc1>, <rc2>, <rc3>**

Cause: An error has been detected by the communication subsystem.

If the connection has already been established, possible causes include the following:

- A communication subsystem error at the client node or server node has caused the connection to go down.
- A communication subsystem error at the SOCKS server, if one is being used, has caused the connection to go down.

SQL30081N   SQL30081N

- A network error has caused the connection to go down.
- The database agent at the server was forced off by the system administrator.
- The database agent at the server was terminated due to an abnormal termination of a key database manager process.

If you are trying to establish a new connection, possible causes include the following:

- The remote database server has not been cataloged correctly at the client.
- The database manager configuration file at the server has not been configured with the proper communication parameters.
- The communication subsystem at the client node or server node has not been configured correctly, or has not been started successfully.
- The communication subsystem at the SOCKS server, if one is being used, has not been configured correctly, or has not been started successfully.
- The DB2COMM environment variable at the server doesn't specify the communication protocol used by the client.
- The database manager at the server has not been started, or has not been started successfully. One or more of the communication protocols specified by DB2COMM may not have been successfully started.

Refer to the token values for details. Depending on the protocol being used and the communication function invoked, some tokens may not be applicable. The following is the explanation of the token values:

<protocol> The actual communication protocol being used. Valid token values are:

- TCP/IP
- APPC
- NETBIOS
- IPX/SPX

<interface> The application programming interface used to invoke the above communication protocol services. Valid token values are:

- SOCKETS
- SOCKS
- CPI-C
- DLR or DD
- TLI or Connection/Bindery

<location> This is a protocol specific ID which uniquely identifies the node that detected the error. If the protocol being used is:

- TCP/IP

The format of the ID is the Internet address in dot notation.

- APPC

The format of the ID is the fully qualified LU name (networkID.LUname).

- NETBIOS

The format of the ID is the Workstation name (nname).

- IPX/SPX

The format of the ID is the network address and node address in hexadecimal notation (network.node).

If the location information is not available at the time that the error occurred, this token is not filled in.

<function> The name of the communication subsystem function that returned the error code(s).

<rc1>, <rc2>, <rc3> The list of the available error codes and subcodes that are specific to each protocol being used. Tokens that are not applicable contain "*".

If the protocol being used is:

- TCP/IP
  - If present, <rc1> contains the global *errno* value from the TCP/IP sockets function call. When using Windows Sockets, if present, <rc1> contains the error from the TCP/IP sockets function call returned by WSAGetLastError().
  - If present, <rc2> contains the global *h_errno* value from the TCP/IP database function call. When using Windows Sockets, if present, <rc2> contains the error from the TCP/IP database function call returned by WSAGetLastError().
  - If <rc3> is present and contains "0", it means that the TCP/IP connection has been closed. This could be caused by one of the following:
    - The database agent at the server was forced off by the system administrator.
    - A database agent could not be started at the server because the *maxagents* database manager configuration parameter has been exceeded. Check the First Failure Service Log (DB2DIAG.LOG) at the server to see if an error message has been logged.
    - The database agent at the server was terminated due to an abnormal termination of a key database manager process.

Note: When using Windows Sockets, if the <function> is WSAStartup, and <rc1> is 0, then <rc2> contains the Windows Sockets Specification version level requested by DB2, and <rc3> contains the Windows Sockets Specification version level supported by the Windows Sockets DLL.

For more information on TCP/IP communication errors, see "TCP/IP" on page 478.

- APPC

<rc1> contains the return code from the CPI-C function. If present, <rc2> contains the global *errno* value from the CPI-C function call. <rc3> is not applicable.

For more information on APPC communication errors, see "APPC" on page 481.

- NETBIOS

<rc1> contains the return code from the call to NetBIOS. <rc2> and <rc3> are not applicable.

For more information on NETBIOS communication errors, see "NETBIOS" on page 484.

- IPX/SPX

<rc1> contains the global *t_errno* value from the call to TLI Services, or the return code from the call to NetWare Connection or Bindery Services. If the <rc1> t_errno is TLOOK, <rc2> contains the TLI event that has occurred. If the <rc2> event is T_DISCONNECT, <rc3> contains the disconnect reason code. On AIX, if the <rc1> t_errno is TSYSERR, <rc3> contains the system errno (defined in sys/errno.h). If <rc1> contains the error return code from a call to AIX NetWare Connection or Bindery Services, <rc2> contains the error generator.

For more information on IPX/SPX communication errors, see "IPX/SPX" on page 485.

Action:

If the connection has already been established, check if:

1. The database agent at the server has been forced off.

2. The database manager at the server has abnormally terminated.

3. A communication subsystem or network error has occurred. See the Messages Reference for more information on specific communication error codes.

If you are trying to establish a new connection, check if:

1. The remote database server is cataloged properly on the client node.

2. The database manager configuration file at the server is configured with the proper communication related parameters. If the database manager configuration parameters have been updated at the server, ensure that you stop and then re-start the database manager, for the changes to take effect.

3. The communication subsystems on both client and server nodes are configured and started up properly.

4. The DB2COMM environment variable at the server specifies the communication protocol used by the client.

5. The database manager at the server is successfully started. Start database manager processing at the server should have returned SQL1063, and not SQL5043. If SQL5043 is returned, check the First Failure Service Log (DB2DIAG.LOG) for more information.

6. A communication subsystem or network error has occurred. See the Messages Reference for more information on specific communication error codes.

If the problem persists, consult with your network administrator and/or communication expert to determine the cause of the problem using the set of tokens provided.

sqlcode: -30081

sqlstate: 08001

SQL30082N

Attempt to establish connection failed with security reason "<reason-code>" ("<reason-string>").

Cause: The attempt to connect to the remote database server was rejected due to invalid or incorrect security information. The cause of the security error is described by the <reason-code> and corresponding <reason-string> value.

The following is a list of reason codes and corresponding reason strings:

0 (NOT SPECIFIED)
  The specific security error is not specified.

1 (PASSWORD EXPIRED)
  The password specified in the request has expired.

2 (PASSWORD INVALID)
>   The password specified in the request is not
>   valid.

3 (PASSWORD MISSING)
>   The request did not include a password.

4 (PROTOCOL VIOLATION)
>   The request violated security protocols.

5 (USERID MISSING)
>   The request did not include a userid.

6 (USERID INVALID)
>   The userid specified in the request is not valid.

7 (USERID REVOKED)
>   The userid specified in the request has been
>   revoked.

8 (GROUP INVALID)
>   The group specified in the request is not valid.

9 (USERID REVOKED IN GROUP)
>   The userid specified in the request has been
>   revoked in the group.

10 (USERID NOT IN GROUP)
>   The userid specified in the request is not in the
>   group.

11 (USERID NOT AUTHORIZED AT REMOTE LU)
>   The userid specified in the request is not
>   authorized at the remote Logical Unit.

12 (USERID NOT AUTHORIZED FROM LOCAL LU)
>   The userid specified in the request is not
>   authorized at the remote Logical Unit when
>   coming from the local Logical Unit.

13 (USERID NOT AUTHORIZED TO TP)
>   The userid specified in the request is not
>   authorized to access the Transaction Program.

14 (INSTALLATION EXIT FAILED)
>   The installation exit failed.

15 (PROCESSING FAILURE)
>   Security processing at the server failed.

Action: Ensure that the proper userid and/or password
is supplied.

sqlcode: -30082

sqlstate: 08001

```
SQL30090N

Operation invalid for application execution
environment.  Reason code = "<reason-code>"
```

Cause: The operation is invalid for the application
execution environment. Applications operating in an XA
Distributed Transaction Processing environment (such as
CICS), applications operating with CONNECT type 2
connection settings, and applications updating multiple
data sources in a DataJoiner database have special
restrictions on statements or APIs. The operation
attempted was rejected.

Possible reason codes are:

- 01 - an SQL request that changes data (such as an
  INSERT or a CREATE) was issued against a
  read-only database; or, a stored procedure was
  invoked against a read-only database. Read-only
  databases include the following types:

  – Databases accessed using DRDA when
    operating in a unit of work which has
    connection setting SYNCPOINT TWOPHASE
    and executing in a non-XA/DTP environment
    when either a Synchpoint Manager is not being
    used or the remote DRDA database does not
    support level 2 DRDA protocol.
  – Databases accessed by DRDA in an XA/DTP
    environment when either a Synchpoint Manager
    gateway is unavailable or the remote DRDA
    database does not support level 2 DRDA
    protocol.
  – Databases which were not the first database
    updated when the SYNCPOINT ONEPHASE
    connection setting is in effect for the unit of
    work.

- 02 - an API such as Precompile, Bind, or
  Reorganize Table which causes internal commits
  was issued in an application which has the setting
  CONNECT type 2 or is operating in an XA/DTP
  environment.

- 03 - an SQL OPEN for a held cursor was issued
  while in an XA/DTP environment when using an
  ENCINA or TUXEDO transaction processing
  monitor.

- 04 - a DISCONNECT statement was issued while in
  an XA/DTP environment.

- 05 - A compound SQL statement which contains a
  COMMIT statement was issued in a CONNECT type
  2 or XA/DTP environment.

- 06 - The SET CLIENT API was issued in an
  XA/DTP environment.

- 07 - A second database is being accessed within a unit of work for which two-phase commit coordination is not being supplied by the transaction manager. To ensure data integrity, the operation has been disallowed.

- 08 - An attempt was made to access a database for use of commit coordination from a source different from a concurrently connected database. The two types of coordination cannot be mixed, and operations against the current database are rejected.

- 09 - An attempt was made to execute an XA/DTP local transaction against a database accessed under Synchpoint Manager coordination.

- 10 - An SQL OPEN for a held cursor was issued in either of these cases:

  - XA/DTP environment, or
  - DataJoiner accessing a nickname defined on a two phase commit datasource

  Cursor hold is not supported in these environments.

- 11 - An operation is not supported for pass-through.

- 12 - The update/delete operation requires the existence of a timestamp column and a unique index due to the restriction on the data source.

  For an update/delete operation accessing data sources, Sybase requires the existence of a timestamp column and a unique index if the protocol column in SYSCAT.SERVERS is 'dblib'.

- 13 - A positioned UPDATE or DELETE operation requires the existence of a column in the SELECT list of the cursor, but that column does not exist in the SELECT list of the cursor.

- 14 - An illegal combination of updatable cursors, cursor hold, and isolation level of Repeatable Read exists. Invalid combinations are:

  - Isolation level Repeatable Read and a WITH HOLD cursor

  - A WITH HOLD cursor with FOR UPDATE

- 15 - Reserved for future use

- 16 - Illegal type column and protocol column value combination in SYSCAT.SERVERS.

- 17 - The REORG utility cannot be issued against a nickname.

- 18 - An update request (or, a DDL operation that results in the update of a system catalog table) has been issued that would result in multiple data sources being updated when one or more of the data sources in the unit of work only supports one-phase commit. Possible causes are:

  - An attempt was made to update a data source that only supports one-phase commit, but a different data source has already been updated in the same unit of work.
  - An attempt was made to update a data source that supports two-phase commit, but a different data source that only supports one-phase commit has already been updated in the same unit of work.
  - An attempt was made to update a local DataJoiner table, but a data source that only supports one-phase commit has already been updated in the same unit of work.
  - An attempt was made to update a data source that only supports one-phase commit when the application is operating with a CONNECT type 2 connection setting.

- 19 - The application host variable data type is not supported by the data source in a pass-through session.

- 20 - Operation not valid for the data source.

- 21 - For Informix 5, a given protocol can only be used to access a given database. This reason code is issued when an application tries to access multiple Informix 5 databases with DataJoiner's Informix 5 protocol (data access module).

Action:

Correct the problem by doing one of the following steps:

- For reasons 01, 02, 03, 04, 06, 19, or 21 remove the statement or API which is not supported.

- Alternately, for reasons 01, 02, 03, 04, or 06, execute the application in a different environment which supports the failing statement or API.

- For reason 05 move the COMMIT request outside of the compound statement.

- For reason 07, ensure that only one database is accessed within a unit of work for which EXEC SQL COMMIT or EXEC SQL ROLLBACK will be issued in place of syncpoint requests to an external transaction manager. If multiple databases must be accessed within the unit of work, utilize the commitment control interface provided by the external transaction manager product.

- For reason 08, ensure that all databases being accessed within a unit of work are under the commitment control of the same type of request: external transaction processing monitor (such as

CICS SYNCPOINT) or local COMMIT and
ROLLBACK EXEC SQL.

- For reason 09, do one of the following steps:

  - Execute the transaction as an XA/DTP global
    transaction.
  - Access the database in a non-XA/DTP
    environment.
  - Do not use the services of the Synchpoint
    Manager for the database connection if the
    transaction is read-only.

- For reasons 10 through 17 and 20 (DataJoiner
  users), isolate the problem to the data source failing
  the request (see the *Problem Determination Guide*)
  and examine the restrictions for that data source.

- For reason 18, do one of the following steps:

  - Submit a COMMIT or ROLLBACK before
    issuing an update to a different data source.
  - If multiple data sources need to be updated
    within a unit of work, make sure the
    two_phase_commit server option has been set
    to 'Y' for all the data sources that need to be
    updated. See the *Application Programming and
    SQL Reference Supplement* for information on
    setting the value used for the
    two_phase_commit setting.
  - If the data source to be updated supports only
    one-phase commit and the application is
    operating with a CONNECT type 2 connection

setting, change the application to operate with a
CONNECT type 1 connection setting.

sqlcode: -30090

sqlstate: 25000

SQL30101W

**Bind options specified on REBIND request are
ignored.**

Cause: Bind options were specified on the REBIND
request however the database server does not support
respecification of any bind options. The bind options
specified will be ignored and the options from the
original BIND request will be used.

Action: No action is required. This is a warning
situation only.

If the database server returned an additional warning
SQLCA, the "sqlerrmc" tokens will indicate the following
information about this additional SQLCA in this order:

- sqlcode   (SQL return code)
- sqlstate (universal SQL return code)
- sqlerrp   (product name)
- sqlerrmc (SQL message tokens)

sqlcode: +30101

sqlstate: 01599

Chapter 8. SPM Messages

SPM0400

Indoubt transaction resolution with DBALIAS="<dbalias>" shows heuristic damage - the database rolled back the UOW and the coordinator with LUNAME="<luname>" committed. The transaction is identified by LUWID="<luwid>".

Cause: DB2 was the DRDA2 AS for the transaction identified by "<luwid>". Resolution with the DB2 database shows heuristic damage occurred. The database identified by "<dbalias>" manually resolved the indoubt transaction. The DB2 database at "<luname>" rolled back the transaction. This is inconsistent with the commit decision of the DRDA2 coordinator at "<luname>".

- The application was told that the transaction was committed.
- Any resources updated at the participant were rolled back.

dbalias   The alias name of the DB2 database.

luname   The LU name of the coordinator.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

Indoubt resolution with the participant completes.

Action: Call the database administrator.

Database Administrator Action: Inform the database administrators at both the COORDINATOR "<luname>" and at the DATABASE "<dbalias>" that heuristic damage occurred for the transaction with "<luwid>". DB2 was a DRDA2 AS for the transaction. The DRDA2 AR at COORDINATOR "<luname>" made the decision to COMMIT the database updates made by "<luwid>". The "<dbalias>" PARTICIPANT made a heuristic decision to ROLL BACK the updates made by "<luwid>".

SPM0401

Indoubt transaction resolution with DBALIAS="<dbalias>" shows heuristic damage - the database committed and the coordinator with LUNAME="<luname>" rolled back. The transaction is identified by LUWID="<luwid>".

Cause: DB2 was the DRDA2 AS for the transaction identified by "<luwid>". Resolution with the DB2 DATABASE shows heuristic damage occurred. The database identified by "<dbalias>" manually resolved the indoubt transaction. The "<dbalias>" committed the transaction. This is inconsistent with the roll back decision of the DRDA2 COORDINATOR at "<luname>".

- The application was told that the transaction was rolled back.
- Any resources updated at the participant were committed.

dbalias   The alias name of the DB2 database.

luname   The LU name of the coordinator.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

Indoubt resolution with the participant completes.

Action: Call the database administrator.

Database Administrator Action: Inform the database administrators at both the COORDINATOR "<luname>" and at the DATABASE "<dbalias>" that heuristic damage occurred for the transaction with "<luwid>". DB2 was a DRDA2 AS for the transaction. The DRDA2 AR at COORDINATOR "<luname>" made the decision to roll back the database updates made by "<luwid>". At the PARTICIPANT "<dbalias>" a heuristic decision was made to COMMIT the updates made by "<luwid>".

SPM0402

Indoubt transaction resolution with participant with LUNAME="<luname>" and DBALIAS="<dbalias>" shows heuristic damage - the participant committed and DB2 rolled back. The transaction is identified by LUWID="<luwid>".

Cause: DB2 has coordinator responsibility for the transaction identified by "<luwid>". Resolution with the

SPM0403 SPM0404

participant shows heuristic damage occurred. The participant identified by "<luname>" and "<dbalias>" manually resolved the indoubt transaction. The action taken was to commit the transaction. This is inconsistent with the roll back decision of the coordinator.

- The application was told that the unit of work was rolled back.
- Any resources updated at the participant were committed.

dbalias   The database alias of the participant.

luname   The LU name of the participant.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

Indoubt resolution with the participant completes.

Action: Call the database administrator.

Database Administrator Action: Inform the local database administrator and the database administrator at "<luname>" that heuristic damage occurred for the transaction with "<luwid>". DB2 was the coordinator for the transaction and made the decision to roll back the database updates made by "<luwid>". At "<luname>" a heuristic decision was made to COMMIT the updates made by "<luwid>".

SPM0403

Indoubt transaction resolution with participant with LUNAME="<luname>" and DBALIAS="<dbalias>", shows heuristic damage - the participant rolled back and DB2 committed. The transaction is identified by LUWID="<luwid>".

Cause: DB2 has coordinator responsibility for the transaction identified by "<luwid>". Resolution with the participant shows heuristic damage occurred. The participant identified by "<luname>" and "<dbalias>" manually resolved the indoubt transaction. The action taken was to roll back the UOW. This is inconsistent with the commit decision of the coordinator.

- The application was told that the transaction committed.
- Any resources updated at the participant were rolled back.

luname   The LU name of the participant.

dbalias   The data base alias of the participant.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

Indoubt resolution with the participant completes.

Action: Call the database administrator.

Database Administrator Action: Inform the local database administrator and the database administrator at "<luname>" that heuristic damage occurred for the transaction with "<luwid>". DB2 was the coordinator for the transaction and made the decision to COMMIT the database updates made by "<luwid>". At "<luname>" a heuristic decision was made to ROLL BACK the updates made by "<luwid>".

SPM0404

Protocol error during indoubt transaction resolution with coordinator with LUNAME="<luname1>" - the DB2 database with LUNAME="<luname2>" has an indoubt transaction which is identified by LUWID="<luwid>".

Cause: DB2 was the DRDA2 AS for the transaction identified by "<luwid>". The DB2 transaction associated at the database with LUNAME="<luname2>" is indoubt. A protocol error occurred during indoubt resolution with the coordinator identified by "<luname1>".

luname1   The LU name of the partner acting as the coordinator.

luname2   The LU name of the database where the transaction is indoubt.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

The indoubt transaction remains indoubt. A Resync Protocol Violation trace record is written.

Action: Call the database administrator.

Database Administrator Action: DB2 does not attempt to automatically resolve the indoubt transaction. The transaction must be manually resolved. The commit or abort decision made at the coordinator must be determined so that the same decision can be made at this participant DB2.

Contact the database administrator at the coordinator with "<luname>" and "<dbalias>", to determine whether the transaction committed or aborted.

Use the LIST INDOUBT TRANSACTIONS command at this (the participant) dbalias to resolve the indoubt transaction.

SPM0405

A transaction with LUWID="<luwid>" at the DB2 database with LUNAME="<luname1>" is indoubt because of a communication failure with the coordinator with LUNAME="<luname2>".

Cause: During execution of the two phase commit protocol with the coordinator at "<luname2>", a communication failure occured. Phase 1 of the protocol completed and the transaction at the database with "<luname1>" is indoubt.

The transaction is placed in the indoubt state and appears in the LIST DRDA INDOUBTS TRANSACTIONS report. Periodic attempts are made to reestablish communication with the coordinator for automatic resolution.

luname1   The LU name of the database where the transaction is indoubt.

luname2   The LU name of the coordinator.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the indoubt unit of work is printed following this message.

Periodic attempts will be made to automatically resolve the indoubt transaction.

Action: Determine the cause of the communication failure and have the problem corrected. DB2 periodically attempts to establish communication for automatic resolution. If automatic resolution does not occur in a reasonable amount of time, call the database administrator. Manual resolution of the indoubt transaction might be necessary to release locked resources.

Database Administrator Action: If manual resolution is necessary:

- Determine the commit or abort decision made at the coordinator so that the same decision can be made at the DB2 database. Contact the database administrator at the coordinator with "<luname>" to determine whether the transaction committed or aborted.

- Use the LIST INDOUBTS TRANSACTION command at this (the participant) dbalias to resolve the indoubt transaction.

SPM0406

A transaction with LUWID="<luwid>" at the participant with LUNAME="<luname>" and DBALIAS="<dbalias>" may be indoubt because of a communication failure. DB2 committed.

Cause: During execution of the two phase commit protocol with the participant at "<luname>", a communication failure occured. Phase 1 of the protocol completed and the transaction is prepared for either commit or abort. The decison to commit the transaction was made, but cannot be communicated to the participant at this time. The participant is indoubt.

DB2 becomes responsible for indoubt resolution with the participant. This responsibility appears in the LIST DRDA INDOUBTS TRANSACTION report. Periodic attempts are made to reestablish communication with the participant for automatic resolution.

luname   The LU unit name of the participant.

dbalias   The database alias name of the participant.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

Periodic attempts will be made to automatically resolve the indoubt transaction at the participant.

Action: Determine the cause of the communication failure and have the problem corrected. DB2 periodically attempts to reestablish communication for automatic resolution. If automatic resolution does not occur in a reasonable amount of time, call the database administrator. Manual resolution of the transaction might be necessary at the participant to release locked resources.

Database Administrator Action: If manual resolution is necessary, inform the database administrator at the participant that the decision is commit.

SPM0407

Automatic resolution of the transaciton with LUWID="<luwid>" with the coordinator at LUNAME="<luname1>" resulted in commit. The DB2 database is at LUNAME="<luname2>".

Cause: The indoubt transaction at the database identified by "<luname2>" was automatically resolved by communication with the coordinator identified by "<luname1>". The transaction was committed.

SPM0408   SPM0410

luname1   The LU name of the coordinator.

luwid   The SNA logical unit of work id of the
           transaction.

luname2   The LU name of the database where the
           changes were made.

The XID associated with the unit of work is printed
following this message.

Processing continues normally.

Cause: One or more indoubt transactions exist with
"<luname>". DB2 attempted to automatically resolve the
indoubt transaction but a communication error occured.

protocol   The communication protocol being used.
            Only "APPC" is supported.

api   The communication api being used. Either
            "CM2 APPC" or "SNA/6000 OS
            Subroutines" may be specified.

codes   For "CM2 APPC" api, the rc1, rc2, and rc3
            contain the primary return code, secondary
            return code and possibly an SNA sense
            code. For "SNA/6000 OS Subroutines" api,
            rc1 contains the errno global variable.

Action: Determine the cause of the communication
failure and have the problem corrected. DB2 periodically
attempts to reestablish communication for automatic
resolution. If automatic resolution does not occur in a
reasonable amount of time, call the database
administrator. Manual resolution of the transaction might
be necessary at the participant to release locked
resources.

Database Administrator Action: If manual resolution is
necessary, inform the database administrator at the
participant that the decision is commit.

Cause: An indoubt transaction exists with the partner at
"<luname>". DB2 is unable to resolve the indoubt
transaction because the partner has lost all knowledge
of indoubt transactions due to a previous cold start.

luwid   The SNA logical unit of work id of the
           transaction.

status   The commit status of the transaction as
           known by DB2. The commit status may be
           indoubt, committed, or rolled back.

responsibility DB2 transaction responsibility. DB2 may be
           the coordinator or may be the participant.

Action: There is probably inconsistent data at the
coordinator and participant. Inform database
administrator of the status of the transaction.

Database Administrator Action: Manual resolution is
necessary. The heuristic decision (that is, to commit or
roll back the transaction) should be coordinated with any
other participants and/or the coordinator. The existence
of other participants might not be easy to determine.
The information might be available in the coordinators
recovery log even though the coordinator performed a
cold start.

The commit or abort decision provided using the LIST
INDOUBT TRANSACTIONS command for the
transaction are propagated to all downstream
participants, if any.

Cause: An attempt to make a warm start connection
with a partner was rejected because the partner
specified a different set of sync point parameters than
the ones that were in use when communications were
lost. DB2 has knowledge of indoubt transactions that

involve the partner as either the coordinator or a participant. This error might be a recoverable error if the partner can restart with the original sync point parameters. If this is not possible, then the partner must perform a cold start connection with DB2.

luname   The LU name of the partner.

oldourname The log name for this DB2 subsystem. This value is null if the partner did not initiate the log name exchange.

newourname The log name for this DB2 subsystem, as remembered by our partner and sent to us in the attempted warm start connection. This value is null if the partner did not initiate the log name exchange.

oldtheirname The log name used by the partner during the last connection, as remembered by DB2.

newtheirname The log name used by the partner in the attempted warm start connection.

oldpa   The sync point protocol used during the last connection, as remembered by the DB2 (PA - Presumed Abort, PN - Presumed Nothing).

newpa   The sync point protocol used during the last connection, as remembered by the partner (PA - Presumed Abort, PN - Presumed Nothing).

oldfb   The use of the PS header byte 2 during the last connection, as remembered by DB2 (F - used as flag byte, N - not used as flag byte).

newfb   The use of the PS header byte 2 during the last connection, as remembered by the partner (F - used as flag byte, N - not used as flag byte).

oldccls   Whether the luname of the conversation correlator was exchanged as part of the sync point protocol during the last connection, as remembered by DB2. (E - was exchanged, N - was not exchanged).

newccls   Whether the luname of the conversation correlator was exchanged as part of the sync point protocol during the last connection, as remembered by the partner (E - was exchanged, N - was not exchanged).

The connection with the partner is rejected. DB2 retains indoubt knowledge.

Action: Call the database administrator.

Database Administrator Action: Contact the database administrator at the partner "<luname>" and determine if it is possible for the partner to perform a warm start with same sync point parameters as ours ('oldourname', 'oldtheirname', 'oldpa', 'oldfb', 'oldccls'). If this is possible, the next attempt to connect will succeed.

If this cannot be done, then there are two other possible solutions:

- Have the partner "<luname>" perform a cold start connection with this DB2 subsystem.

- Force the next connection with the partner to be a cold start connection.

SPM0411

Cold start connection by coordinator with LUNAME="<luname>" accepted. Indoubt transactions need manual resolution.

Cause: DB2 was the DRDA2 AS and has participant responsibility for indoubt transactions. The coordinator informed DB2 that it performed a cold start operation and lost all knowledge of indoubt transactions. The indoubt transactions at this DB2 must be manually resolved with the LIST INDOUBT TRANSACTIONS command.

dbalias   The database alias name of the coordinator.

The connection with the partner is accepted. A trace record is written.

Action: Call the database administrator.

Database Administrator Action: DB2 is a participant with one or more indoubt transactions where the coordinator is "<luname>". The DBMS at "<luname>" performed a cold start. The DB2 participant assumes that the coordinator recovery log was lost or damaged and indoubt transaction resolution cannot be achieved. There is probably inconsistent data at the coordinator.

The heuristic decision (that is, to commit or abort the transaction should be coordinated with any other participants. The existence of other participants might not be easy to determine. The information might be available in the coordinators recovery log even though the coordinator performed a cold start.

The commit or abort decision provided using the LIST INDOUBT TRANSACTIONS command for the transaction are propagated to all downstream participants, if any.

SPM0412 SPM0413

SPM0412

Protocol error detected in syncpoint communications with coordinator with LUNAME="<luname1>". The transaction with LUWID="<luwid>" at the DB2 database with LUNAME="<luname2>" may be indoubt.

Cause: DB2 is a participant in the transaction. A protocol error occurred during the SNA sync point exchange with the coordinator identified by "<luname>". The protocol error fits into one of the following categories:

- An unknown SNA PS header was received.

- An SNA PS header was received in the wrong sequence.

- An invalid LU6.2 verb sequence was used to deliver the SNA PS header.

- A DEALLOCATE TYPE(ABEND_SVC) was received from the coordinator, indicating that the coordinator detected a protocol violation in the PS headers transmitted by DB2.

luname1   The LU name of the coordinator.

luname2   The LU name of the database where the transaction is indoubt.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

If the protocol error was detected before the commit decision, the transaction at the database with LU name="<luname2>" may be indoubt. DB2 does not automatically resolve such an indoubt transaction because of the protocol error.

If the protocol error was detected after the commit decision, the transaction either completed commit or abort processing.

A Syncpoint Protocol Violation trace is written.

Action: The database administrator might need to manually resolve the indoubt transaction.

Database Administrator Action: Determine if the transaction is indoubt. If it is indoubt, it must be manually resolved using the LIST INDOUBT TRANSACTIONS command. The commit or abort decision made at the coordinator must be determined so that the same decision can be made at DB2.

Contact the database administrator at the coordinator dbalias to determine whether the transaction with LUWID="<luwid>" committed or aborted.

If the coordinator system is another DB2, then the following steps can be taken at the DB2 coordinator to determine the commit or abort decision.

SPM0413

Protocol error detected in syncpoint communications with participant with LUNAME="<luname>" and DBALIAS="<dbalias>". The transaction with LUWID="<luwid>" may be indoubt at the participant. DB2 committed.

Cause: DB2 is the coordinator of the transaction. A protocol error occurred during the SNA sync point exchange with the participant identified by "<luname>" and "<dbalias>". The protocol error fits into one of the following categories:

- An unknown SNA PS header was received.

- An SNA PS header was received in the wrong sequence.

- An invalid LU6.2 verb sequence was used to deliver the SNA PS header.

- A DEALLOCATE TYPE(ABEND_SVC) was received from the participant, indicating that the participant detected a protocol violation in the PS headers transmitted by DB2.

luname   The LU name of the participant.

dbalias   The database alias name of the participant.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

If application was told that the transaction committed.

There may be an indoubt transaction at the participant and if so, the indoubt transaction must be manually resolved. DB2 does not automatically resolve the indoubt transaction because of the protocol error.

A Syncpoint Protocol Violation trace record is written.

Action: Call the database administrator. The participant might need to manually resolve the indoubt transaction.

SPM0414

Protocol error during indoubt transaction resolution with participant with LUNAME="<luname>" and DBALIAS="<dbalias>". The transaction with LUWID="<luwid>" may be indoubt at the participant. DB2 rolled back.

Cause: DB2 has coordinator responsibility for the transaction which was rolled back. A protocol error occurred during indoubt resolution with the participant identified by "<luname>" and "<dbalias>".

The transaction at the participant remains indoubt. DB2 will not attempt to automatically resolve the indoubt transaction because of the protocol violation.

luname   The LU name of the participant.

dbalias   The database alias name of the participant.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

A Resync Protocol Violation trace record is written.

Action: Call the database administrator. The participant might need to manually resolve the indoubt transaction.

Database Administrator Action: If the transaction is indoubt at the participant, it must be manually (heuristically) resolved.

SPM0415

Automatic resolution of the transaction with LUWID="<luwid>" with the coordinator at LUNAME="<luname1>" resulted in roll back. The DB2 database is at LUNAME="<luname2>".

Cause: The indoubt transaction at the database identified by "<luname2>" was automatically resolved by communication with the coordinator identified by "<luname1>". The transaction was rolled back.

luname1   The LU name of the coordinator.

luwid   The SNA logical unit of work id of the transaction.

luname2   The LU name of the database where the changes were made.

The XID associated with the unit of work is printed following this message.

Processing continues normally.

SPM0416

Cold start connection rejected by partner with LUNAME "<luname>".

Cause: DB2 attempted to make a cold-start connection with a partner dbalias. The partner rejected this attempted connection.

luname   The LU name of the coordinator.

The connection was not made.

Action: Call the database administrator.

Database Administrator Action: DB2 is not able to connect the partner "<luname>" until the partner "<luname>" allows a cold-start connection with DB2. Contact the database administrator at the partner "<luname>".

Contact your IBM Support Center for further assistance.

SPM0417

Protocol error detected in syncpoint communications with participant with LUNAME="<luname>" and DBALIAS="<dbalias>". The transaction with LUWID="<luwid>" may be indoubt at the participant. DB2 rolled back.

Cause: DB2 is the coordinator of the transaction. A protocol error occurred during the SNA sync point exchange with the participant identified by "<luname>" and "<dbalias>". The protocol error fits into one of the following categories:

- An unknown SNA PS header was received.

- An SNA PS header was received in the wrong sequence.

- An invalid LU6.2 verb sequence was used to deliver the SNA PS header.

- A DEALLOCATE TYPE(ABEND_SVC) was received from the participant, indicating that the participant detected a protocol violation in the PS headers transmitted by DB2.

luname   The LU name of the participant.

dbalias   The database alias name of the participant.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

If application was told that the transaction rolled back.

SPM0420　SPM0423

There may be an indoubt transaction at the participant and if so, the indoubt transaction must be manually resolved. DB2 does not automatically resolve the indoubt transaction because of the protocol error.

A Syncpoint Protocol Violation trace record is written.

Action: Call the database administrator. The participant might need to manually resolve the indoubt transaction.

SPM0420

Cold start connection by participant LUNAME="<luname>" accepted. Possible damage.

Cause: DB2 has coordinator responsibility for indoubt transactions at a participant and just connected with the participant, which lost all knowledge of indoubt transactions because of a previous cold start. There might be damage at the participant.

luname　　The LU name of the participant where there might be damage.

The connection with the partner is accepted.

Action: Call the database administrator.

Database Administrator Action: DB2 is the coordinator with indoubt transaction resolution responsibility for one or more indoubt units of work at "<luname>". The DBMS at "<luname>" performed a cold start. DB2 assumes that the participant recovery log was lost or damaged and indoubt transaction resolution cannot be achieved. There is probably inconsistent data at the participant. Minimally, the participant might not completely reflect the final outcome of the transactions that were indoubt at the time the failure occurred.

SPM0421

SNA XLN protocol violation by partner with LUNAME="<luname>"

Cause: DB2 detected a protocol violation in the SNA Exchange Log Names (XLN) exchange with the partner at the specified "<luname>".

luname　　　The LU name of the partner that sent the invalid XLN message.

The attempt to connect with the remote site fails. An XLN Protocol Violation trace record is written.

Action: Contact the system programmer for the remote site. The invalid XLN message is recorded in the trace record. The system logic error that causes the invalid XLN message must be corrected at the remote site.

SPM0422

Warm start connection by partner LUNAME="<luname>" rejected because the partner remembers our log name incorrectly. Our log name is "<name1>" and the partner remembers is as "<name2>".

Cause: An attempt to make a warm start connection with a partner was rejected because the partner specified our log name as name2. Our log name is name1, which is the luname of the local DB2. This error might be a recoverable error if the partner can restart with our log name as name1. If this is not possible, then the partner must perform a cold start connection with DB2.

luname　　The LU name of the partner with which the connection failed.

name1　　Our log name, as remembered by us.

name2　　Our log name, as remembered by our partner and sent to us in the attempted warm start connection.

The connection with the partner is rejected.

Action: Call the database administrator.

Database Administrator Action: Contact the database administrator at the partner "<luname>" and determine if it is possible for the partner to perform a warm start with our log name specified as the luname of this DB2. If this is possible, the next attempt to connect will succeed. Or have the partner "<luname>" perform a cold start connection with DB2.

SPM0423

Automatic resolution of the transaction with LUWID="<luwid>" with the partner at LUNAME="<luname>" and DBALIAS="<dbalias>" resulted in commit.

Cause: The indoubt unit of work was automatically resolved by communication with the participant. The participant has been notified of the commit decision.

luname　　The LU name of the participant.

dbalias　　The database alias name of the participant.

luwid　　The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

Processing continues normally.

SPM0424

Automatic resolution of the transaction with LUWID="<luwid>" with the participant at LUNAME="<luname>" and DBALIAS="<dbalias>" resulted in roll back.

Cause: The indoubt unit of work was automatically resolved by communication with the participant. The participant has been notified of the roll back decision.

luname   The LU name of the participant.

dbalias   The database alias name of the participant.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

Processing continues normally.

SPM0425

A transaction with LUWID="<luwid>" at the participant with LUNAME="<luname>" and DBALIAS="<dbalias>" may be indoubt because of a communication failure. DB2 rolled back.

Cause: During execution of the two phase commit protocol with the participant at "<luname>", a communication failure occured. Phase 1 of the protocol completed and the transaction is prepared for either commit or abort. The decison to roll back the transaction was made, but cannot be communicated to the participant at this time. The participant is indoubt.

DB2 becomes responsible for indoubt resolution with the participant. This responsibility appears in the LIST DRDA INDOUBTS TRANSACTION report. Periodic attempts are made to reestablish communication with the participant for automatic resolution.

luname   The LU unit name of the participant.

dbalias   The database alias name of the participant.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

Periodic attempts will be made to automatically resolve the indoubt transaction at the participant.

Action: Determine the cause of the communication failure and have the problem corrected. DB2 periodically attempts to reestablish communication for automatic

resolution. If automatic resolution does not occur in a reasonable amount of time, call the database administrator. Manual resolution of the transaction might be necessary at the participant to release locked resources.

Database Administrator Action: If manual resolution is necessary, inform the database administrator at the participant that the decision is roll back.

SPM0426

Protocol error detected during indoubt transaction resolution with participant at LUNAME="<luname>" and DBALIAS="<dbalias>". The transaction with LUWID="<luwid>" may be indoubt at the participant. DB2 committed.

Cause: DB2 has coordinator responsibility for the transaction which was committed. A protocol error occurred during indoubt resolution with the participant identified by "<luname>" and "<dbalias>".

The transaction at the participant remains indoubt. DB2 will not attempt to automatically resolve the indoubt transaction because of the protocol violation.

luname   The LU name of the participant.

dbalias   The database alias name of the participant.

luwid   The SNA logical unit of work id of the transaction.

The XID associated with the unit of work is printed following this message.

A Resync Protocol Violation trace record is written.

Action: Call the database administrator. The participant might need to manually resolve the indoubt transaction.

Database Administrator Action: If the transaction is indoubt at the participant, it must be manually (heuristically) resolved.

SPM0434

DB2 syncpoint manager not available - incorrect communications level.

Cause: The local communications release level is older than the minimum release level or the communication manager is incorrectly configured to support APPC SYNCLEVEL(SYNC) conversations.

The attempt to create a protected conversation fails.

SPM0438 SPM0439

Action: Install and configure the correct communications level required to support SYNCLEVEL(SYNC) conversations.

SPM0438

DB2 Syncpoint Manager recovery log is bad.

Cause: The DB2 Syncpoint Manager recovery log is inconsistent and cannot be used to perform recovery during DB2 start up processing.

Action: Indoubt transactions may exist at DRDA2 application servers. These indoubt transactions must be recovered manually.

Call the database administrator.

Database Administrator Action: To start the Syncpoint Manager, erase the spmlog directory and start DB2. This will cause DB2 to create new syncpoint log files and to establish cold start connections with all DRDA2 application servers.

SPM0439

DB2 Syncpoint Manager unrecoverable error while attempting to write to the SPM recovery log.

Cause: The DB2 Syncpoint Manager log is inconsistent and cannot be used. An unrecoverable error has been detected while attempting to write to the SPM log during DB2 processing.

Action: The DB2 Syncpoint Manager will not allow any new synclevel(twophase) connections. Issue the LIST DRDA INDOUBT TRANSACTIONS command to determine the status of any indoubt transactions.

Call the database administrator.

Database Administrator Action: To start the Syncpoint Manager, erase the spmlog directory and start DB2. This will cause DB2 to create new syncpoint log files and to establish cold start connections with all DRDA2 application servers.

Chapter 9. SQLSTATE Messages

Table 3. SQLSTATE Class Codes

Class Code	Meaning	For subcodes, refer to...
00	Unqualified Successful Completion	Table 4 on page 348
01	Warning	Table 5 on page 348
02	No Data	Table 6 on page 349
07	Dynamic SQL Error	Table 7 on page 350
08	Connection Exception	Table 8 on page 350
09	Triggered Action Exception	Table 9 on page 350
0A	Feature Not Supported	Table 10 on page 350
0F	Invalid Token	Table 11 on page 350
21	Cardinality Violation	Table 12 on page 351
22	Data Exception	Table 13 on page 351
23	Constraint Violation	Table 14 on page 352
24	Invalid Cursor State	Table 15 on page 352
25	Invalid Transaction State	Table 16 on page 352
26	Invalid SQL Statement Identifier	Table 17 on page 353
28	Invalid Authorization Specification	Table 18 on page 353
2D	Invalid Transaction Termination	Table 19 on page 353
2E	Invalid Connection Name	Table 20 on page 353
34	Invalid Cursor Name	Table 21 on page 353
38	External Function Exception	Table 22 on page 353
39	External Function Call Exception	Table 23 on page 354
40	Transaction Rollback	Table 24 on page 354
42	Syntax Error or Access Rule Violation	Table 25 on page 354
44	WITH CHECK OPTION Violation	Table 26 on page 359
51	Invalid Application State	Table 27 on page 359
54	SQL or Product Limit Exceeded	Table 28 on page 360
55	Object Not in Prerequisite State	Table 29 on page 361
56	Miscellaneous SQL or Product Error	Table 30 on page 362
57	Resource Not Available or Operator Intervention	Table 31 on page 362
58	System Error	Table 32 on page 363

Table 4. Class Code 00: Unqualified Successful Completion

SQLSTATE Value	Meaning
00000	Execution of the SQL statement was successful and did not result in any type of warning or exception condition.

Table 5 (Page 1 of 2). Class Code 01: Warning

SQLSTATE Value	Meaning
01002	A DISCONNECT error occurred.
01003	Null values were eliminated from the argument of a column function.
01004	The value of a string was truncated when assigned to a host variable.
01005	Insufficient number of entries in an SQLDA.
01503	The number of result columns is larger than the number of host variables provided.
01504	The UPDATE or DELETE statement does not include a WHERE clause.
01506	An adjustment was made to a DATE or TIMESTAMP value to correct an invalid date resulting from an arithmetic operation.
01509	Blocking was cancelled for a cursor because there is insufficient storage in the user virtual machine.
01526	Isolation level has been escalated.
01539	Connection is successful but only SBCS characters should be used.
01543	A duplicate constraint has been ignored.
01545	An unqualified column name has been interpreted as a correlated reference.
01550	The index was not created, because an index with the specified description already exists.
01562	The new path to the log (newlogpath) in the database configuration file is invalid.
01563	The current path to the log file (logpath) is invalid. The log file path is reset to the default.
01586	Setting OFF the constraints for a parent table in a referential structure has resulted in one or more descendent tables being automatically placed into a check-pending state
01589	A statement contains redundant specifications.
01592	In a CREATE FUNCTION statement that references a SOURCE function, either: • an input parameter has a larger length, precision, or scale than that of the corresponding parameter of the source function; or • the RETURNS or CAST FROM parameter has a smaller length, precision, or scale than that of the source function; or • the CAST FROM parameter in the CREATE FUNCTION statement has a larger length, precision, or scale than that of the RETURNS parameter. Truncation may occur at run time (and may result in an error at that time).

Table 5 (Page 2 of 2). Class Code 01: Warning

SQLSTATE Value	Meaning
01594	Insufficient number of entries in an SQLDA for ALL information (i.e. not enough descriptors to return the distinct name).
01595	The view has replaced an existing, invalidated view.
01596	Comparison functions were not created for a distinct type based on a long string data type.
01598	An attempt has been made to activate an active event monitor or deactivate an inactive event monitor.
01599	Bind options were ignored on REBIND.
01602	The optimization level has been reduced.
01603	CHECK DATA processing found constraint violations and moved them to exception tables.
01604	The SQL statement was explained and not executed.
01605	A recursive common table expression may contain an infinite loop.
01606	The node or system database directory is empty.
01607	The difference between the times on nodes in a read-only transactions exceed the defined threshold.
01610	One or more ad hoc result sets were returned from the procedure.
01611	The cursor that was closed has been re-opened on the next result set within the chain.
01620	Some base tables of UNION ALL may be the same table
01621	The retrieved LOB value may have been changed.
01Hxx	Valid warning SQLSTATEs returned by a user-defined function or external procedure CALL.

Table 6. Class Code 02: No Data

SQLSTATE Value	Meaning
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.

Table 7. Class Code 07: Dynamic SQL Error

SQLSTATE Value	Meaning
07001	The number of host variables is not the same as the number of parameter markers.
07002	The call parameter list or control block is invalid.
07003	The statement identified in the EXECUTE statement is a select-statement, or is not in a prepared state.
07004	The USING clause is required for dynamic parameters.
07005	The statement name of the cursor identifies a prepared statement that cannot be associated with a cursor.
07006	An input host variable cannot be used, because of its data type.

Table 8. Class Code 08: Connection Exception

SQLSTATE Value	Meaning
08001	The application requester is unable to establish the connection.
08002	The connection already exists.
08003	The connection does not exist.
08004	The application server rejected establishment of the connection.
08502	The CONNECT statement issued by an application process running with a SYNCPOINT of TWOPHASE has failed, because no transaction manager is available.

Table 9. Class Code 09: Triggered Action Exception

SQLSTATE Value	Meaning
09000	A triggered SQL statement failed.

Table 10. Class Code 0A: Feature Not Supported

SQLSTATE Value	Meaning
0A001	The CONNECT statement is invalid, because the process is not in the connectable state.

Table 11. Class Code 0F: Invalid Token

SQLSTATE Value	Meaning
0F001	The LOB token variable does not currently represent any value.

Table 12. Class Code 21: Cardinality Violation

SQLSTATE Value	Meaning
21000	The result of a SELECT INTO is a result table of more than one row, or the result of the subquery of a basic predicate is more than one value.
21501	A multiple-row INSERT into a self-referencing table is invalid.
21502	A multiple-row UPDATE of a primary key is invalid.
21504	A multiple-row DELETE from a self-referencing table with a delete rule of RESTRICT or SET NULL is invalid.

Table 13. Class Code 22: Data Exception

SQLSTATE Value	Meaning
22001	Character data, right truncation occurred; for example, an update or insert value is a string that is too long for the column, or a datetime value cannot be assigned to a host variable, because it is too small.
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.
22003	A numeric value is out of range.
22007	An invalid datetime format was detected; that is, an invalid string representation or value was specified.
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.
22011	A substring error occurred; for example, an argument of SUBSTR is out of range.
22012	Division by zero is invalid.
22018	The character value for the CAST, DECIMAL, FLOAT, or INTEGER scalar function is invalid.
22019	The LIKE predicate has an invalid escape character.
22021	A character is not in the coded character set.
22024	A NUL-terminated input host variable or parameter did not contain a NUL.
22025	The LIKE predicate string pattern contains an invalid occurrence of an escape character.
22501	The length control field of a variable length string is negative or greater than the maximum.
22504	A mixed data value is invalid.
22506	A reference to a datetime special register is invalid, because the TOD clock is malfunctioning or the operating system timezone parameter is out of range.
22522	A CCSID value is not valid at all, not valid for the data type or subtype, or not valid for the encoding scheme.

Table 14. Class Code 23: Constraint Violation

SQLSTATE Value	Meaning
23001	The update or delete of a parent key is prevented by a RESTRICT update or delete rule.
23502	An insert or update value is null, but the column cannot contain null values.
23503	The insert or update value of a foreign key is invalid.
23504	The update or delete of a parent key is prevented by a NO ACTION update or delete rule.
23505	A violation of the constraint imposed by a unique index or a unique constraint occurred.
23511	A parent row cannot be deleted, because the check constraint restricts the deletion.
23512	The check constraint cannot be added, because the table contains rows that do not satisfy the constraint definition.
23513	The resulting row of the INSERT or UPDATE does not conform to the check constraint definition.
23514	Check data processing has found constraint violations.
23515	The unique index could not be created or unique constraint added, because the table contains duplicate values of the specified key.
23520	The foreign key cannot be defined, because all of its values are not equal to a parent key of the parent table.
23521	The update of a catalog table violates an internal constraint.

Table 15. Class Code 24: Invalid Cursor State

SQLSTATE Value	Meaning
24501	The identified cursor is not open.
24502	The cursor identified in an OPEN statement is already open.
24504	The cursor identified in the UPDATE, DELETE, SET, or GET statement is not positioned on a row.
24506	The statement identified in the PREPARE is the statement of an open cursor.
24516	A cursor has already been assigned to a result set.

Table 16. Class Code 25: Invalid Transaction State

SQLSTATE Value	Meaning
25000	An update operation is invalid for the application execution environment.
25501	The statement is only allowed as the first statement in a unit of work.

Table 17. Class Code 26: Invalid SQL Statement Identifier

SQLSTATE Value	Meaning
26501	The statement identified does not exist.

Table 18. Class Code 28: Invalid Authorization Specification

SQLSTATE Value	Meaning
28000	Authorization name is invalid.

Table 19. Class Code 2D: Invalid Transaction Termination

SQLSTATE Value	Meaning
2D521	SQL COMMIT or ROLLBACK are invalid in the current operating environment.
2D528	Dynamic COMMIT is invalid for the application execution environment.
2D529	Dynamic ROLLBACK is invalid for the application execution environment.

Table 20. Class Code 2E: Invalid Connection Name

SQLSTATE Value	Meaning
2E000	Connection name is invalid.

Table 21. Class Code 34: Invalid Cursor Name

SQLSTATE Value	Meaning
34000	Cursor name is invalid.

Table 22 (Page 1 of 2). Class Code 38: External Function Exception

SQLSTATE Value	Meaning
38xxx	Valid error SQLSTATEs returned by a user-defined function, external procedure, or trigger.
38502	The external function is not allowed to execute SQL statements.
38503	A user-defined function has abnormally terminated (abend).
38504	A user-defined function has been interrupted by the user to stop a probable looping condition.

Table 22 (Page 2 of 2). Class Code 38: External Function Exception

SQLSTATE Value	Meaning
38552	A function in the SYSFUN schema (supplied by IBM) has abnormally terminated.
	One of the following reason codes can be found in the message text:

01	Numeric value out of range
02	Division by zero
03	Arithmetic overflow or underflow
04	Invalid date format
05	Invalid time format
06	Invalid timestamp format
07	Invalid character representation of a timestamp duration
08	Invalid interval type (must be one of 1, 2, 4, 8, 16, 32, 64, 128, 256)
09	String too long
10	Length or position in string function out of range
11	Invalid character representation of a floating point number

Table 23. Class Code 39: External Function Call Exception

SQLSTATE Value	Meaning
39001	A user-defined function has returned an invalid SQLSTATE.
39501	The eye-catcher associated with an argument was modified.

Table 24. Class Code 40: Transaction Rollback

SQLSTATE Value	Meaning
40001	Deadlock or timeout with automatic rollback occurred.
40003	The statement completion is unknown.
40504	A system error has caused the unit of work to be rolled back.

Table 25 (Page 1 of 6). Class Code 42: Syntax Error or Access Rule Violation

SQLSTATE Value	Meaning
42501	The authorization ID does not have the privilege to perform the specified operation on the identified object.
42502	The authorization ID does not have the privilege to perform the operation as specified.
42504	A specified privilege cannot be revoked from a specified authorization-name.
42506	Owner authorization failure occurred.
42508	The specified database privileges cannot be granted to PUBLIC.
42601	A character, token, or clause is invalid or missing.
42602	A character that is invalid in a name has been detected.

Table 25 (Page 2 of 6). Class Code 42: Syntax Error or Access Rule Violation

SQLSTATE Value	Meaning
42603	An unterminated string constant has been detected.
42604	An invalid numeric or string constant has been detected.
42605	The number of arguments specified for a scalar function is invalid.
42606	An invalid hexadecimal constant has been detected.
42607	An operand of a column function is invalid.
42608	The use of NULL or DEFAULT in VALUES is invalid.
42609	All operands of an operator or predicate are parameter markers.
42610	A parameter marker is not allowed.
42611	The column or argument definition is invalid.
42612	The statement string is an SQL statement that is not acceptable in the context in which it is presented.
42613	Clauses are mutually exclusive.
42614	A duplicate keyword is invalid.
42615	An invalid alternative was detected.
42617	The statement string is blank or empty.
42618	A host variable is not allowed.
42621	The check constraint is invalid.
42622	A name or label is too long.
42625	A CASE expression is invalid.
42701	A duplicate column name in an INSERT or UPDATE statement was detected.
42702	A column reference is ambiguous, because of duplicate names.
42703	An undefined column name was detected.
42704	An undefined object or constraint name was detected.
42705	An undefined server-name was detected.
42707	A column name in ORDER BY does not identify a column of the result table.
42709	A duplicate column name in a PRIMARY, UNIQUE, or FOREIGN KEY clause was detected.
42710	A duplicate object or constraint name was detected.
42711	A duplicate column name was detected in the object definition or ALTER TABLE statement.
42712	A duplicate table designator was detected in the FROM clause.
42720	The nodename for the remote database was not found in the node directory.
42723	A function with the same signature already exists in the schema.
42724	Unable to access an external program used for a user-defined function or a procedure.

Table 25 (Page 3 of 6). Class Code 42: Syntax Error or Access Rule Violation

SQLSTATE Value	Meaning
42725	A function was referenced directly (not by either signature or by specific instance name), but there is more than one specific instance of that function.
42726	Duplicate names for named derived tables were detected.
42727	No default primary tablespace exists for the new table.
42728	A duplicate node was detected in the nodegroup definition.
42729	The node is not defined.
42730	The container name is already used by another tablespace.
42731	The container name is already used by this tablespace.
42732	A duplicate schema name in the SET CURRENT PATH statement was detected.
42802	The number of insert or update values is not the same as the number of columns.
42803	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.
42804	The result expressions in a CASE expression are not compatible.
42805	An integer in the ORDER BY clause does not identify a column of the result table.
42806	A value cannot be assigned to a host variable, because the data types are not compatible.
42807	The INSERT, UPDATE, or DELETE is not permitted on this object.
42808	A column identified in the INSERT or UPDATE statement is not updateable.
42809	The identified object is not the type of object to which the statement applies.
42810	A view is identified in a FOREIGN KEY clause.
42811	The number of columns specified is not the same as the number of columns in the SELECT clause.
42813	WITH CHECK OPTION cannot be used for the specified view.
42815	The data type, length, scale, value, or CCSID is invalid.
42816	A datetime value or duration in an expression is invalid.
42818	The operands of an operator or function are not compatible.
42819	An operand of an arithmetic operation or an operand of a function that requires a number is not a number.
42820	A numeric constant is too long, or it has a value that is not within the range of its data type.
42821	An update or insert value is not compatible with the column.
42823	The SELECT clause of a subquery specifies multiple columns.
42824	An operand of LIKE is not a string, or the first operand is not a column.
42825	The rows of UNION, INTERSECT, EXCEPT, or VALUES do not have compatible columns.

Table 25 (Page 4 of 6). Class Code 42: Syntax Error or Access Rule Violation

SQLSTATE Value	Meaning
42826	The rows of UNION, INTERSECT, EXCEPT, or VALUES do not have the same number of columns.
42827	The table identified in the UPDATE or DELETE is not the same table designated by the cursor.
42828	The table designated by the cursor of the UPDATE or DELETE statement cannot be modified, or the cursor is read-only.
42829	FOR UPDATE OF is invalid, because the result table designated by the cursor cannot be modified.
42830	The foreign key does not conform to the description of the parent key.
42831	A column of a primary or unique key cannot allow null values.
42832	The operation is not allowed on system objects.
42834	SET NULL cannot be specified, because the foreign key does not allow null values.
42835	Cyclic references cannot be specified between named derived tables.
42836	The specification of a recursive, named derived table is invalid.
42838	An invalid use of a tablespace was detected.
42839	Indexes and long columns cannot be in separate tablespaces from the table.
42840	An invalid use of the AS CAST option was detected.
42841	A parameter marker can not be a user-defined type.
42842	A column definition is invalid, because a specified option is inconsistent with the column description.
42845	An invalid use of a VARIANT or EXTERNAL ACTION function was detected.
42846	Cast from source type to target type is not supported.
42852	The privileges specified in GRANT or REVOKE are invalid or inconsistent. (For example, GRANT ALTER on a view.)
42863	An undefined host variable in REXX has been detected.
42866	The data type in either the RETURNS clause of the CAST FROM clause in the CREATE FUNCTION statement is not appropriate for the data type returned by the source function.
42877	The column name cannot be qualified.
42878	An invalid function or procedure name was used with the EXTERNAL keyword.
42879	The data type of one or more input parameters in the CREATE FUNCTION statement is not appropriate for the corresponding data type in the source function.
42880	The CAST TO and CAST FROM data types are incompatible, or would always result in truncation of a fixed string.
42882	The specific instance name qualifier is not equal to the function name qualifier.
42883	No function was found with a matching signature.

Table 25 (Page 5 of 6). Class Code 42: Syntax Error or Access Rule Violation

SQLSTATE Value	Meaning
42884	No function or procedure was found with the specified name and compatible arguments.
42885	The number of input parameters specified on a CREATE FUNCTION statement does not match the number provided by the function named in the SOURCE clause.
42888	The table does not have a primary key.
42889	The table already has a primary key.
42890	A column list was specified in the references clause, but the identified parent table does not have a unique constraint with the specified column names.
42893	The object or constraint cannot be dropped, because other objects are dependent on it.
42894	The DEFAULT value is invalid.
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.
428A0	An error occurred with the sourced function on which the user-defined function is based.
428A1	Unable to access a file referenced by a host file variable.
428A2	A table cannot be assigned to a multi-node node group, because it does not have a partition key.
428A3	An invalid path has been specified for an event monitor.
428A4	An invalid value has been specified for an event monitor option.
428A5	An exception table named in a SET CONSTRAINTS statement either does not have the proper structure, or it has been defined with constraints or triggers.
428A6	An exception table named in a SET CONSTRAINTS statement cannot be the same as one of the tables being checked.
428A7	There is a mismatch in the number of tables being checked and in the number of exception tables specified in the SET CONSTRAINTS statement.
428A8	Cannot reset the check-pending state using the SET CONSTRAINTS statement on a descendent table while a parent table is in the check-pending state.
428A9	The node range is invalid.
428B2	The path name for the container is not valid.
428B3	An invalid SQLSTATE was specified on RAISE_ERROR.
428C5	No data type mapping was found for a data type from the data source.
42901	A column function does not include a column name.
42903	A WHERE clause or SET clause includes an invalid reference, such as a column function.
42907	The string is too long.
42908	The statement does not include a required column list.

Table 25 (Page 6 of 6). Class Code 42: Syntax Error or Access Rule Violation

SQLSTATE Value	Meaning
42911	A decimal divide operation is invalid, because the result would have a negative scale.
42912	A column cannot be updated, because it is not identified in the UPDATE clause of the select-statement of the cursor.
42914	The DELETE is invalid, because a table referenced in a subquery can be affected by the operation.
42915	An invalid referential constraint has been detected.
42916	The alias cannot be created, because it would result in a repetitive chain of aliases.
42917	The object cannot be explicitly dropped.
42918	A user-defined data type cannot be created with a system-defined data type name (for example, INTEGER).
42921	Containers cannot be added to the tablespace.
42925	Recursive named derived tables cannot specify SELECT DISTINCT and must specify UNION ALL.
42932	The program preparation assumptions are incorrect.
42939	The object cannot be created, because the specified identifier is reserved for system use.
42962	A long column or a LOB column cannot be used in an index, a key, or a constraint.
42968	The connection failed, because there is no current software license.
42987	The statement is not allowed in a procedure or trigger.
42991	The BOOLEAN data type is currently only supported internally.
42993	The column, as defined, is too large to be logged.
42994	Raw device containers are not currently supported on this system.
429A9	SQL statement cannot be processed by DataJoiner.

Table 26. Class Code 44: WITH CHECK OPTION Violation

SQLSTATE Value	Meaning
44000	The INSERT or UPDATE is not allowed, because a resulting row does not satisfy the view definition.

Table 27 (Page 1 of 2). Class Code 51: Invalid Application State

SQLSTATE Value	Meaning
51002	The package corresponding to an SQL statement execution request was not found.

Table 27 (Page 2 of 2). Class Code 51: Invalid Application State

SQLSTATE Value	Meaning
51003	Consistency tokens do not match.
51004	An address in the SQLDA is invalid.
51008	The release number of the precompiled program is not valid.
51017	The user is not logged on.
51021	SQL statements cannot be executed until the application process executes a rollback operation.
51022	A CONNECT that specifies an authorization name is invalid when a connection (either current or dormant) already exists to the server named in that CONNECT statement.
51023	The database is already in use by another instance of the database manager.
51024	A view cannot be used, because it has been marked inoperative.
51025	An application in the XA transaction processing environment is not bound with SYNCPOINT TWOPHASE.
51026	An event monitor cannot be turned on, because its target path is already in use by another event monitor.
51027	The IMMEDIATE CHECKED option of the SET CONSTRAINTS statement is not valid since a table is not in the check-pending state.
51028	A package cannot be used, because it is marked inoperative.

Table 28 (Page 1 of 2). Class Code 54: SQL or Product Limit Exceeded

SQLSTATE Value	Meaning
54001	The statement is too long or too complex.
54002	A string constant is too long.
54004	The statement has too many table names or too many items in a SELECT or INSERT list.
54006	The result of concatenation is too long.
54008	The key is too long, or has too many columns.
54010	The record length of the table is too long.
54011	Too many columns were specified for a table or view.
54023	The limit for the number of parameters or arguments for a function or a procedure has been exceeded.
54028	The maximum number of concurrent LOB handles has been reached.
54029	The maximum number of open directory scans has been reached.
54030	The maximum number of event monitors are already active.
54031	The maximum number of files have already been assigned the event monitor.
54032	The maximum size of a table has been reached.

Table 28 (Page 2 of 2). Class Code 54: SQL or Product Limit Exceeded

SQLSTATE Value	Meaning
54033	The maximum number of partitioning maps has been reached.
54034	The combined length of all container names for the tablespace is too long.
54035	Tablespace limit exceeded.
54036	The path name for the container is too long.
54037	The container map for the tablespace is too complicated.
54038	Maximum depth of cascaded triggers exceeed.

Table 29. Class Code 55: Object Not in Prerequisite State

SQLSTATE Value	Meaning
55001	The database must be migrated.
55002	The explanation table is not defined properly.
55006	The object cannot be dropped, because it is currently in use by the same application process.
55007	The object cannot be altered, because it is currently in use by the same application process.
55009	The system attempted to write to a read-only file or a write-protected diskette.
55024	The tablespace cannot be dropped, because data related to a table is also in another tablespace.
55025	The database must be restarted.
55026	A temporary tablespace cannot be dropped.
55031	The format of the error mapping file is incorrect.
55032	The CONNECT statement is invalid, because the database manager was stopped after this application was started.
55033	An event monitor cannot be activated in the same unit of work in which it is created or modified.
55034	An active event monitor cannot be dropped.
55036	The node cannot be dropped, because it has not been removed from the partitioning map.
55037	The partitioning key cannot be dropped, because the table is in a multi-node nodegroup.
55038	The nodegroup cannot be used, because it is being rebalanced.
55039	The access or state transition is not allowed, because the tablespace is not in an appropriate state.
55041	Containers cannot be added to a tablespace while a rebalance is in progress.

Table 30. Class Code 56: Miscellaneous SQL or Product Error

SQLSTATE Value	Meaning
56031	The clause or scalar function is invalid, because mixed and DBCS data are not supported on this system.
56084	LOB data is not supported in DRDA.
56091	Multiple errors occurred as a result of executing a compound SQL statement.
56092	The type of authorization cannot be determined, because the authorization name is both a user id and group id.
56097	LONG VARCHAR and LONG VARGRAPHIC fields are not permitted in TABLESPACEs which are built on DEVICEs.
560A0	Action on a LOB value failed.

Table 31. Class Code 57: Resource Not Available or Operator Intervention

SQLSTATE Value	Meaning
57007	The object cannot be used, because a DROP or ALTER is pending.
57009	Virtual storage or database resource is temporarily unavailable.
57011	Virtual storage or database resource is not available.
57012	A non-database resource is not available. This will not affect the successful execution of subsequent statements.
57013	A non-database resource is not available. This will affect the successful execution of subsequent statements.
57014	Processing was canceled as requested.
57016	The table cannot be accessed, because it is inactive.
57017	Character conversion is not defined.
57019	The statement was not successful, because of a problem with a resource.
57020	The drive containing the database is locked.
57021	The diskette drive door is open.
57030	Connection to application server would exceed the installation-defined limit.
57032	The maximum number of concurrent databases have already been started.
57033	Deadlock or timeout occurred without automatic rollback.
57036	The transaction log does not belong to the current database.
57046	A new transaction cannot start because the database or instance is quiesced.
57047	An internal database file cannot be created, because the directory is not accessible.
57048	An error occurred while accessing a container for a tablespace.
57049	The operating system process limit has been reached.

Table 32. Class Code 58: System Error

SQLSTATE Value	Meaning
58004	A system error (that does not necessarily preclude the successful execution of subsequent SQL statements) occurred.
58005	A system error (that prevents the successful execution of subsequent SQL statements) occurred.
58008	Execution failed due to a distribution protocol error that will not affect the successful execution of subsequent DDM commands or SQL statements.
58009	Execution failed due to a distribution protocol error that caused deallocation of the conversation.
58010	Execution failed due to a distribution protocol error that will affect the successful execution of subsequent DDM commands or SQL statements.
58011	The DDM command is invalid while the bind process in progress.
58012	The bind process with the specified package name and consistency token is not active.
58014	The DDM command is not supported.
58015	The DDM object is not supported.
58016	The DDM parameter is not supported.
58017	The DDM parameter value is not supported.
58018	The DDM reply message is not supported.
58030	An I/O error has occurred.
58031	The connection was unsuccessful, because of a system error.
58032	Unable to use the process for a fenced mode user-defined function.
58034	An error was detected while attempting to find pages for an object in a DMS tablespace.
58035	An error was detected while attempting to free pages for an object in a DMS tablespace.
58036	The internal tablespace ID specified does not exist.

Section 3.   Classic Connect Messages

Chapter 10. MVS Console Messages

The following section contains MVS write to operator (WTO) messages.

DJX00001I   START CIB NOT FREED

Cause: MSCAOPER was unable to free the CIB used to start the kernel.

Action: Operations continue. No action is required to continue processing.

DJX00002I   MODIFY COMMAND RECEIVED

Cause: MSCAOPER acknowledged receipt of a Modify (F) command. This is an informational message only.

Action: Normal processing continues.

DJX00003I   INTERFACE ENABLED

Cause: This is an informational message issued to indicate that the operator interface is now active.

Action: No action is required.

DJX00020C MEMORY EXHAUSTED

Cause: The CICS APIF extension was unable to acquire storage for its save area. The request for Classic Connect bridge services is abended.

Action: Check total storage allocation and usage in the affected CICS region.

DJX00030C MEMORY EXHAUSTED

Cause: The Classic Connect CICS dispatcher was unable to acquire storage for its save area. The request is ended with a return code equal to the GETMAIN return code.

Action: Check total storage allocation and usage in the requesting CICS region.

DJX00040C ERR: VT LOAD ERROR

Cause: The Classic Connect Virtual host system (VHS) table is not loaded or the area is corrupt.

Action: The requesting task is not terminated.

DJX00041I   <taskname> ENDED

Cause: The Classic Connect system ended the named task. This is an informational message.

Action: No action is required. Processing continues.

DJX00050C ERROR:UNSUPPORTED OPERATING SYSTEM

Cause: System initialization failed. The communication vector table (CVT) does not indicate that the operating system is at least the MVX/XA level.

Action: Verify the system is IBM MVS/XA or above. Verify contents of the communication vector table data control block (CVTDCB) on a running version of the operating system.

DJX00051C LOAD DD STATEMENT MISSING

Cause: The DD statement for LOAD is missing. System initialization terminates.

Action: Correct the JCL for the Classic Connect address space by including a LOAD DD statement for the Classic Connect unauthorized load library.

DJX00052W INVALID TASKPARM - DEFAULT USED

Cause: The execution time parameters for the Classic Connect kernel are invalid.

Action: Correct the value of the PARM configuration parameter used when starting the Classic Connect kernel.

DJX00053I   INIT SUCCESSFUL

Cause: Classic Connect initialization was successful. This is an informational message during initialization.

Action: No action is required.

DJX000541 INITIALIZATION COMPLETE

Cause: The full initialization of the Classic Connect environment completed. This is an informational message.

Action: No action is required.

DJX00055I TERMINATION STARTED

Cause: Termination of the Classic Connect address space began. This is an informational message.

Action: No action is required.

DJX00056I TERMINATION COMPLETE

Cause: Termination of the Classic Connect address space is complete. This is an informational message.

Action: No action is required.

DJX00057W ERR *XXXXXXXX* ',MF=L Error return code from SRC or member name of module being loaded (bldl).

Cause: Either an invoked function returned a nonzero return code, or a required Classic Connect module could not be loaded. The address space will be terminated.

Action: Check the return code to identify the error. The return code displays on the console and is logged in the subsystem genlog file. Many (not all) of the return codes are listed in Chapter 28, "MVS System Error Message" on page 407. Take corrective action noted, and restart the address space.

Alternatively, verify that the requested module is available in the LOAD data set. Check for other link/load/xctl system messages.

DJX00060W XMIN SSB FREEMAIN FAILED

Cause: The freemain for the SSB failed during subsystem initialization. CSA-based storage might have been lost until the next IPL.

Action: Report this message to an authorized IBM service representative. Initialization of the subsystem continues.

DJX00070C LENGTH OF DATA TO BE CONVERTED IS INVALID

Cause: Data passed to data comparison routine was of an invalid length.

Action: Correct the problem in the invoking routine and recompile.

DJX00080I ACTIVE TASK TABLE RECYCLED

Cause: The active task table was purged of previously killed entries. This is an informational message.

Action: No action is required.

DJX00081W TASK TABLE IS FULL

Cause: The Maximum number of tasks was started by the Classic Connect address space.

Action: Increase the value of the MAX NUMBER OF TASKS kernel configuration parameter.

DJX00082I PORT INIT ERROR

Cause: A port for a remote request could not be initialized. An error return code is passed to the remote requester.

Action: Check for other error messages on local and remote systems.

DJX00090I ATT RECYCLED

Cause: The active task table was purged of terminated entries while attaching a new task. This is an informational message.

Action: No action is required.

DJX00100I *XXXXXXXX* ENDED ATTTNAME (task name)

Cause: The named external task ended and was removed from the active task table. This is an informational message.

Action: Normal processing continues.

DJX00110W BADGTID

Cause: The generalized task identifier (GTID) for the task being terminated did not match. The task is not removed from the active task table, and an error code is returned to the requesting task.

Action:

DJX00111W WRN: SHUT MSG PORT FAILED

Cause: During task termination the port for an external client could not be closed.

Action: Normal processing continues. A nonzero return code is set.

DJX00112W WRN: SEMAERR

Cause: During task termination all semaphores did not complete cleanup.

Action: Normal processing continues, and a nonzero return code is sent to the requesting task.

DJX00120I ACTIVE TASK TABLE RECYCLED

Cause: During the start of an external task, the active task table was purged of killed entries. This is an informational message.

Action: No action is required.

DJX00130W INVALID XM REQUEST

Cause: An invalid request was posted to the Monitor task. The request is ignored, and the Monitor returns to a wait state.

Action: No action is required.

DJX00131W ADD ERR

Cause: An error occurred while building an active task table for a new task. The new task is not created, and an error return code is returned.

Action: Check for other error messages indicating the specific error condition.

DJX00132W and 133W DEL ERR ROUTER SHUTDOWN FAILED

Cause: During deletion of the active task table entry (ATTE) for a task, an error occurred. Processing continues, and a nonzero return code is returned.

Action: Check for other error messages that indicate the error condition.

DJX00140W ROUTER NOTIFY FAILED

Cause: During task end processing the notification of the router task failed. The task is terminated without router notification.

Action: No action is required.

DJX00141W SHUT MSG PORT FAILED

Cause: During task end processing the closing of a message port failed. Processing continues.

Action: None

DJX00142W SEMA4ERR

Cause: During task end processing an error was encountered in semaphore cleanup. This is a warning message. Processing continues.

Action: None

DJX00143W VT LOAD ERROR

Cause: Task end processing could not locate a valid virtual host system vector table (VHSVT). Processing terminates with an error return code.

Action: None

DJX00144I *XXXXXXXX* ENDED atttname - task name

Cause: Task end processing finished for the indicated task. This is an informational message.

Action: No action is required.

DJX00150W UNKNOWN CPIC ERROR

Cause: APPC/VTAM API processing received an unknown CPIC error code. The CPIC return code is passed to the caller.

Action:

DJX00160I DIAG1 - *xxxxxxxxxx*

Cause: APPC TLI diagnostic information from mstmablk (LU 6.2 CI).

Action: Report the error to an authorized IBM service representative.

DJX00161I DIAG2 - *xxxxxxxxxx*

Cause: APPC TLI diagnostic information from mstmablk (LU 6.2 CI).

Action: Report the error to an authorized IBM service representative.

DJX00180I UNSOLICITED MESSAGE RECEIVED FROM TCPIP

Cause: An unsolicited message was received from TCP/IP. The message is ignored, and normal processing continues

Action: No action is required.

DJX00190C UNSUPPORTED OPERATING SYSTEM

Cause: Classic Connect is being executed on an unsupported operating system. The minimum system requirement is IBM MVS/XA.

Action: Check the level of the operating system. Also check the contents of the communication vector table data control block (CVTDCB) on the running system.

DJX00191W ERROR DURING INIT

Cause: An error occurred during initialization of a client service. Processing continues with an error return code.

Action: Check for other associated error messages.

DJX00210C SERVER: UNSUPPORTED OPERATING SYSTEM

Cause: Classic Connect is being executed on an unsupported operating system. The minimum system requirement is IBM MVS/XA.

Action: Check the level of the operating system. Also check the contents of the communication vector table data control block (CVTDCB) on the running system.

DJX00211C SERVER ERROR DURING INIT

Cause: An error occurred during initialization of a server function.

Action: Check for other associated error messages. An error return code is returned to the service requester.

DJX00301I   ATT RECYCLED

Cause: The ATT has been scanned and the first available ATT Entry was used.

Action: No user action is required.

DJX00302E NO INPUT JOB DEFINITION

Cause: An internal error has occurred while attempting to start a BATCH DMSI.

Action: Report the error to an authorized IBM service representative.

DJX00303E JOBRDR DCB FAILED TO OPEN

Cause: The JOBRDR DD could not be opened.

Action: Confirm that the DD statement is present and properly coded as an INTERNAL reader.

DJX00304E JOBDEF DCB FAILED TO OPEN FOR LOADING

Cause: The JOBDEF DD could not be opened to process a batch job request.

Action: Confirm that the DD statement is present and that the dataset referenced is a valid job definition dataset.

DJX00305W PARAMETER FIELD CANNOT EXCEED 100 CHARACTERS IN LENGTH

Cause: A parameter field in excess of 100 characters was received for processing.

Action: Report the error to an authorized IBM service representative.

DJX00306W NULL ENTRY ENCOUNTERED IN PARAMETER FIELD

Cause: A empty entry was encountered in the input parameter field.

Action: Report the error to an authorized IBM service representative.

DJX00307E NAME LONGER THAN 8 CHARACTERS

Cause: A batch job name of more than eight characters was encountered.

Action: Report the error to an authorized IBM service representative.

DJX00308E THE JOBDEF BEGINNING jjjjjjjj WILL BE IGNORED

Cause: The batch job definition (name) beginning with jjjjjjjj is invalid and will be ignored.

Action: Check your TASK INFO ENTRY parameters for this definition.

DJX00309E BLDL FAILURE -C- AGAINST JOBDEF

Cause: BLDL processing encountered an error while attempting to locate the definition for a batch job.

Action: Check the JOBDEF dataset for errors.

DJX00310E BLDL FAILURE -D- AGAINST JOBDEF

Cause: BLDL processing encountered an error while attempting to locate the definition for a batch job.

Action: Check the JOBDEF dataset for errors.

DJX00311E Job jjjjjjjj NOT FOUND BY BLDL

Cause: The requested job (jjjjjjjj) was not found in the JOBDEF dataset.

Action: Add an entry in the JOBDEF dataset for jjjjjjjj or change the TASK INFO ENTRY parameter that references job jjjjjjjj (to use another jobname).

DJX00800I   ACTIVE TASK DISPLAY iiiiiiii nnnnnnnn aaaaaaaa p s ffffffff uuuuuuuu dddddddd

Cause:

This message displays the following information for each entry in the active task table (ATT).

iiiiiiii

taskID; kernel ID for task

nnnnnnnn

taskname; name of task

aaaaaaaa

ASID; ASCB address for task

pppp

entry type

T   A TSO session

D   A local DMSI

R   A remote client

S   A SI attached service

ssss

status of entry

A   entry is available

I   entry is in use by a session

P   'private' entry for kernel usage

R   the entry is temporarily reserved (changing states)

ffffffff

FLAGS; ATT flag bytes

00xxxxxx ATT entry available

x1xxxxxx ATT in use

x2xxxxxx ATT private entry

x4xxxxxx ATT temporarily reserved

xx1xxxxx Task is in KILL status

xx2xxxxx Subtask terminated (ETX received control)

xx8xxxxx External task ended

xxxx8xxx External task ended

xxxxx1xx No RACF profile available

......xx Reserved

uuuuuuuu

userID attached to the task

dddddddd

ASID of user.

Action: No action is required.

DJX00803I   KERNEL NOW ACTIVE

Cause: This message confirms the completion of the ACTIVATE command.

Action: No action is required.

DJX00804I   INVALID COMMAND *ccccccccccc*

Cause: This is a warning message. The command entered for the kernel is invalid.

Action: Correct the command syntax and reenter.

DJX00804I   KERNEL QUIESCING

Cause: This message confirms the start of the QUIESCE command.

Action: No action is required.

DJX00805I   KERNEL IS SHUTTING DOWN

Cause: This message confirms the start of the SHUTDOWN command.

Action: No action is required.

DJX00806I   KERNEL IS IN *xxxxx* **STATE**

Cause: This message displays the current status of the kernel after issuing the STATUS command. The status can be:

ACTIVE

The kernel is accepting and processing service requests.

SUSPEND

The kernel is no longer accepting new service requests.

QUIESCE

The kernel is quiesced and is no longer processing service requests.

Action: No action is required.

DJX00807I   KERNEL STOP PROCESSING INITIATED

Cause: This message confirms the start of the STOP command.

Action: No action is required.

DJX00808I KERNEL NOW IN SUSPEND STATUS

Cause: This message confirms the completion of the SUSPEND command.

Action: No action is required.

DJX00810W TASK NOT FOUND *nnnnnnnn ,iiiiiii*

Cause: The task name (*task name,taskid*) specified in the operator command was not found in the active task table (ATT).

Action: Check the task name (task name,taskid) entered.

DJX00811   TASK NAME NOT UNIQUE *nnnnnnnn*

Cause: The task name entered in the operator command is not unique.

Action: Check that the correct task name was entered, or enter both the task name and the task ID to uniquely identify the task.

DJX00830I task.name,taskid TERMINATED

Cause: This message confirms the completion of the TERMINATE command.

Action: No action is required.

DJX00840I task.name,taskid BEING CANCELLED

Cause: This message confirms the issuance of the CANCEL command.

Action: No action is required.

DJX00900C NO MEMORY

Cause: During End-Of-Memory (EOM) processing, the subsystem function was unable to acquire storage for its save areas.

Action: No action is taken on behalf of the terminating address space.

DJX00902I   SUBSYSTEM CLEANUP COMPLETE

Cause: During End-Of-Memory (EOM) processing, the subsystem cleanup completed. This is an informational message.

Action: No action is required.

DJX00902I   SUBSYSTEM NOT ACTIVE

Cause: During subsystem End-Of-Memory (EOM) processing, the subsystem options block (SSOB) indicated the subsystem was not active.

Action: No further actions are taken for the terminating memory.

DJX00903I   SUBSYSTEM NOT READY

Cause: During subsystem End-Of-Memory (EOM) processing, the subsystems options block (SSOB) indicated the subsystem was not in an ACTIVE state.

Action: No further actions are taken for the terminating memory.

DJX00904W PC FAILED

Cause: During subsystem End-Of-Memory (EOM) processing, the program call (PC) to the Classic Connect address space returned a nonzero return code. EOM processing completes.

Action: No action is required.

DJX00909I   DUMP COMPLETE

Cause: This message confirms the completion of the DUMP command.

Action: No action is required.

DJX00910C NO MEMORY

Cause: During subsystem end of task (EOT) processing, memory was not available for required save areas.

Action: No action is required. No further processing takes place for the task in EOT processing.

DJX00911I   SUBSYSTEM CLEANUP COMPLETE

Cause: During subsystem end of task (EOT) processing, subsystem cleanup completed. This is an informational message.

Action: No action is required.

DJX00912I   SUBSYSTEM NOT ACTIVE

Cause: During subsystem end of task (EOT) processing, the Classic Connect subsystem was found to be in an inactive state.

Action: No action is required. No processing occurs for the task undergoing EOT.

DJX00913I   SUBSYSTEM CLEANUP COMPLETE

Cause: During subsystem end of task (EOT) processing, cleanup for the terminating task completed. This is an informational message.

Action: No action is required.

DJX00914W PC FAILED

Cause: During subsystem end of task (EOT) processing, the program call (PC) to the Classic Connect address space returned a nonzero return code. This is an informational message.

Action: No action is required.

DJX00920I   ERROR RECOVERY ENTERED

Cause: Subsystem initialization encountered an error, and the error recovery routine was successfully entered. This is an informational message.

Action: No action is required.

DJX00921C VT ACCESS FAILED

Cause: During subsystem init error recovery, the virtual host system vector table (VHSVT) was not available. Further cleanup of virtual host system table (VHST) anchored storage is not possible.

Action: Investigate why the VHST was not available or was damaged.

DJX00922C SSB ACCESS FAILED

Cause: During subsystem init error recovery, the SSB was not available or was damaged. No cleanup of SSB anchored fields is possible, nor can the DMSI ACEE storage (sp241) be attempted.

Action: Investigate why the SSB was not available or damaged.

DJX00930I   ERROR RECOVERY ENTERED

Cause: The Classic Connect server error recovery was successfully entered. This is an informational message.

Action: No action is required.

DJX00931C VT ACCESS FAILED

Cause: During error recovery the virtual host system vector table (VHSVT) was not accessible or was damaged. No further recovery is attempted.

Action: Investigate the absence or damage of the VHSVT.

DJX00932C INTERNAL ERROR

Cause: During error recovery an error in the contents or structure of the ASCB was encountered. No further recovery is attempted.

Action: This is a severe error. Determine the damage to the ASCB structure.

DJX00933C NOTIFY KERNEL FAILED

Cause: During error recovery the program call (PC) to the Classic Connectaddress space returned a return code of 4 or greater.

Action: This is a severe error. Determine the problem with the Classic Connect address space.

DJX00934C KERNEL NOT ACTIVE

Cause: During error recovery, the SSB of the Classic Connectaddress space could not be found. No Classic Connect cleanup is possible.

Action: Determine why the Classic Connect address space is not in the SST.

DJX009441I ERROR RECOVERY ENTERED

Cause: Error recovery for the Classic Connect monitor was successfully entered. This is an informational message.

Action: No action is required.

DJX009441C   DJX009441C

DJX009441C VT ACCESS FAILED

Cause: During Classic Connect monitor error recovery,
the virtual host system vector table (VHSVT) was not
accessible or was damaged. No cleanup of the Classic
Connect-based information is possible.

Action: Determine why the VHSVT was not accessible
or damaged.

Chapter 11. General Messages

The following messages contain general information or general errors.

DJX0 (X'00000000') This indicates an informational message.

Cause: An informational message is followed by a Classic Connect message. A Classic Connect message is always displayed in US English.

Action: No action is required.

DJX1 (X'00000001') This indicates the start of a function.

Cause: A Classic Connect function started.

Action: No action is required.

Chapter 12. Configuration Error Messages

The following messages are related to configuration errors.

DJX65537 (X'00010001') Feature not implemented.

Cause: The function is not supported, or there is a configuration or syntax error in an SQL statement.

Action: Check for and correct configuration or syntax errors in SQL statements. If this does not correct the problem, contact an authorized IBM service representative to see if the function is supported.

DJX65538 (X'00010002') KEYWORD is not found in the Classic Connect configuration file.

Cause: KEYWORD is not found in the Classic Connect configuration file.

Action: Verify the Classic Connect configuration file for all keywords and values.

DJX65539 (X'00010003') The Classic Connect configuration file is too large.

Cause: The number of lines in the configuration file exceeds 1000.

Action: Verify that the Classic Connect configuration file contains fewer than 1000 lines.

DJX65540 (X'00010004') There is no such NODE ID in the Node Address Table.

Cause: A client (or another kernel) is attempting to access the Classic Connect. The Node ID supplied by the client or kernel is not defined in this kernel's configuration file.

Action: Contact an authorized IBM service representative.

DJX65541 (X'00010005') There is no such task in the Task Information Table.

Cause: A client is attempting to access a data source that is configured improperly. The SYSLOCATION parameter is referencing a nonexistent TASK INFO ENTRY service name.

Action: Contact an authorized IBM service representative.

DJX65542 (X'00010006') Unable to initialize the session table because the requested size is 0.

Cause: The value of the SESSION TABLE SIZE keyword in the configuration file is 0.

Action: Verify that the value of SESSION TABLE SIZE is not 0 in the Classic Connect configuration file.

DJX65543 (X'00010007') There is no such half session ID in the session table.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX65544 (X'00010008') An invalid pointer was detected.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX65545 (X'00010009') An invalid field was found in the Classic Connect table.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX65546 (X'0001000A') There are no free elements in the global session table.

Cause: All entries in the session table are in use. The number of entries is established at the initialization of the Classic Connectsubsystem by the configuration parameter SESSION TABLE SIZE. Because the number of active sessions is dynamic depending on subsystem activity, this condition might be temporary.

Action: Restarting the application at another time might solve the problem. Otherwise, an authorized IBM service representative should review the configuration parameters.

DJX65547 (X'0001000B') There is no such location in the syslocations table.

Cause: The client is attempting to access an improperly configured data source.

Action: Contact an authorized IBM service representative.

DJX65548 (X'0001000C') Unexpected error occurred while reading a configuration file.

Cause: An unexpected I/O error was received while reading the configuration file. The file format is invalid.

Action: Verify that the configuration file was not corrupted.

DJX65549 (X'0001000D') Unexpected EOF found while reading a configuration file.

Cause: An unexpected EOF error was received while reading the configuration file. The file format is invalid, which might be caused by null values in the configuration file.

Action: Ensure that the configuration file was closed properly. Then restart.

DJX65550 (X'0001000E') Unable to open the specified configuration file.

Cause: Depending upon how the Classic Connect configuration file is defined in the JCL, either there is no DD statement with the name VHSCONF pointing to the configuration file, or the VHSCONF DD statement points to a nonexistent member of a partitioned data set.

Action: Check DD statements.

DJX65551 (X'0001000F') The CTABLE cell is not initialized properly for the IVCnfg function.

Cause: This might be the result of improper platform initialization or the random destruction of memory. This is probably an internal error.

Action: Contact an authorized IBM service representative.

DJX65552 (X'00010010') The NODE ADDRESS ENTRY parameter provided contains an invalid Node ID value and will be ignored.

Cause: This is a configuration error. A nonnumeric Node ID or a Node ID greater than 32767 was supplied.

Action: Correct the configuration file.

DJX65553 (X'00010011') Multiple NODE ADDRESS ENTRY parameters were given using Node ID xxxxx.

Cause: Only the first parameter will be used. The ignored parameter text is xxxxx.

Action: Correct the configuration file so each Node ID is unique. This might also require reconfiguration of the duplicated kernel's Node ID.

DJX65554 (X'00010012') A NODE ADDRESS ENTRY parameter was given without a CPI service name field and will be ignored.

Cause: The NODE ADDRESS ENTRY parameter format is invalid. The ignored parameter text is xxxxxxxx.

Action: Correct the configuration file. Refer to the configuration documentation for the NODE ADDRESS ENTRY parameter.

DJX65555 (X'00010013') A NODE ADDRESS ENTRY parameter was given with bad or extraneous characters in the NODE ID field.

Cause: The NODE ID parameter of the NODE ADDRESS ENTRY parameter is invalid. The node ID must be numeric and in the range of 1 to 32767.

Action: Correct the NODE ADDRESS ENTRY parameter in the configuration file.

DJX65556 (X'00010014') A NODE ADDRESS ENTRY was given with a CPI service name that contains xxxxx characters.

Cause: The service name in the NODE ADDRESS ENTRY parameter is too long. The service name must be one to eight characters in length. However, for AIX, the service name is not constrained to eight characters.

Action: Correct the NODE ADDRESS ENTRY parameter in the configuration file.

DJX65557 (X'00010015') DJX65565 (X'0001001D')

DJX65557 (X'00010015') A NODE ADDRESS ENTRY was given without a colon (:) and will be ignored.

Cause: The communications interface section of the NODE ADDRESS ENTRY parameter is invalid. This information is delimited with a colon (:).

Action: Correct the NODE ADDRESS ENTRY parameter in the configuration file.

DJX65558 (X'00010016') A NODE ADDRESS ENTRY was given with bad or extraneous characters between the CPI service name and the colon (:).

Cause: The NODE ADDRESS ENTRY parameter is invalid. Extra characters were found between the service name and the colon.

Action: Correct the NODE ADDRESS ENTRY parameter in the configuration file by removing the extraneous characters.

DJX65559 (X'00010017') Too many SYSLOCATION parameters were given. Only the first *xxxxx* entries can be used.

Cause: The ignored parameter text is *xxxxx*.

Action: Check the configuration file and locate the extra SYSLOCATION parameters. Delete unnecessary parameters.

DJX65560 (X'00010018') A SYSLOCATION was given with a location name field that contains *xxxxx* characters.

Cause: The location name in the SYSLOCATION parameter is too long. The location name must be one to six characters in length.

Action: Correct the SYSLOCATION parameter in the configuration file.

DJX65561 (X'00010019') Multiple SYSLOCATION parameters were given using the same location name.

Cause: The SYSLOCATION parameter is invalid. The location name is duplicated.

Action: Correct the configuration file by removing duplicate location names.

DJX65562 (X'0001001A') A SYSLOCATION parameter was given with only a location name field and will be ignored.

Cause: The SYSLOCATION parameter is invalid. The node ID and service name must follow the location name.

Action: Correct the SYSLOCATION parameter in the configuration file.

DJX65563 (X'0001001B') A SYSLOCATION parameter was given with bad or extraneous characters in the NODE ID field.

Cause: The node ID in the SYSLOCATION parameter is invalid. The node ID must follow the location name and be a numeric value between 1 and 32,767.

Action: Correct the SYSLOCATION parameter in the configuration file.

DJX65564 (X'0001001C') A SYSLOCATION parameter was given without a service name field and will be ignored.

Cause: The service name is missing from the SYSLOCATION parameter. The service name must follow the node ID.

Action: Correct the SYSLOCATION parameter in the configuration file.

DJX65565 (X'0001001D') A SYSLOCATION was given with a service name field that contains *xxxxx* characters.

Cause: The service name is invalid. The service name must be one to eight characters in length and follow the node ID. However, for AIX, the service name is not constrained to eight characters.

Action: Correct the SYSLOCATION parameter in the configuration file.

Chapter 13. File Management Error Messages

The following messages are related to file management errors.

DJX131072 (X'00020000') Feature not implemented.

Cause: The function is not supported, or there is a configuration or syntax error in an SQL statement.

Action: Check for and correct configuration or syntax errors in SQL statements. If this does not correct the problem, contact an authorized IBM service representative to see if the function is supported.

DJX131073 (X'00020001') Unable to close file.

Cause: Unable to close file.

Action: No action is required.

DJX131074 (X'00020002') End of file reached.

Cause: The end of the file was reached during file access.

Action: No action is required.

DJX131078 (X'00020006') Attempt to position file pointer outside of file.

Cause: The catalog file was corrupted.

Action: Contact an authorized IBM service representative.

DJX131079 (X'00020007') Unable to open a file.

Cause: A file cannot be opened or does not exist.

Action: Verify that the NL US ENGLISH CAT keyword is in the Classic Connectconfiguration file and that the correct language catalog is present.

DJX131080 (X'00020008') Error reading a file.

Cause: Unable to read from the file.

Action: Verify that the NL US ENGLISH CAT keyword is in the Classic Connectconfiguration file and that the correct language catalog is present and not empty.

DJX131081 (X'00020009') Error writing to a file.

Cause: There was an error writing to the Classic Connect log file.

Action: Verify that the log file is present and that it has write permission.

DJX131082 (X'0002000A') Received an invalid file origin parameter.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX131083 (X'0002000B') Cannot determine the current file position.

Cause: The catalog file was corrupted.

Action: Contact an authorized IBM service representative.

Chapter 14. User ID and Password Error Messages

The following messages are related to user ID and password errors.

DJX196608 (X'00030000') Feature not implemented.

Cause: The function is not supported, or there is a configuration or syntax error in an SQL statement.

Action: Check for and correct configuration or syntax errors in SQL statements. If this does not correct the problem, contact an authorized IBM service representative to see if the function is supported.

DJX196609 (X'00030001') The user ID string is empty.

Cause: The user ID was not supplied.

Action: Verify that USERID contains the correct user ID in the Classic Connect configuration file, or verify that the user ID is supplied in the connect statement.

DJX196610 (X'00030002') Cannot change the authority of an existing task on this platform.

Cause: The DMSI is attempting to run in guarded mode. However, this mode is not supported for the DMSI platform.

Action: Contact an authorized IBM service representative.

DJX196623 (X'0003000F') The transport layer did not provide an error number to CI.

Cause: The communications layer returned an unknown error code to a Classic Connect component. The log files might contain the error code returned. This situation usually indicates a configuration error on either the client or the server.

Action: Verify that the local configuration is correct, and then contact an authorized IBM service representative.

Chapter 15. Message Catalog Error Messages

The following messages are related to memory management errors.

DJX262144 (X'00040000') Feature not yet implemented.

Cause: The function is not supported, or there is a configuration or syntax error in an SQL statement.

Action: Check for and correct configuration or syntax errors in SQL statements. If this does not correct the problem, contact an authorized IBM service representative to see if the function is supported.

DJX262145 (X'00040001') Memory is not available.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX262146 (X'00040002') Unable to write to the protected buffer.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

Chapter 16. Resource Management Error Messages

The following messages are related to resource management errors.

DJX327681 (X'00050001') National language ID is invalid.

Cause: The current national language ID is not a valid national language ID.

Action: Verify that the correct value for national language ID is in the Classic Connect configuration file.

DJX327682 (X'00050002') Received an invalid resource set number.

Cause: The resource set number specified is not in the language catalog file. This is probably an internal error.

Action: Contact an authorized IBM service representative.

DJX327683 (X'00050003') Received an invalid resource message number.

Cause: The resource message number specified is not in the language catalog file. This is probably an internal error.

Action: Contact an authorized IBM service representative.

DJX327684 (X'00050004') The national language ID received is not currently supported.

Cause: The only national language ID supported is US English. This is probably an internal error.

Action: Contact an authorized IBM service representative.

DJX327685 (X'00050005') The CTABLE is not initialized and cannot obtain the language catalog name.

Cause: The CTABLE is not yet initialized and cannot obtain the language catalog name or the message text. This might be the result of improper platform initialization or the random destruction of memory. This is probably an internal error.

Action: Contact an authorized IBM service representative.

DJX327686 (X'00050006') Unable to get the requested memory for the semaphore handle.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX327687 (X'00050007') The IVResourceInit function is not called yet.

Cause: This might be the result of improper platform initialization or the random destruction of memory. This is probably an internal error.

Action: Contact an authorized IBM service representative.

DJX327688 (X'00050008') The CTABLE is not initialized properly for this task or function.

Cause: This might be the result of improper platform initialization or the random destruction of memory. This is probably an internal error.

Action: Contact an authorized IBM service representative.

DJX327689 (X'00050009') No NL value was found in the configuration file. The default national language of US ENGLISH will be used.

Cause: No NL value was found in the configuration file. The default national language of US ENGLISH will be used.

Action: No action is required.

DJX327690 (X'0005000A') Bad or extraneous characters were encountered while reading the RESOURCE VERSION value from the configuration file.

Cause: The use of RESOURCE VERSION in the configuration files is now obsolete. Replace the RESOURCE VERSION parameter with the NL keyword.

Action: Contact an authorized IBM service representative. Replace the RESOURCE VERSION parameter with the NL keyword. For example, the usage is NL = US English.

DJX327691 (X'0005000B') DJX327694 (X'0005000E')

DJX327691 (X'0005000B') The text provided for a national language configuration parameter is not currently supported.

Cause: Classic Connect supports only NL US English.

Action: Use NL US English supported text only.

DJX327692 (X'0005000C') No national language catalog file name parameter was given for a specified language.

Cause: This is a configuration error.

Action: Use the NL keyword, and specify US English.

DJX327693 (X'0005000D') An unsupported language value was passed to the LLRedCk function.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX327694 (X'0005000E') The use of RESOURCE VERSION in the configuration file is now obsolete.

Cause: Replace this parameter with the NL keyword.

Action: Replace the RESOURCE VERSION parameter with the NL keyword. For example, usage is NL = US English.

Chapter 17. Semaphore Error Messages

The following messages are related to semaphore errors.

DJX393233 (X'00060011') The MAX NUMBER OF SEMAPHORES value of *xxxxx* given in the configuration file is below the minimum allowed value of *xxxxx*.

Cause: The minimum allowed value will be used.

Action: This is an informational message. The system will operate, but will use more resources than necessary. No action is required. However, you should tune the MAX NUMBER OF SEMAPHORES to your actual resource requirements.

DJX393234 (X'00060012') The MAX NUMBER OF SEMAPHORES value of *xxxxx* given in the configuration file exceeds the maximum allowed value of *xxxxx*.

Cause: The maximum allowed value will be used.

Action: This is an informational message. The system will operate, but will use more resources than necessary. No action is required. However, you should tune the MAX NUMBER OF SEMAPHORES to your actual resource requirements.

DJX393235 (X'00060013') Bad or extraneous characters were encountered while reading the MAX NUMBER OF SEMAPHORES parameter from the configuration file.

Cause: The MAX NUMBER OF SEMAPHORES value is invalid. The value should be between 200 and 32767.

Action: Correct the MAX NUMBER OF SEMAPHORES parameter in the configuration file.

DJX393236 (X'00060014') The MAX NUMBER OF SEMAPHORES parameter was not provided in the configuration file.

Cause: This is an informational message. The default number of semaphores (200) will be used.

Action: You might need to define the MAX NUMBER OF SEMAPHORES parameter if the number of resources exceeds the default number.

DJX393237 (X'00060015') The amount (*xxxxx* bytes) of memory needed to support the *xxxxx* semaphores could not be obtained.

Cause: The memory needed to support the semaphores could not be obtained.

Action: Reduce the MAX NUMBER OF SEMAPHORES parameter in the configuration file or provide more memory.

Chapter 18. Subtask Management Error Messages

The following messages are related to subtask management errors.

DJX458752 (X'00070000') Unable to find caller's global task ID.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX458754 (X'00070002') The process timed out.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX458755 (X'00070003') An invalid message was received.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX458757 (X'00070005') Monitor serialization failed.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

Chapter 19. API Error Messages

The following messages are related to API errors.

DJX524290 (X'00080002') Invalid message type was received.

Cause: A message was received by the API that had an unknown message type. The message was ignored.

Action: No action is required. This will be displayed in the Classic Connect log.

DJX524291 (X'00080003') Memory allocation error occurred, and session was rejected.

Cause: An attempt to allocate memory for control information failed. This occurred while trying to establish a connection with a DMSI. The current SQL statement being executed failed.

Action: Attempt to reexecute the SQL statement.

DJX524292 (X'00080004') Request for connection with DMSI was unsuccessful.

Cause: An attempt to connect with a DMSI was unsuccessful. The DMSI to which the connection was attempted might be local or remote. There might be an error in the configuration file used for the location to which the user was attempting to connect. This message can also occur if the location is remote and communication with the remote node failed, or when a user ID or password was not provided.

Action: When executing a test program requiring data from a remote node via the CI, and while the remote subsystem is up with an empty configuration file, the SQL code -524292 will go to the application program following a PREPARE statement. This code refers to the message "DMSI session was rejected" because the location was marked as not reachable at this point. Restart the application and try to use this location again making sure that the user configuration file is valid and that a proper user ID and password are provided. If this fails, contact an authorized IBM service representative.

DJX524293 (X'00080005') Cannot allocate an API session control block.

Cause: An attempt to allocate memory for control information failed. This occurred while trying to establish a connection with a DMSI. The current SQL statement being executed failed.

Action: Attempt to reexecute the SQL statement.

DJX524294 (X'00080006') Cannot find the session control block.

Cause: A control block error occurred. This means that API/DMSI session information was corrupted. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX524295 (X'00080007') Cannot extract the message data for processing.

Cause: An attempt to extract message data into local API buffers for processing failed because an attempt to allocate memory failed. The current SQL statement being executed failed.

Action: Attempt to reexecute the SQL statement.

DJX524296 (X'00080008') The location is unknown.

Cause: The location specified in the query could not be found in the SYSLOCATION table.

Action: Try another location, or contact an authorized IBM service representative.

DJX524297 (X'00080009') Unable to allocate a message buffer.

Cause: An attempt to allocate a message buffer to send data to or from API/DMSI failed.

Action: Attempt to reexecute the SQL statement.

DJX524298 (X'0008000A') The session with the DMSI is unusable.

Cause: The location specified cannot be used because the session is not usable at this time. This might be the result of an earlier attempt at access that contained an error that marked the session as unusable.

Action: Check the DMSI output for messages relating to the cause of the problem.

DJX524299 (X'0008000B') DMSI connection processing failed.

Cause: An attempt to process a connection between the API and DMSI failed in the API. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX524300 (X'0008000C') A message cannot be prepared for forwarding.

Cause: An error occurred while trying to prepare a message to send to a DMSI. This is usually due to memory allocation errors within the API.

Action: Attempt to reexecute the SQL statement.

DJX524301 (X'0008000D') During the query decomposition, a memory allocation error occurred.

Cause: While attempting to decompose the query for processing, a memory allocation error occurred within the API.

Action: The COMPRESSION setting must be consistent across all platforms. Check that the COMPRESSION parameter in the client configuration file matches the COMPRESSION parameter in the server configuration file. Then, attempt to reexecute the SQL statement.

DJX524302 (X'0008000E') The recombination of message responses failed.

Cause: While combining the responses from DMSIs into one message to return to the application, a control block state error occurred. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX524303 (X'0008000F') The API query control block state is invalid.

Cause: A control block in the API reached an invalid state. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX524304 (X'00080010') The dynamic parser determined an error.

Cause: Preprocessing or post processing of the Run-time Parser determined either a memory allocation error or an internal structural inconsistency.

Action: Attempt to reexecute the SQL statement. If the error persists, contact an authorized IBM service representative.

DJX524305 (X'00080011') The query tried to perform multi-site database access.

Cause: The query specified more than one location against which to perform the database access. This version of Classic Connect allows only one location for access within any given query.

Action: Replace multiple locations with the same location qualifiers in the query.

DJX524306 (X'00080012') Memory allocation error occurred while allocating CSCB.

Cause: An attempt to allocate memory for control information failed. This occurred while trying to process a query.

Action: Attempt to reexecute the SQL statement.

DJX524307 (X'00080013') A Cursor was referenced for which a Statement was not in a prepared state.

Cause: The application tried to use a Cursor that was tied to a Statement Name, and that statement was not prepared.

Action: Prepare the Statement Name and try the Cursor access again.

DJX524308 (X'00080014')   DJX524311 (X'00080017')

DJX524308 (X'00080014') A Cursor was attempted to be used which was not in an OPEN state.

Cause: The Cursor was not in an OPEN or valid state when a CLOSE or FETCH was issued against it.

Action: Open the Cursor first.

DJX524310 (X'00080016') A session is already connected to the location.

Cause: A CONNECT statement was issued for a location that is already in a connected state.

Action: Restart the application to close the connection.

DJX524311 (X'00080017') Memory allocation error occurred.

Cause: An attempt to allocate memory during data encoding or decoding failed in the API.

Action: Contact an authorized IBM service representative.

Chapter 20. ETB Subsystem Error Messages

The following messages are related to ETB subsystem errors.

DJX589824 (X'00090000') The event timing block header is invalid.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX589825 (X'00090001') The event timing block list is empty.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX589827 (X'00090003') Wait interrupted by message.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX589828 (X'00090004') Wait interrupted by expiration of event.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX589829 (X'00090005') Invalid time value.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

Chapter 21. Intertask Communication Messages

The following messages are related to intertask communications.

DJX655360 (X'000A0000') Specified port entry is larger than table.

Cause: There are two possible causes of this error. The first might come from the function IVPortInitTable. The error is issued if the port specified by the port handle variable is beyond the end of the port table. The second possible cause is in the HIPEntryInit function. The error is issued if the port specified by the port handle variable is beyond the end of the port table.

Action: Contact an authorized IBM service representative.

DJX655361 (X'000A0001') Attempt to reuse a port table entry that was not freed.

Cause: This message is issued from the HIPEntryInit function. It indicates that the port is not in the correct state (FREE) to be initialized. The port might be open and should be closed before calling the HIPEntryInit function.

Action: Contact an authorized IBM service representative.

DJX655362 (X'000A0002') An invalid queue type was specified.

Cause: This message can be issued from either the LLVCheckMsg or the LLRecvMsg function. It indicates that the queue parameter passed to the function is not valid. This can be caused by no queue specified or if multiple queues are specified for the LLRecvMsg function.

Action: Contact an authorized IBM service representative.

DJX655363 (X'000A0003') An invalid port handle was specified.

Cause: This message can be issued from several functions. It indicates that the port handle passed to the function is invalid. For the LLOpenMsgPort and LLVCheckMsg functions, it might be caused if the port indicated by the handle is outside the port table. It might also be caused if the specified port state is free or the instance value of the handle does not match that of the port.

For the LLShutMsgPort function, it might be caused if the port indicated by the handle is outside of the port table. For the LLGetNextMsg and LLRecvMsg functions, it might be caused if the port indicated by the handle is outside of the port table. It might also be caused if the specified port state is not open or the instance value of the handle does not match that of the port.

Action: Contact an authorized IBM service representative.

DJX655365 (X'000A0005') Warning, an attempt was made to get a message from an empty queue.

Cause: This warning can be issued from either the VHGetNextMsg or the VHRecvMsg function. It indicates that no message is currently available on the requested queue.

Action: Reissue the request at a later time.

DJX655366 (X'000A0006') The message direction specified is invalid.

Cause: This error can be issued from either the VHSndMessage or the VHSwMessage function. It indicates that the bMsgDirection field of the message header is invalid. It might not have been initialized properly or memory may have been overwritten.

Action: Contact an authorized IBM service representative.

DJX655367 (X'000A0007') The session table entry does not exist for the specified index.

Cause: This is a configuration error.

Action: Contact an authorized IBM service representative.

DJX655368 (X'000A0008') Unable to obtain memory for the port table.

Cause: This error occurs at initialization time. The amount of memory required for the port table could not be obtained.

Action: Contact an authorized IBM service representative.

DJX655369 (X'000A0009') The session table entry is not valid.

Cause: This error can come from either the VHSndMessage or the VHSwMessage function. In the case of the VHSndMessage function, it indicates that the fSessionLocal field of the session table entry is not LOCAL_SESSION or REMOTE_SESSION. In the case of the VHSwMessage, it indicates that the fSessionLocal field of the session table entry is not LOCAL_SESSION.

Action: Contact an authorized IBM service representative.

DJX655370 (X'000A000A') The caller's input message port is not open, attempted to close port.

Cause: If the severity is a recoverable error, then the caller attempted to close a port that is not open. If the severity is a warning, then a caller sent a message to a port that is not opened yet. The warning is useful to indicate whether a task initialized itself.

Action: No action is required.

DJX655371 (X'000A000B') The port entry is already opened.

Cause: An attempt was made to open a message port that was already open.

Action: Contact an authorized IBM service representative.

DJX655372 (X'000A000C') The message priority is not valid.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX655373 (X'000A000D') The task terminated.

Cause: An attempt was made to send a message to a task that terminated.

Action: Contact an authorized IBM service representative.

DJX655374 (X'000A000E') Feature not yet implemented.

Cause: The function is not supported, or there is a configuration or syntax error in an SQL statement.

Action: Check for and correct configuration or syntax errors in SQL statements. If this does not correct the problem, contact an authorized IBM service representative to see if the function is supported.

DJX655375 (X'000A000F') A session table entry previously in use was freed and is no longer valid.

Cause: This is an informational message.

Action: No action is required.

Chapter 22. DMSI Common Error Messages

The following messages are related to DMSI common errors.

DJX720896 (X'000B0000') A nonrecoverable error was generated, the task was stopped.

Cause: The DMSI encountered an error that placed it in a state that is not dependable. Therefore, the DMSI is terminating.

Action: Ensure that the client and communications links are in a correct state.

DJX720897 (X'000B0001') The DMSI failed initialization.

Cause: The DMSI could not initialize itself.

Action: Check that the DMSI supports the operating system on which it is running and supports the client revision that is installed.

DJX720899 (X'000B0003') An internal error was detected.

Cause: The DMSI encountered an internal error that prevents it from processing the statement. The statement is ignored, and processing will continue with the next statement.

Action: Check that the statement is valid. Report the statement that is causing the error to an authorized IBM service representative.

DJX720900 (X'000B0004') DB2 DMSI engine cannot connect to DB2.

Cause: This is an internal error initializing the Call Attach Facility.

Action: Add a DSNTRACE DD SYSOUT=A DD card to the DB2 JCL, and analyze the DSNTRACE output to determine where the CAF failure occurred. Contact an authorized IBM service representative if you are unable to determine the source of the failure.

DJX720901 (X'000B0005') The DMSI attempt to disconnect from DB2 failed.

Cause: This is an internal error detaching from the Call Attach Facility.

Action: Contact an authorized IBM service representative.

DJX720902 (X'000B0006') The referenced cursor name is undefined.

Cause: A user application program is attempting to use a cursor that was not declared.

Action: Correct the cursor name in the application program.

DJX720903 (X'000B0007') The DMSI encountered an unexpected data type.

Cause: The data type code is unknown.

Action: Be sure to use valid SQLDA data type codes.

DJX720904 (X'000B0008') The DMSI timed out.

Cause: The DMSI did not receive a response from the client within the expected time frame.

Action: Verify that the client is running and that the communication link is stable.

DJX720905 (X'000B0009') Unable to get a port handle.

Cause: All available client ports are in use.

Action: Disconnect from a connected database.

DJX720906 (X'000B000A') The DMSI failed on check of port.

Cause: The connection is unstable.

Action: Verify the stability of the communication link.

DJX720907 (X'000B000B') Unable to get the next message.

Cause: A communication error between the DMSI and client is preventing the DMSI from getting the next message.

Action: Verify the stability of the communication link.

DJX720908 (X'000B000C') The DMSI received an unexpected message type.

Cause: The client request is unknown to the DMSI. The request is ignored.

Action: Contact an authorized IBM service representative.

DJX720909 (X'000B000D') The expected SSDU was not encountered.

Cause: An unrecognized data message was received by the DMSI. The current statement is ignored.

Action: Contact an authorized IBM service representative.

DJX720910 (X'000B000E') The DMSI received a request for nonimplemented functionality.

Cause: The client requested a function of the DMSI that is not yet implemented in the DMSI.

Action: Contact an authorized IBM service representative to get a DMSI and client that match in revision.

DJX720911 (X'000B000F') The DMSI encountered an error encoding/decoding ASN1.

Cause: An internal error was encountered when encoding or decoding a data message.

Action: Contact an authorized IBM service representative.

DJX720912 (X'000B0010') An entry with this session ID was not found.

Cause: An internal error corrupted the session identifier for the DMSI.

Action: Contact an authorized IBM service representative.

DJX720913 (X'000B0011') The DMSI received a stop task request.

Cause: The application requested a disconnect from the DMSI. This is an informational message.

Action: No action is required.

DJX720914 (X'000B0012') The DMSI received an incorrect SQL statement.

Cause: An internal error was encountered when encoding or decoding an SQL statement.

Action: Contact an authorized IBM service representative.

DJX720915 (X'000B0013') The DMSI had a wrong arlist.

Cause: The data type specified in an SQLDA is unknown.

Action: Be sure to use valid SQLDA data type codes.

DJX720916 (X'000B0014') Unable to find a free CURSOR.

Cause: The application attempted to open more than the maximum number of simultaneously open cursors supported.

Action: Close an open cursor before attempting to open another cursor.

DJX720917 (X'000B0015') Maximum number of SQLVARS exceeded.

Cause: The application attempted to select more than the maximum number of items in a select list.

Action: Eliminate some of the select items, or break the select into two select statements.

DJX720918 (X'000B0016') The DMSI exceeded the memory limit.

Cause: Internal memory limits have been exceeded.

Action: Break the select statement into multiple statements that each return a subset of the desired data.

DJX720920 (X'000B0018') LD_ERR_MAX_RETRY_EXCEEDED

Cause: The maximum number of message retries was exceeded. The DMSI tried to resend a response message through a CPI back to the client system. This failed 10 times since the DMSI started. The Communication link is not considered reliable anymore. The DMSI will shut down.

Action: If the DMSI shuts down often, contact an authorized IBM service representative to inspect the communications paths.

Chapter 23. Router Error Messages

The following messages are related to router errors.

DJX786432 (X'000C0000') Feature not yet implemented.

Cause: The function is not supported, or there is a configuration or syntax error in an SQL statement.

Action: Check for and correct configuration or syntax errors in SQL statements. If this does not correct the problem, contact an authorized IBM service representative to see if the function is supported.

DJX786433 (X'000C0001') Router initialization error.

Cause: Unable to initialize the router subsystem.

Action: Restart the Classic Connect.

DJX786435 (X'000C0003') Bad time limit value in the configuration table.

Cause: Configuration table gave a bad value. This is an installation error.

Action: Verify that value for ROUT PROC TIMELIMIT is greater than 0 in the Classic Connect configuration file.

DJX786436 (X'000C0004') Invalid trace level value in the configuration table.

Cause: Configuration table gave a bad value. This is an installation error.

Action: Verify that the value for ROUTER TRACE LEVEL is greater than 0 in the Classic Connect configuration file.

DJX786440 (X'000C0008') The message buffer pointer is NULL.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786441 (X'000C0009') Router is unable to process message.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786442 (X'000C000A') Unrecognized message type.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786443 (X'000C000B') Router detected the illegal use of message type.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786445 (X'000C000D') Unable to forward message.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786446 (X'000C000E') Unable to communicate with remote system.

Cause: Failure to communicate with remote node. Possible causes:

- Remote system is down
- Remote CI is down
- Communications link between the two systems is down

Action: Contact an authorized IBM service representative.

DJX786447 (X'000C000F') Classic Connect is unable to bind to a remote router.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786448 (X'000C0010') Router was unable to bind to remote service task.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786449 (X'000C0011') Unable to bind to CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786450 (X'000C0012') Invalid service ID.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786451 (X'000C0013') Unable to disconnect from an unconnected session.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786452 (X'000C0014') The router attempted to cancel an inappropriate task.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786456 (X'000C0018') An unrecognized service name was encountered.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786457 (X'000C0019') Index overflow occurred.

Cause: The Router Tasks Table is not synchronized with the active task table. The router will not be able to continue.

Action: Restart the Classic Connect.

DJX786459 (X'000C001B') The task number is outside the allowable range.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786460 (X'000C001C') Probable attempt to disconnect from a task other than the service task.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786461 (X'000C001D') The table is corrupt.

Cause: No Router-Service session ID in Router Task Table. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786462 (X'000C001E') The service information table index is not valid.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786463 (X'000C001F') Task port handle is not available.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786464 (X'000C0020') The service task policy is not recognized.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786465 (X'000C0021') The task was not started.

Cause: Router is attempting to kill an inactive task.

Action: No action is required.

DJX786466 (X'000C0022')   DJX786482 (X'000C0032')

DJX786466 (X'000C0022') The global task ID is not in the Service Information Table.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786467 (X'000C0023') Node ID is out of range.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786468 (X'000C0024') The router-to-router session does not exist.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786469 (X'000C0025') No CI connects to the node where the service resides.

Cause: The application attempted to bind to a service on a remote Classic Connect node, but there is no CI to connect the client instance to the server node. This error might occur because the Classic Connectkernel configuration file contains an error or because the CI, which should connect this node to the service node, failed. This might be the result of one of the following conditions:

* The node ID in field 2 of the SYSLOCATION entry for the desired service does not appear in field 1 of any NODE ADDRESS ENTRY
* The CCI service name in field 2 of the matching NODE ADDRESS ENTRY does not appear in field 2 of any TASK INFO ENTRY
* There was an error in the SYSLOCATION, NODE ADDRESS ENTRY, or TASK INFO ENTRY.

If this error resulted from termination of the CI, which should connect this node with the node where the service resides, the Classic Connect log will contain an error message pertaining to termination and restarting of a CI (for example, 786495, 786499, or 786501).

Action: Check the message log for errors that result in CI termination.

DJX786470 (X'000C0026') The CI name is not in the router session table.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786471 (X'000C0027') The session ID is not found in the Router Session Table.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786476 (X'000C002C') Router time out.

Cause: Router timed out. Router has been idle for an extended period of time. This condition is logged and will continue.

Action: No action is required.

DJX786477 (X'000C002D') The event timing block is not set.

Cause: The event timing block is not set.

Action: No action is required.

DJX786479 (X'000C002F') An invalid router state.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786480 (X'000C0030') Invalid global task ID.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786481 (X'000C0031') Router is terminating.

Cause: A stop task request was received. Router will terminate.

Action: No action is required.

DJX786482 (X'000C0032') This is a router internal error.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786484 (X'000C0034') Task was terminated by router.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786485 (X'000C0035') Invalid pointer detected.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786486 (X'000C0036') Trying to send nonITC message type to remote node.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786487 (X'000C0037') Message direction has neither of the permitted values.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786488 (X'000C0038') Trying to bind a guarded service without supplying a user ID.

Cause: The DMSI or other service is guarded, but the user or application program did not provide a user ID.

Action: Supply USERID and USERPASSWORD entries in the application configuration file.

DJX786489 (X'000C0039') CI connecting client and server went down during session.

Cause: While a client and server running on different instances of Classic Connect were connected in a session, the Router detected a failure of the communications interface connecting those two instances. The session was terminated. No session can be established between the two instances until the CI is up again.

Action: Try again in a few minutes.

DJX786490 (X'000C003A') CI reported the link down during the session.

Cause: While a client and server running on different instances of Classic Connect were connected in a session, the communications interface connecting the two instances detected that the link between them had gone down. The session was terminated. The next time a client running on one of those instances attempts to bind to a service on the other instance, the CI will attempt to reestablish the link between the two instances.

Action: Try again in a few minutes.

DJX786491 (X'000C003B') Task data is unavailable for the CI which ended.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX786492 (X'000C003C') CI Restart Interval too big for a long integer.

Cause: A value specified for the configuration parameter CI RESTART INTERVAL is too big to be held in a long integer on this computing system. The limit is 2,147,483,647 milliseconds on all platforms currently supported. The Classic Connect will use the default value of CI RESTART INTERVAL.

Action: Update the configuration file, and provide a valid CI RESTART INTERVAL value. The default value of the restart interval is too long.

DJX786493 (X'000C003D') CI Restart Interval specified REPEAT without a preceding number.

Cause: Configuration parameter CI RESTART INTERVAL was specified as REPEAT without first giving a number to be repeated. Classic Connect will use the default value of the CI RESTART INTERVAL.

Action: Update the configuration file, and provide a REPEAT value.

DJX786494 (X'000C003E') DJX786503 (X'000C0047')

DJX786494 (X'000C003E') CI Restart Interval is neither an integer nor the word REPEAT.

Cause: Configuration parameter CI RESTART INTERVAL was incorrectly specified. Classic Connect will use the default value of the CI RESTART INTERVAL.

Action: Update the configuration file, and provide a valid CI RESTART INTERVAL value.

DJX786495 (X'000C003F') A CI exhausted the list of restart intervals.

Cause: A communications interface failed and was restarted as many times as there are intervals specified in the CI RESTART INTERVAL configuration parameter, which did not include the REPEAT specification. Classic Connect will make no further attempt to restart the CI.

Action: Check for link errors, and restart the kernel.

DJX786496 (X'000C0040') CI Restart specified negative number of milliseconds.

Cause: A negative value was specified for the configuration parameter CI RESTART INTERVAL. Only nonnegative integers and the word REPEAT are valid values for this parameter. Classic Connect will use the default value of the CI RESTART INTERVAL.

Action: Update the configuration file, and provide a valid CI RESTART INTERVAL value.

DJX786497 (X'000C0041') CI restart data out of range.

Cause: This is an internal error.

Action: Update the configuration file to provide correct restart information.

DJX786498 (X'000C0042') Unable to delete the CI task number from the session table.

Cause: This problem is usually caused by a CI terminating before a session was established.

Action: Examine the Classic Connect log for an earlier message from the CI indicating the reason for the termination.

DJX786499 (X'000C0043') CI task number terminated.

Cause: Using the value from the CI RESTART INTERVAL parameter, router will restart the CI immediately.

Action: Examine the Classic Connect log for an earlier message from the CI, indicating the reason for the termination.

DJX786500 (X'000C0044') The CI task is being restarted.

Cause: This message refers to the CI task mentioned in message number 786499. It identifies the CI being restarted. This is an informational message.

Action: No action is required.

DJX786501 (X'000C0045') CI task terminated.

Cause: A CI terminated and is being scheduled for deferred restart with a nonzero entry in the CI RESTART INTERVAL parameter. Using the value from the RESTART INTERVAL parameter, the router will restart the CI. This is an informational message.

Action: No action is required.

DJX786502 (X'000C0046') The CI task will be started.

Cause: The CI specified in the message terminated and is being scheduled for deferred restart with a nonzero entry in the CI RESTART INTERVAL parameter.

Action: Examine the Classic Connect log for an earlier message from the CI indicating the reason for the termination, and correct that problem.

DJX786503 (X'000C0047') The wait time for a CI restart expired.

Cause: The CI is now being restarted. A deferred restart, previously reported by a pair of messages, 786502 and 786503, is now taking place. This is an informational message.

Action: No action is required.

DJX786504 (X'000C0048') Every service task in the pool is busy with a client.

Cause: All tasks in the pool (set by the value assigned to field 8 of the TASK INFO ENTRY) are busy.

Action: This is most likely a temporary condition. Submit the application at another time. If the condition persists, the setting for field 8 of the TASK INFO ENTRY might need to be increased.

Chapter 24. VHS Header Stack Error Messages

The following messages are related to virtual host system (VHS) header stack errors.

DJX851968 (X'000D0000') No Subsystem headers are left.

Cause: An attempt to extract header information from a message failed because there is no information available.

Action: Contact an authorized IBM service representative.

DJX851969 (X'000D0001') Invalid pointer received.

Cause: An attempt to extract header information from a message failed because an invalid message pointer was passed.

Action: Contact an authorized IBM service representative.

DJX851970 (X'000D0002') Invalid header type received.

Cause: An attempt was made to extract or store invalid header information from a message.

Action: Contact an authorized IBM service representative.

DJX851971 (X'000D0003') No room left for headers.

Cause: An attempt was made to store header information in a message in which there was not enough room.

Action: Contact an authorized IBM service representative.

DJX851972 (X'000D0004') Received incorrect header.

Cause: A message header protocol error occurred while trying to extract header information.

Action: Contact an authorized IBM service representative.

Chapter 25. MVS VTAM LU 6.2 Log Error Messages

The following messages are related to MVS
VTAM LU 6.2 log errors.

**DJX917504 (X'000E0000') VTAM SETLOGON Failed.
CI program error, local LU or VTAM is
not active.**

Cause: The CI was unable to log on to the VTAM
application. This is either caused by an incorrect LU
name in the configuration file, or if VTAM is not running.

Action: Check the system configuration file, and check
that VTAM is up.

**DJX917505 (X'000E0001') Close ACB failed. CI
program error, local LU or VTAM is not
active.**

Cause: The CI was unable to close an ACB in
connection with a VTAM session. This can be caused by
an error in CI processing, or by VTAM not running.

Action: Contact an authorized IBM service
representative.

**DJX917506 (X'000E0002') VTAM RECEIVE failed. CI
program error, local LU error or VTAM is
not active.**

Cause: An APPC command to VTAM to receive any
messages failed. This can be caused by an error in CI
processing, or by VTAM not running.

Action: Contact an authorized IBM service
representative.

**DJX917507 (X'000E0003') No remote partner is
defined in Configuration File. CI will
terminate.**

Cause: The system configuration did not specify any
remote connections for this CI to use. The CI is
terminating normally.

Action: No action is required.

**DJX917508 (X'000E0004') CI received a message
other than TR_BIND_RES during
initialization.**

Cause: During initialization, the CI was expecting a
message of type TR_BIND_RES but received a
message of a different type first. That message was
ignored.

Action: No action is required. If the error persists,
contact an authorized IBM service representative.

**DJX917510 (X'000E0006') CI tried to start
communication with a Remote Classic
Connect and failed.**

Cause: CI tried to start communications with a Remote
Classic Connect but failed. If the Remote node is not
active, this message can be ignored. The communication
will be established by the Remote node when it comes
up. If the Remote node is already up, then the system
configuration file might have the wrong LU names
specified in it.

Action: Start the remote node, or correct the system
configuration file if there is an error.

**DJX917511 (X'000E0007') VTAM APPC ALLOCATE
error.**

Cause: The CI was unable to allocate a local
conversation with which to communicate. This can be
caused by an incorrect LU name or a bad Mode in the
system configuration file.

Action: Correct the system configuration file.

**DJX917512 (X'000E0008') CI tried to free memory.
Passed from VTAM Exit routine.**

Cause: A memory block, passed from VTAM, could not
be freed. This can be a CI processing bug or a VTAM
memory allocation bug.

Action: Contact an authorized IBM service
representative.

DJX917513 (X'000E0009') DJX917523 (X'000E0013')

DJX917513 (X'000E0009') CI received remote information containing an LU name that is unknown.

Cause: A VTAM CNOS packet was received containing an LU name that is not found in the local system configuration.

Action: Check that the local and remote system configuration files contain the same LU names in the Node Address Table entries for the nodes in question.

DJX917514 (X'000E000A') CI received CNOS from LU whose state in LUCB does not match.

Cause: A VTAM control block was in an inconsistent state during VTAM message processing. Processing will continue.

Action: Contact an authorized IBM service representative.

DJX917515 (X'000E000B') VTAM RECEIVE FMH5 error. CI program error, local LU error, or VTAM is not active.

Cause: An error occurred while trying to service a FMH5 VTAM interrupt.

Action: Contact an authorized IBM service representative.

DJX917516 (X'000E000C') CI received a LOSS packet containing an LU name that is unknown.

Cause: A VTAM LOSS packet was received containing an LU name that is not found in the local system configuration.

Action: Check that the Local and Remote system configuration files contain the same LU names in the Node Address Table entries for the nodes in question.

DJX917517 (X'000E000D') Message with bad Message Stamp was received.

Cause: A message with a bad message stamp was received by the CI from within the Classic Connect kernel.

Action: Contact an authorized IBM service representative.

DJX917518 (X'000E000E') Close ACB failed. CI program error, local LU, or VTAM is not active.

Cause: The CI was unable to close an ACB in connection with a VTAM session. This can be caused by an error in CI processing or by VTAM not running.

Action: Contact an authorized IBM service representative.

DJX917519 (X'000E000F') CI terminated normally.

Cause: This is an informational message advising that the CI ended.

Action: No action is required.

DJX917520 (X'000E0010') CI received a message whose destination is not REMOTE.

Cause: A message was sent to the CI that has a local destination. This could have resulted from an incorrect routing of messages within the Classic Connect system.

Action: Contact an authorized IBM service representative.

DJX917521 (X'000E0011') Message does not have large enough space for CI headers.

Cause: An attempt to store CI information in the message header failed due to lack of space.

Action: Contact an authorized IBM service representative.

DJX917522 (X'000E0012') VTAM Send Data and Confirmed request failed.

Cause: An error occurred while sending a message with a confirmation request.

Action: Contact an authorized IBM service representative.

DJX917523 (X'000E0013') VTAM CHECK returns bad return code.

Cause: A call to VTAM CHECK returned an error code. More detailed information will be present in the Classic Connect log.

Action: Contact an authorized IBM service representative.

DJX917524 (X'000E0014') Receive ANY VTAM request received data on an invalid conversation.

Cause: A message was received with an invalid conversation. This could be a CI program error or a VTAM error.

Action: Contact an authorized IBM service representative.

DJX917525 (X'000E0015') VTAM Send Error request failed.

Cause: An error occurred while trying to send an error message over a conversation.

Action: Contact an authorized IBM service representative.

DJX917526 (X'000E0016') VTAM Message with bad Message Stamp was received.

Cause: A message with a bad message stamp was received by the CI from a REMOTE CI.

Action: Contact an authorized IBM service representative.

DJX917527 (X'000E0017') VTAM RECEIVE Specific Sync request failed.

Cause: While trying to synchronize data movement on a conversation, an error occurred.

Action: Contact an authorized IBM service representative.

DJX917528 (X'000E0018') VTAM Send Confirmed request failed.

Cause: An error occurred while trying to send a Confirm message.

Action: Contact an authorized IBM service representative.

DJX917529 (X'000E0019') VTAM RECEIVE Specific request failed.

Cause: An error occurred while trying to receive data on a conversation.

Action: Contact an authorized IBM service representative.

DJX917530 (X'000E001A') Deallocate is received on a Conversation while a message was being processed.

Cause: A conversation was deallocated while expecting more data to be sent.

Action: Contact an authorized IBM service representative.

DJX917531 (X'000E001B') Unexpected type of inbound data received from the network.

Cause: An unexpected incoming VTAM data packet was received.

Action: Contact an authorized IBM service representative.

DJX917532 (X'000E001C') Invalid CI message header received on Conversation.

Cause: A incoming message from VTAM did not have a proper CI header.

Action: Contact an authorized IBM service representative.

DJX917533 (X'000E001D') Incoming VTAM Message being processed has invalid state.

Cause: An incoming message packet from VTAM is not consistent with state information, which is expecting a different message packet.

Action: Contact an authorized IBM service representative.

DJX917534 (X'000E001E') VTAM Send Data request failed.

Cause: While the system was trying to send a data packet to VTAM, it detected an error. This could be caused by VTAM not being active.

Action: Contact an authorized IBM service representative.

DJX917535 (X'000E001F') CI received remote FMH5 information containing an LU name that is unknown.

Cause: A VTAM FMH5 packet was received containing an LU name that is not found in the local system configuration.

Action: Check that the Local and Remote system configuration files contain the same LU names as the Node Address Table entries for the nodes in question.

DJX917536 (X'000E0020') DJX917539 (X'000E0023')

DJX917536 (X'000E0020') Unable to allocate a Conversation Control Block.

Cause: A conversation control block was unable to be allocated due to lack of conversation space. It is possible that the Partner LU allocated more space than necessary for the conversation.

Action: Contact an authorized IBM service representative.

DJX917537 (X'000E0021') VTAM DEALLOCATE request failed. CI program error, or local LU or VTAM is not active.

Cause: While the system was trying to deallocate a conversation, it detected an error.

Action: Contact an authorized IBM service representative.

DJX917538 (X'000E0022') CI general program error.

Cause: This is a CI informational error message that indicates an error in identifying the specifics of an error condition. This means a program error occurred.

Action: Contact an authorized IBM service representative.

DJX917539 (X'000E0023') CI general program error. CI program error or local LU or VTAM is not active.

Cause: While the system was trying to perform a Reset Receive on a conversation, an error occurred.

Action: Contact an authorized IBM service representative.

Chapter 26. MVS VTAM LU 6.2 Operator Error Messages

The following messages are related to MVS VTAM LU 6.2 operator errors.

DJX983040 (X'000F0000') VTAM ACB Open Failed. Opening of a VTAM ACB failed.

Cause: This can be caused by an incorrect LU name being specified in the system configuration file, or by VTAM not being active.

Action: Contact an authorized IBM service representative.

DJX983041 (X'000F0001') Local LU is not defined in Remote Configuration file. CI cannot communicate with this node.

Cause: During remote handshaking, the Remote CI sent an LU name that is not known. The system configuration files should specify the same LU names.

Action: Contact an authorized IBM service representative.

DJX983042 (X'000F0002') CI Terminated with error.

Cause: This is an informational message saying that previous CI errors have been logged.

Action: Contact an authorized IBM service representative.

Chapter 27. VHS Message Allocation Error Messages

The following messages are related to virtual host system (VHS) message allocation errors.

DJX1048576 (X'00100000') VHS message buffer not allocated, requested amount of memory not available.

Cause: The message indicates that currently there is not enough memory available to allocate a message of the requested size.

Action: Keep trying, or restart subsystem with a larger buddy system.

DJX1048577 (X'00100001') VHS message buffer not freed, specified message address is invalid.

Cause: The message pointer specified is not within the message area address space.

Action: Contact an authorized IBM service representative.

DJX1048578 (X'00100002') VHS message buffer operation failed, message is not currently allocated.

Cause: The pointer given for an existing message indicates the message is not allocated. A recoverable error is returned if the message is currently free. A nonrecoverable error indicates one of two possible errors: the pointer is outside the message area, or it is attempting to free a block. The block is already on the free list. The nonrecoverable instance should never occur.

Action: Contact an authorized IBM service representative.

DJX1048579 (X'00100003') VHS message buffer not freed, requester is not owner of message.

Cause: Only the current owner of messages can perform message functions. The requester of an operation is not the current owner. The requester might have freed or enqueued, or given up control in some other manner.

Action: Contact an authorized IBM service representative.

DJX1048580 (X'00100004') VHS message buffer not freed, specify message address if not on a proper boundary.

Cause: The address given for a message is invalid for the size of the message. The bad address specified for the message, the message control area, or both might be overwritten.

Action: Contact an authorized IBM service representative.

DJX1048581 (X'00100005') VHS message buffer not freed, start picket of message is invalid.

Cause: The leading picket is not correct. The message memory was overwritten.

Action: Contact an authorized IBM service representative.

DJX1048582 (X'00100006') VHS message buffer not freed, end picket of message is invalid.

Cause: The trailing picket is not correct. The message memory was overwritten.

Action: Contact an authorized IBM service representative.

Chapter 28. MVS System Error Message

The following messages are generated by Classic Connect on MVS. In general, a reference to a configuration parameter is a reference to Classic Connect configuration parameters on MVS. These messages do not relate to the configuration parameters of the DataJoiner instances' Classic Connect interface.

DJX1114368 (X'00110100') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1114369 (X'00110101') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1114370 (X'00110102') The Active Task Table is full.

Cause: All entries in the active task table (ATT) are occupied. This situation occurs when there are as many Classic Connects tasks active as there are entries in the ATT. The number of entries is determined at the Classic Connectsubsystem initialization time by the initialization parameter MAXIMUM NUMBER OF TASKS = *xxxxx*, where *xxxxx* denotes the number of tasks. There can never be more tasks active than *xxxxx* at any given time. To increase the number, stop the subsystem and restart it with a larger value of *xxxxx*. This problem might be temporary because the Classic Connecttasks start and end dynamically.

Action: Restart your application at a later time because this problem might simply be caused by a saturated system. If you suspect a software problem with the Classic Connect, cancel the subsystem with the DUMP option and forward the error information to an authorized IBM service representative. Make sure your installation

is running the subsystem with a reasonable value for MAXIMUM NUMBER OF TASKS. Each client occupies at least two entries in the active task table.

DJX1114371 (X'00110103') The cross-memory POST of the external task failed.

Cause: After an entry is added to the active task table, the external task is posted to allow it to continue processing. This POST is done asynchronously by the kernel. If the external task terminates after the POST is issued and before the POST is executed by the operating system, this error occurs.

This message should not be submitted while the external task is in a WAIT state (where it is supposed to be during execution of LLADDENT).

Action: If this message appears without the external task terminating, cancel the subsystem with the DUMP option and forward the error information to an authorized IBM service representative.

DJX1114372 (X'00110104') An error occurred when obtaining the port handle.

Cause: For each restarted task, there must be a port handle obtained to enable the messaging system to send and receive messages on behalf of that task. If this port handle cannot be obtained, the new task cannot be initialized. This could be a serious error within the kernel.

Action: If the kernel continues processing without repeating this or other serious error messages, ignore the message and rerun the application. If this error recurs, cancel the kernel with the DUMP option and forward the error information to an authorized IBM service representative.

DJX1114373 (X'00110105') Server not found in Active Task Table.

Cause: The DMSI tries to find its preallocated entry in the active task table. This search failed. The information needed to identify the DMSI is not passed correctly to the kernel.

Action: Under debug conditions, look for storage overlays in the DMSI address space. If the problem recurs, call an authorized IBM service representative.

%3Eδ ξδρ)%3Eνρ ♥ιδσ

DJX1114624 (X'00110200') DJX1114632 (X'00110208')

DJX1114624 (X'00110200') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1114625 (X'00110201') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1114626 (X'00110202') The Time-Of-Day clock is invalid.

Cause: During client initialization, the Time-Of-Day clock (TOD) is accessed to store the time value for accounting and statistic purposes. If the CPU indicates an invalid TOD, the TOD clock value is unreliable. This is considered a fatal error. The operating system should react to the invalid TOD state immediately as well.

Action: This is a fatal error for the operating system and must be handled accordingly. It might require an IPL of the operating system.

DJX1114627 (X'00110203') SMF is not recording.

Cause: The operating system was IPLed without SMF recording. The Classic Connect is not designed to run in such an environment because it does not allow for accounting.

Action: Re-IPL the operating system using the SMF option.

DJX1114628 (X'00110204') Component initialization failed.

Cause: During the initialization of the client, one of the function calls for special component initialization returned with an error condition. This is a fatal condition at this time.

Because most initialization functions are building and initializing tables and loading parameters and modules, the problem should be related to a shortage in virtual memory. Probable causes are a REGION size that is too small or excessive initialization parameter values for the client.

Action: Check the operating system parameters related to the allocation of virtual storage at JOB, STC, and LOGON initialization time.

DJX1114629 (X'00110205') An internal error occurred.

Cause: While retrieving information from MVS system control blocks, an inconsistency was found. This problem occurs only when those control blocks are overlaid. This is a fatal situation and cannot be recovered.

Action: Restart the application. If the problem recurs, make sure that the software calling the client didn't overlay system storage, especially the ASCB and related areas.

DJX1114630 (X'00110206') Cross-memory call failed.

Cause: The attempt to communicate with the Task Monitor inside the kernel failed. This is most likely a performance problem; for example, the Task Monitor is too busy to accept your call. It also can happen if the Task Monitor terminates at the moment it is called.

Action: Rerun the application. If the problem persists, cancel the kernel with the DUMP option, and forward the problem to an authorized IBM service representative.

DJX1114631 (X'00110207') Timeout for cross-memory wait.

Cause: The client waits for a limited time to be posted by the kernel. Because the processing time for the Init request to the Task Monitor is very short, this error can happen only when the cross-memory POST is scheduled too late. A possible cause is an overloaded operating system.

Action: Restart the application specifying a larger value for EXTERNAL TASK WAIT TIME, or specify a larger value for the DMSI's EXTERNAL TASK WAIT TIME.

DJX1114632 (X'00110208') Initialization failed inside the kernel.

Cause: The part of the external task initialization that must be done by the kernel did not complete successfully.

Action: Restart the application. If the problem persists, restart the kernel.

DJX1114633 (X'00110209') The Classic Connect kernel (subsystem) is not active.

Cause: The subsystem with the name specified in the SUBSYSTEM NAME parameter in the client configuration file is not active. Clients cannot be started before the kernel is started and completely initialized.

Action: Make sure that the SUBSYSTEM NAME parameter specifies the correct name. If the subsystem is not active, ask your system operator to start it. Wait for the confirmation message DJXSI INIT SUCCESSFUL. Then start your applications.

DJX1114640 (X'00110210') The subsystem name provided by the user is too long.

Cause: A subsystem name of more than 4 characters was specified in the configuration file.

Action: Correct the SUBSYSTEM NAME parameter in the client configuration file.

DJX1114641 (X'00110211') Not enough virtual storage available.

Cause: There was not enough virtual memory to allocate the messaging system. At this time, it is a fatal error because the client is initializing.

Action: Check the REGION size allocated for the client. Also check the configuration file for excessive parameter specifications, especially for the table parameters. Comments within the configuration file are not read into memory.

DJX1114880 (X'00110300') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1114881 (X'00110301') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1114882 (X'00110302') An error occurred when obtaining the port handle.

Cause: For each restarted task, there must be a port handle obtained to enable the messaging system to send and receive messages on behalf of that task. If this port handle cannot be obtained, the new task cannot be initialized.

Action: This could be a serious error within the kernel. If the kernel continues processing without repeating this or other serious error messages, ignore the message and rerun the application. If this error recurs, cancel the kernel with the DUMP option, and forward the error information to an authorized IBM service representative.

DJX1114883 (X'00110303') The Active Task Table is full.

Cause: All entries in the active task table (ATT) are occupied. This situation occurs when there are as many Classic Connecttasks active as there are entries in the ATT. The number of entries is determined at Classic Connectsubsystem initialization time by the initialization parameter MAXIMUM NUMBER OF TASKS = *xxxxx*, where *xxxxx* denotes the number of tasks. There can never be more tasks active than *xxxxx* at any given time. To increase the number, stop the subsystem and restart it with a larger value of *xxxxx*. This problem might be temporary because Classic Connecttasks start and end dynamically.

Action: Restart your application at a later time, because the problem might simply be caused by a saturated system. If you suspect a software problem with the Classic Connect, cancel the subsystem with the DUMP option and forward the error information to an authorized IBM service representative. Make sure your installation is running the subsystem with a reasonable value for MAXIMUM NUMBER OF TASKS. Each client occupies at least two entries in the active task table.

DJX1114884 (X'00110304') The ATTACH of a subtask failed.

Cause: The execution of the ATTACH macro call to the operating system failed. There are many possible reasons for this. The two most likely are that the virtual storage for the kernel is exhausted or that the load module which was supposed to be attached is not in the load library of the kernel.

Action: If the kernel ran out of virtual storage, there should be other indications for this situation as well. Restart the kernel with a larger REGION size. If that is not possible, experiment with the start policy for

DJX1114885 (X'00110305') DJX1115652 (X'00110604')

kernel-tasks, as described in the DataJoiner Classic Connect*MVS Installation and Configuration Guide*. See information about the TASK INFO ENTRY parameter. If the module is missing, make sure the task name in the TASK INFO ENTRY is specified correctly and the load module of that name is in the access path for the kernel load modules (STEPLIB, LINKLIST).

DJX1114885 (X'00110305') The requested module is not in the library.

Cause: Before the new task is attached, the load module is searched for in the directory of the load library to make sure that it is available. This search failed.

Action: Check the TASK INFO ENTRY parameters in the kernel configuration file and match them with the kernel's load library. Make sure that all module names are in the load library.

DJX1115136 (X'00110400') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1115137 (X'00110401') VHS vector table invalid.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1115138 (X'00110402') Virtual memory allocation failed.

Cause: There was not enough virtual memory to allocate the active task table. A fatal error occurred while the kernel was initializing.

Action: Check the REGION size allocated for the kernel. It should be larger than 8 MB to allow for successful kernel initialization. Also, check the configuration file for excessive parameter specification, especially for table parameters. Comments within the configuration file are not read into memory.

DJX1115392 (X'00110500') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1115393 (X'00110501') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1115648 (X'00110600') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1115649 (X'00110601') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1115650 (X'00110602') The DD for the configuration file is missing in the JCL.

Cause: The processing of the configuration file requires the use of a DD statement with the DDNAME VHSCONF in the JCL for the client, the kernel, and the DMSI. This DD statement is missing.

Action: Insert a DD statement with the DDNAME VHSCONF into the JCL that defines the configuration file.

DJX1115651 (X'00110603') Error opening the configuration file.

Cause: During the OPEN processing of the configuration file, an error occurred.

Action: Check the DD statement and the configuration file itself for proper specifications.

DJX1115652 (X'00110604') Error building configuration file in memory.

Cause: A previously stored pointer to the configuration file in memory is invalid. This is a fatal error at initialization time.

Action: Restart the kernel, and report the problem to an authorized IBM service representativeif the error recurs.

DJX1115653 (X'00110605') Virtual memory allocation failed.

Cause: There was not enough virtual memory to allocate the configuration file. A fatal error occurred as the kernel was initializing.

Action: Check the REGION size allocated for the kernel. It should be larger than 8 MB to allow for successful kernel initialization. Also check the configuration file for excessive size (comments are not read into memory).

DJX1115904 (X'00110700') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1115905 (X'00110701') VHS vector table access failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1115906 (X'00110702') The task name specified is not in the Active Task Table.

Cause: The task name provided to this function is not the name of a task known by the kernel. This can be a serious malfunction of the kernel.

Action: If the kernel continues operating normally, do nothing. Otherwise, stop the kernel and restart it.

DJX1115907 (X'00110703') The task number provided is higher than the maximum number.

Cause: The task number is checked against the maximum value. If the maximum value is exceeded, this message appears. It can be a serious condition inside the kernel.

Action: If the kernel continues operating normally, do nothing. Otherwise, stop the kernel and restart it.

DJX1116160 (X'00110800') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1116161 (X'00110801') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1116162 (X'00110802') Task number is invalid.

Cause:

Action: Correct the task number.

DJX1116416 (X'00110900') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1116417 (X'00110901') Access to Vector Table failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1116418 (X'00110902') The calling task is not in the Active Task Table.

Cause: The calling task, which is supposed to be a Classic Connect task, is not in the active task table. This is either a serious malfunction of the kernel or an integrity violation. For example, someone might have tried to execute this code without authorization.

Action: Cancel the kernel with the DUMP option, and forward all available error information to an authorized IBM service representative. Restart the kernel after replacing the modules in the load library of the kernel with the modules on the Classic Connectproduct delivery tape.

DJX1118208 (X'00111000') DJX1119232 (X'00111400')

DJX1118208 (X'00111000') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1118209 (X'00111001') Access to Vector Table failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1118464 (X'00111100') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1118465 (X'00111101') Access to Vector Table failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1118466 (X'00111102') The serialization of the Task Monitor failed.

Cause: Task Monitor processing is partially serialized using a lockword in the active task table. The lock could not be obtained. This is most likely a performance problem due to too many applications starting at the same time.

Action: Restart the application at a time when there are fewer applications competing with initialization.

DJX1118720 (X'00111200') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1118721 (X'00111201') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1118723 (X'00111203') Keyword value with more than 16 characters.

Cause: A keyword contains more than 16 characters.

Action: Restart the application at a time when there are fewer applications competing with initialization.

DJX1118976 (X'00111300') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1118977 (X'00111301') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1118978 (X'00111302') The serialization of the Task Monitor failed.

Cause: The Task Monitor processing is partially serialized using a lockword in the active task table. The lock could not be obtained. This is most likely a performance problem due to too many applications starting at the same time.

Action: Restart the application at a time when there are fewer applications competing with initialization.

DJX1119232 (X'00111400') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1119233 (X'00111401') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1119234 (X'00111402') The Time-Of-Day clock (TOD) is not valid.

Cause: During server initialization, the Time-Of-Day clock (TOD) is accessed to store the time value for accounting and statistic purposes. If the CPU indicates an invalid TOD, the TOD clock value is unreliable. This is considered a fatal error. The operating system should react to the invalid TOD state immediately as well.

Action: This is a fatal error for the operating system, and should be handled accordingly. For example, it might require an IPL of the operating system.

DJX1119235 (X'00111403') SMF is not recording.

Cause: The operating system was IPLed without SMF recording. The Classic Connect is not designed to run in such an environment because it does not allow for accounting.

Action: Re-IPL the operating system using the SMF option.

DJX1119236 (X'00111404') Component initialization failed.

Cause: During the initialization of the server, one of the function calls for special component initialization returned with an error condition. This is a fatal condition at this time.

Because most initialization functions are building and initializing tables and loading parameters and modules, the problem should be related to a shortage in virtual memory. Probable causes are a REGION size that is too small, or excessive values for initialization parameters for the server.

Action: Check the operating system parameters related to the allocation of virtual storage at JOB, STC, and LOGON initialization time.

DJX1119237 (X'00111405') An internal error occurred.

Cause: While retrieving information from MVS system control blocks, an inconsistency was found. This problem occurs only when those control blocks are overlaid. This is a fatal situation and cannot be recovered.

Action: Restart the application. If the problem recurs, make sure that the software calling the server did not overlay system storage, especially the ASCB and related areas.

DJX1119238 (X'00111406') Cross-Memory call failed.

Cause: The attempt to communicate with the Task Monitor inside the kernel failed. This is most likely a performance problem. For example, the Task Monitor might be too busy to accept your call. It can also happen if the Task Monitor terminates at the moment it is called.

Action: Rerun the application. If the problem persists, cancel the kernel with the DUMP option and forward the problem to an authorized IBM service representative.

DJX1119239 (X'00111407') Timeout for cross-memory wait.

Cause: The server waits for a limited time to be posted by the kernel. Because the processing time for the Init request to the Task Monitor is very short, this error can happen only when the Cross-Memory POST is scheduled too late. A possible cause is an overloaded operating system.

Action: Restart the application specifying a larger value for EXTERNAL TASK WAIT TIME.

DJX1119240 (X'00111408') Initialization failed inside the kernel.

Cause: The part of the external task initialization that is done by the kernel did not complete successfully.

Action: Restart the application. If the problem persists, restart the kernel.

DJX1119241 (X'00111400') The Classic Connect kernel (subsystem) is not active.

Cause: The subsystem with the name specified in the SUBSYSTEM NAME parameter in the server configuration file is not active. servers cannot be started before the kernel is started and completely initialized.

Action: Make sure that the SUBSYSTEM NAME parameter specifies the correct name. If the subsystem is not active, ask your system operator to start it. Wait

for the confirmation message DJXSI INIT SUCCESSFUL. Then start your applications.

DJX1119248 (X'00111410') The subsystem name provided by the user is too long.

Cause: A subsystem name of more than 4 characters was specified in the configuration file.

Action: Correct the SUBSYSTEM NAME parameter in the server configuration file.

DJX1119249 (X'00111411') Not enough virtual storage available.

Cause: There was not enough virtual memory to allocate the messaging system. At this time it is a fatal error because the server is still initializing.

Action: Check the REGION size allocated for the server (DMSI). Also check the configuration file for excessive parameter specification, especially for the table parameters. Comments within the configuration file are not read into memory. SMF is not recording.

DJX1119488 (X'00111500') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1119489 (X'00111501') Access to Vector Table failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1119744 (X'00111600') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1119746 (X'00111602') GETMAIN failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1120000 (X'00111700') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1120002 (X'00111702') FREEMAIN failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1120256 (X'00111800') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1120257 (X'00111801') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1120258 (X'00111802') The internal START command failed.

Cause: DMSIs are started by the kernel using the internal START command. This command failed. The probable reason is a misspelled DMSI procedure name in the TASK INFO ENTRY of the kernel's configuration file or the absence of a procedure of that name in one of the procedure libraries used by the JES.

Action: Correct the kernel's configuration file or add the requested DMSI procedure to the procedure library.

DJX1120259 (X'00111803') The Active Task Table is full.

Cause: All entries in the active task table (ATT) are occupied. This situation occurs when there are as many Classic Connecttasks active as there are entries in the ATT. The number of entries is determined at Classic Connectsubsystem initialization time by the initialization parameter MAXIMUM NUMBER OF TASKS = *xxxxx*

DJX1120260 (X'00111804')   DJX1120516 (X'00111904')

it with a larger value of *xxxxx*

DJX1120517 (X'00111905') DJX1120531 (X'00111913')

in the access path for the kernel load modules
(STEPLIB, LINKLIST).

**DJX1120517 (X'00111905') Adding or deleting an
API/DMSI entry to/from the active task
table (ATT) failed.**

Cause: During the initialization of an external task, an
entry must be added to the active task table. This
process failed. A probable reason is a saturated kernel.

Action: Restart the application that failed at a later
time. If the problem occurs frequently, add more entries
to the active task table by specifying a larger value for
MAXIMUM NUMBER OF TASKS in the kernel's
configuration file.

**DJX1120518 (X'00111906') The internal START
command (MGCR) for a DMSI failed.**

Cause: DMSIs are started by the kernel using an
internal operator command. That command failed for the
same reasons a real operator command would fail if
something was wrong with the procedure that is being
started.

Action: Check the procedure library for the correct
specification of the name and the JCL for the DMSI to
be started. Also check the kernel's configuration file for
the correct specification of the DMSI procedure name in
the TASK INFO ENTRY parameter.

**DJX1120519 (X'00111907') The DETACH for a
subtask failed.**

Cause: The system call to remove a subtask from the
system did not complete successfully. This is not of
concern if it happens rarely.

Action: Report this problem to an authorized IBM
service representative if it happens frequently.

**DJX1120520 (X'00111908') The authorization check
for the client failed.**

Cause: The authorization check is done using standard
system calls to RACF or compatible products. If the
client application is not properly authorized, this error
message will appear.

Action: Check the authorization of the application
program and its user.

**DJX1120521 (X'00111909') Attempt to kill a
nonexistent task.**

Cause: An attempt was made to remove a task that did
not exist. This can happen due to the asynchronous task
management between the router and Task Monitor. This
is a warning message for the Router. This is an
informational message only.

Action: No action is required.

**DJX1120528 (X'00111910') Attempted to kill an
external task.**

Cause: An attempt was made to remove an external
task. This can happen due to the asynchronous task
management between the router and Task Monitor. This
is a warning message for the router. This is an
informational message only.

Action: No action is required.

DJX1120529 (X'00111911') Shut message port failed.

Cause: An attempt was made to shut an unopened
message port. This can happen due to the
asynchronous task management between the router and
Task Monitor. This is a warning message for the router.
This is an informational message only.

Action: No action is required.

**DJX1120530 (X'00111912') The cleanup for the
Semaphore Table failed.**

Cause: An attempt was made to clean unused
Semaphores. This can happen due to the asynchronous
task management between the router and Task Monitor.
This is a warning message for the router. This is an
informational message only.

Action: No action is required.

**DJX1120531 (X'00111913') An invalid Global Task ID
was passed by router.**

Cause: A generalized task identifier (GTID) was
passed that is no longer in use. This can happen due to
the asynchronous task management between the router
and Task Monitor. This is a warning message for the
router. This is an informational message only.

Action: No action is required.

DJX1120532 (X'00111914') The task kill process is completed.

Cause: A valid kill request was issued by the router task and was processed. This is an informational message only.

Action: No action is required.

DJX1122304 (X'00112000') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1122305 (X'00112001') Vector Table access failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1122306 (X'00112002') The cross-memory POST of an external task failed.

Cause: The attempt to signal the external task across memory boundaries failed. Most likely, the external task terminated in the time between issuing the cross-memory POST and scheduling the SRB.

Action: Check the processing status of the external task and the reason for its termination. Restart the application.

DJX1122307 (X'00112003') The task number provided is higher than the maximum number.

Cause: The task number is checked against the maximum value. If it is exceeded, this message appears. It can be a serious condition inside the kernel.

Action: If the kernel continues operating normally, do nothing. Otherwise, stop the kernel and restart it.

DJX1122308 (X'00112004') The external task ended.

Cause: While the kernel processed the request from an external task, that task ended processing for an unknown reason.

Action: Check the processing status of the external task and the reason for its termination. Restart the application.

DJX1122560 (X'00112100') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1122561 (X'00112101') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1122562 (X'00112102') Synchronization lock cannot be set.

Cause: The attempt to obtain the requested lock failed. The kernel was probably too slow in handling the requests due to heavy workload.

Action: If this error message appears very often, change the two parameters in the kernel configuration file (SPIN WAIT TIME and SPIN WAIT LOOP COUNT). Be careful not to set the parameters to a value that is too high because that might affect the kernel performance under normal load. Restart the application.

DJX1122816 (X'00112200') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1122817 (X'00112201') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1122818 (X'00112202') The LOAD for the error recovery module failed.

Cause: The execution of the LOAD macro call to the operating system failed. There are many possible reasons for this. The two most likely are that the virtual storage for the kernel is exhausted or that the load module DJXSIER is not in the authorized load library of the kernel.

DJX1122819 (X'00112203') DJX1122832 (X'00112210')

Action: If the kernel ran out of virtual storage, there should be other indications for this situation as well. Restart the kernel with a larger REGION size. If the module is missing, make sure that the load module DJXSIER is in the access path for the kernel load modules (STEPLIB, LINKLIST).

DJX1122819 (X'00112203') The attempt to establish an error recovery environment failed.

Cause: The execution of the ESTAE macro call to the operating system failed. This is an internal error.

Action: Check with an authorized IBM service representative to see if a new operating system version was installed that changed the use of the ESTAE macro. Report the problem to an authorized IBM service representative.

DJX1122820 (X'00112204') The operating system is not supported.

Cause: An attempt was made to start the kernel on an unsupported version of the operating system. This is an internal error.

Action: Check with an authorized IBM service representative to see if a new operating system version was installed that does not support programs designed to run on MVS/XA. MVS/370 is currently not supported by the Classic Connect product. MVS/ESA is supported.

DJX1122821 (X'00112205') The time of day clock is in an unreliable state.

Cause: During kernel initialization, the Time-Of-Day clock (TOD) is accessed to store the time value for accounting and statistic purposes. If the CPU indicates an invalid TOD, the TOD clock value is unreliable. This is considered a fatal error. The operating system should react to the invalid TOD state immediately as well.

Action: This is a fatal error for the operating system and should be handled accordingly. For example, it might require an IPL of the operating system.

DJX1122822 (X'00112206') SMF not recording.

Cause: The operating system was IPLed without SMF recording. The Classic Connect is not designed to run in such an environment as it does not allow for accounting.

Action: Re-IPL the operating system using the SMF option.

DJX1122823 (X'00112207') The Virtual memory is exhausted.

Cause: One of the many memory allocations during kernel initialization failed. This is a fatal error at this time.

Action: Restart the kernel with a larger REGION size. Check the configuration parameters for excessive values for the tables. Check the configuration file for excessive size (comments are not read into memory).

DJX1122824 (X'00112208') The component initialization failed.

Cause: During the initialization of the kernel, one of the function calls for special component initialization returned with an error condition. This is a fatal condition at this time.

Action: Because most initialization functions are building and initializing tables and loading parameters and modules, the problem should be related to a shortage in virtual memory. Probable causes are a REGION that is too small or excessive initialization parameter values for the kernel. Check the operating system parameters related to the allocation of virtual storage at JOB, STC, and LOGON initialization time.

DJX1122825 (X'00112209') The BLDL for the Classic Connect load modules failed.

Cause: The execution of the BLDL macro call to the operating system failed. One or more Classic Connect load modules are not in the load library used by the kernel (subsystem).

Action: Refer to the *DataJoiner Classic Connect: MVS Planning, Installation, and Configuration Guide* and make sure that all Classic Connect load modules are in the load library used by the Classic Connect subsystem.

DJX1122832 (X'00112210') The ATTACH for the service tasks failed.

Cause: The execution of the ATTACH macro call to the operating system failed for one of the server tasks. Most likely is that the virtual storage for the kernel is exhausted or that the load module is not in the load library of the kernel. If the kernel ran out of virtual storage, there should be other indications for this situation as well.

Action: Restart the kernel with a larger REGION size. Compare the contents of the kernel's load library with the contents of the distribution tape of the Classic Connectproduct, and restore missing load modules.

**DJX1122833 (X'00112211') The LOG task
initialization routine failed.**

Cause: The internal initialization step for the LOG task failed. This is a fatal error at this time.

Action: Restart the kernel. If the problem persists, restore the load library from the distribution tape. Check the REGION size specified for the kernel, it should be at least 8 MB.

**DJX1122834 (X'00112212') The subsystem
verification failed.**

Cause: The system calls to verify that this is a valid subsystem from the operating system's point of view.

Action: Check with an authorized IBM service representative to see that the name of the procedure for the kernel is put into the parameter library of the operating system as a valid subsystem name.

**DJX1122835 (X'00112213') The cross-memory
environment initialization failed.**

Cause: Either the operating system capacity for cross-memory environments is exhausted or virtual storage is overlaid.

Action: Restart the kernel. If the problem persists, check with an authorized IBM service representativeto see if an IPL of the operating system is necessary.

**DJX1122836 (X'00112214') The "stop subsystem"
request to the Task Monitor failed.**

Cause: The function call to signal the Task Monitor shutdown of the kernel failed.

Action: Ignore this problem because the kernel will shut down anyway. If this message appears frequently, report the problem to an authorized IBM service representative.

**DJX1122837 (X'00112215') The stop request to the
Log task failed.**

Cause: At subsystem shutdown time, the Log task is signaled to close the Log file and stop processing. The Log task did not respond to the signal.

Action: Ignore this problem because the kernel will shut down anyway. If this message appears frequently, report the problem to an authorized IBM service representative.

DJX1122838 (X'00112216') Invalid POST code.

Cause: The Task Monitor posted SI with an invalid post code. This is an internal error.

Action: If the kernel continues executing without errors, ignore the problem. Otherwise, contact an authorized IBM service representative.

**DJX1122839 (X'00112217') ATTACH for subtask
failed.**

Cause: The execution of the ATTACH macro call to the operating system failed. There are many possible reasons for this. The two most likely are that the virtual storage for the kernel is exhausted or that the load module that was supposed to be attached is not in the load library of the kernel. If the kernel ran out of virtual storage, there should be other indications for this situation as well.

Action: Restart the kernel with a larger REGION size. If that is not possible, experiment with the start policy for kernel-tasks as described in the Classic Connect*DataJoiner Classic Connect: MVS Planning, Installation, and Configuration Guide* See information about the TASK INFO ENTRY parameter. If the module is missing, make sure the task name in the TASK INFO ENTRY is specified correctly and the load module of that name is in the access path for the kernel load modules (STEPLIB, LINKLIST).

**DJX1123072 (X'00112300') Normal completion of
function.**

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

**DJX1123073 (X'00112301') LOAD for VHS Vector
Table failed.**

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1123074 (X'00112302') DJX1123845 (X'00112605')

DJX1123074 (X'00112302') The serialization of the Task Monitor failed.

Cause: The Task Monitor processing is partially serialized using a lockword in the active task table. The lock could not be obtained.

Action: This is most likely a performance problem due to too many applications starting at the same time. Restart the application at a time when there are fewer applications competing with initialization.

DJX1123075 (X'00112303') Format error in user name.

Cause: The provided user name is either of length 0 or greater than 8.

Action: Correct the USERID/USERPASSWORD configuration parameter in the API configuration file, and rerun the application.

DJX1123076 (X'00112304') Format error in password.

Cause: The provided password is either of length 0 or greater than 8.

Action: Correct the USERID/USERPASSWORD configuration parameter in the API configuration file, and rerun the application.

DJX1123328 (X'00112400') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1123329 (X'00112401') Access to Vector Table failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1123330 (X'00112402') The serialization of the Task Monitor failed.

Cause: The Task Monitor processing is partially serialized using a lockword in the active task table. The lock could not be obtained.

Action: Ignore this problem because the kernel will shut down anyway.

DJX1123584 (X'00112500') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1123840 (X'00112600') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it is completed without error.

Action: No action is required.

DJX1123842 (X'00112602') Insufficient storage for user exit initialization.

Cause: Not enough virtual memory was available to complete the initialization of the kernel. The error occurred during user exit initialization.

Action: Allocate more virtual memory.

DJX1123843 (X'00112603') User exit initialization was unable to OPEN the DJXLOAD library.

Cause: An error occurred attempting to access the library defined on the DJXLOAD DD statement.

Action: Look in the kernel JCL for the DJXLOAD DD statement and check it for errors.

DJX1123844 (X'00112604') User exit initialization was unable to LOAD the DJXSVVAL user exit.

Cause: An error occurred during an MVS LOAD of the DJXSVVAL load module from the DJXLOAD library.

Action: Check the library to ensure DJXSVVAL is properly placed in the library.

DJX1123845 (X'00112605') The Service Validation Parameter List was found to be corrupted.

Cause: Upon return from the Service Validation user exit, the SVPL parameter list was found to be corrupted.

Action: Check the user exit code for logic errors and invalid use of the parameter list or the register save area.

DJX1123846 (X'00112606') DJX1124354 (X'00112802')

DJX1123846 (X'00112606') The Service Validation Environment Vector was found to be corrupted.

Cause: The storage used to save the user exit environment vector was corrupted.

Action: Contact an authorized IBM service representative.

DJX1123847 (X'00112607') Invalid service name format.

Cause: The format of the service name passed to the user exit interface was invalid.

Action: Report this error to an authorized IBM service representative.

DJX1123848 (X'00112608') Format of the Logon ID is invalid.

Cause: The format of the Logon ID passed to the user exit interface was invalid.

Action: Report this error to an authorized IBM service representative.

DJX1123850 (X'0011260A') User exit initialization was unable to LOAD the DJXACVAL user exit.

Cause: An error occurred during an MVS LOAD of the DJXACVAL load module from the DMSI STEPLIB library.

Action: Check the DMSI JCL and the STEPLIB to ensure DJXACVAL is properly placed in that library.

DJX1123851 (X'0011260B') The Access Validation Parameter List was found to be corrupted.

Cause: Upon return from the Access Validation user exit, the AVPL parameter list was found to be corrupted.

Action: Check the user exit code for logic errors and invalid use of the parameter list or the register save area.

DJX1123852 (X'0011260C') The Access Validation Environment Vector was found to be corrupted.

Cause: The storage used to save the user exit environment vector was corrupted.

Action: Report this error to an authorized IBM service representative.

DJX1124096 (X'00112700') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1124097 (X'00112701') Access to Vector Table failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1124352 (X'00112800') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1124353 (X'00112801') Access to Vector Table failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1124354 (X'00112802') The ECB list provided is not valid.

Cause: The format of the ECB list passed to this function is invalid. This could be a serious problem in the kernel processing.

Action: Restart the kernel if it does not terminate at this time.

DJX1124355 (X'00112803')   DJX1126913 (X'00113201')

DJX1124355 (X'00112803') The serialization of the timer exit routine failed.

Cause: Due to a heavily loaded system, a lockword could not be accessed.

Action: If this problem occurs often, try to reduce the load on the kernel or change the configuration parameters that affect the load such as MAXIMUM NUMBER OF TASKS.

DJX1124356 (X'00112804') The time interval passed to the wait function expired.

Cause: When the wait function is called, a time limit is passed to limit the time in the wait-state. This time limit expired. If this problem occurs often, try to reduce the load on the kernel or change the configuration parameters that affect the load, such as MAXIMUM NUMBER OF TASKS. If this message appears in an external task, the kernel might be too busy or the time interval specified too short.

Action: Check AF WAIT TIME in the client configuration file and LD WAIT TIME in the server configuration file.

DJX1124608 (X'00112900') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1124609 (X'00112901') VHS vector table invalid.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1124610 (X'00112902') The Classic Connect subsystem was not found.

Cause: The subsystem with the name specified in the SUBSYSTEM NAME parameter in the client configuration file is not active. At this time this means that the kernel stopped processing.

Action: Restart the kernel before restarting the application.

DJX1124611 (X'00112903') The target buffer is too small.

Cause: The size of the receiving buffer, for example, the return parameter buffer for the external task, is too small to receive the return parameters. Most likely there is a processing failure within the external task. It also might be a mismatch of the Classic Connect versions in the kernel and the external tasks load libraries.

Action: Check to see if the software level of the Classic Connectprograms running in the kernel and in the external task use different load libraries. Rerun the application.

DJX1124612 (X'00112904') The return code from the cross-memory operation is not valid.

Cause: The resulting return code from a cross-memory operation is unknown to the external task. Possible causes are storage overlay in the external task or a mismatch of the Classic Connectversions in the kernel and the external tasks load libraries.

Action: Check to see that the software level of the Classic Connectprograms running in the kernel and in the external task use different load libraries. Rerun the application.

DJX1126656 (X'00113100') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1126912 (X'00113200') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1126913 (X'00113201') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1126914 (X'00113202') The Classic Connect subsystem was not found.

Cause: The subsystem with the name specified in the SUBSYSTEM NAME parameter in the client configuration file is not active. At this time this means that the kernel stopped processing.

Action: Restart the kernel before restarting the application.

DJX1126915 (X'00113203') The target buffer is too small.

Cause: The size of the receiving data buffer, for example, the one allocated inside the external task, is too small to receive the resulting data from the kernel. Most likely there is a processing failure within the external task.

Action: Stop and rerun the application. If the problem persists, cancel the external task with the DUMP option and forward all error information to an authorized IBM service representative.

DJX1126916 (X'00113204') The return code from the cross-memory operation is not valid.

Cause: The resulting return code from a cross-memory operation is unknown to the external task. Possible causes are storage overlay in the external task or a mismatch of the Classic Connect versions in the kernel and the external tasks load libraries.

Action: Check to see if the software level of the Classic Connectprograms running in the kernel and in the external task use different load libraries. Rerun the application.

DJX1127424 (X'00113400') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1127425 (X'00113401') VHS vector table invalid.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1127426 (X'00113402') The Classic Connect subsystem was not found.

Cause: The subsystem with the name specified in the SUBSYSTEM NAME parameter in the client configuration file is not active. At this time this means that the kernel stopped processing for some reason.

Action: Restart the kernel before restarting the application.

DJX1127427 (X'00113403') The target buffer is too small.

Cause: The size of the receiving data buffer, for example, the one allocated inside the kernel, is too small to receive the data from the external task. Most likely there is a processing failure within the kernel.

Action: Stop the kernel and rerun the application. If the problem persists, cancel the kernel with the DUMP option and forward all error information to an authorized IBM service representative.

DJX1127680 (X'00113500') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1127681 (X'00113501') VHS vector table invalid.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1127682 (X'00113502') The task is not in the Active Task Table.

Cause: The calling task, which is supposed to be the cross-memory service task, is not in the active task table. This is either a serious malfunction of the kernel or an integrity violation. For example, this occurs if someone tries to execute this code without authorization.

Action: Cancel the kernel with the DUMP option and forward all available error information to an authorized IBM service representative. Restart the kernel after replacing the modules in the load library of the kernel with the modules on the Classic Connect product delivery tape.

DJX1127683 (X'00113502') DJX1128450 (X'00113802')

DJX1127683 (X'00113502') The target buffer is too small.

Cause: The size of the receiving parameter buffer, for example, the one for the cross-memory service task, is too small to receive the parameters obtained from the circular buffer. Most likely there is a processing failure within the kernel.

Action: Stop the kernel and rerun the application. If the problem persists, cancel the kernel with the DUMP option and forward all error information to an authorized IBM service representative.

DJX1127936 (X'00113600') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1127937 (X'00113601') VHS vector table invalid.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1127938 (X'00113602') The task is not in the Active Task Table.

Cause: The task number passed to this function by the cross-memory service task is invalid. This could be a serious problem.

Action: Stop the kernel and rerun the application. If the problem persists, cancel the kernel with the DUMP option and forward all error information to an authorized IBM service representative.

DJX1127939 (X'00113603') The target buffer is too small.

Cause: The size of the receiving buffer, for example, the return parameter buffer for the external task, is too small to receive the return parameters. Most likely there is a processing failure within the kernel.

Action: Stop the kernel and rerun the application. If the problem persists, cancel the kernel with the DUMP

option and forward all error information to an authorized IBM service representative.

DJX1127940 (X'00113604') The cross-memory POST of the external task failed.

Cause: The attempt to signal the external task across memory boundaries failed. Most likely the external task terminated in the time gap between issuing the cross-memory POST and the scheduling of the SRB.

Action: Check the processing status of the external task and the reason for its termination. Restart the application.

DJX1127941 (X'00113605') The external task ended.

Cause: While the kernel processed the request from an external task, that task ended processing for some reason.

Action: Check the processing status of the external task and the reason for its termination. Restart the application.

DJX1128448 (X'00113800') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1128449 (X'00113801') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1128450 (X'00113802') The Classic Connect subsystem was not found.

Cause: The subsystem with the name specified in the SUBSYSTEM NAME parameter in the client configuration file is not active. At this time this means that the kernel stopped processing.

Action: Restart the kernel before restarting the application.

DJX1128451 (X'00113803') The circular buffer is full.

Cause: The circular buffer, which is used for communication between kernel tasks and external tasks, filled up faster than the cross-memory service task could process the requests. This might happen if a very large number of applications start at the same time or if the kernel was suspended from processing the incoming requests.

Action: Start your applications at a slower pace or restart the kernel after increasing the value for the kernel configuration parameter CIRCULAR BUFFER SIZE.

DJX1128452 (X'00113804') Invalid function ID (FID).

Cause: Each function of the kernel must perform on behalf of an external task that has an ID. The ID passed to the kernel is invalid. Possible causes are storage overlay in the external task or a mismatch of the Classic Connectversions in the kernel's and the external task's load libraries.

Action: Check to see if the software level of the Classic Connect programs running in the kernel and in the external task use different load libraries. Rerun the application.

DJX1128453 (X'00113805') The return code from the cross-memory operation is not valid.

Cause: The resulting return code from a cross-memory operation is unknown to the external task. Possible causes are storage overlay in the external task or a mismatch of the Classic Connect versions in the kernel and the external tasks load libraries.

Action: Check to see if the software level of the Classic Connect programs running in the kernel and in the external task use different load libraries. Rerun the application.

DJX1128454 (X'00113806') The parameters do not fit into the circular buffer.

Cause: An attempt to place function parameters into the Circular Buffer of the cross-memory service task failed. The parameters are larger than the size of the circular buffer. This means that the external task passed a value that is too high to the kernel. The suspected reasons are either memory overlay in the external task or a mismatch of the Classic Connect code running in the kernel and in the external task.

Action: Check to see if the software level of the Classic Connect programs running in the kernel and in

the external task use different load libraries. Rerun the application.

DJX1128455 (X'00113807') The WAIT on the cross-memory service task timed out.

Cause: The external task waits for a limited time to be posted by the kernel. Because the processing time for the request to the cross-memory service task is relatively short, this error can happen only when the cross-memory POST is scheduled too late. A possible cause is an overloaded operating system.

Action: Restart the application specifying a larger value for EXTERNAL TASK WAIT TIME for the API and the DMSI.

DJX1130752 (X'00114100') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1130753 (X'00114101') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1130754 (X'00114102') An internal error occurred.

Cause: While retrieving information from MVS system control blocks, an inconsistency was found. This problem occurs only when those control blocks are overlaid. This is a fatal situation and cannot be recovered.

Action: Restart the application. If the problem recurs, make sure that the software executing does not overlay system storage, especially the ASCB and related areas.

DJX1130755 (X'00114103') Cross-Memory call failed.

Cause: The attempt to communicate with the Task Monitor inside the kernel failed. This is most likely a performance problem, for example, the Task Monitor is too busy to accept your call. It can also happen if the Task Monitor terminates when it is called.

Action: Rerun the application. If the problem persists, cancel the kernel with the DUMP option and forward the problem to an authorized IBM service representative.

DJX1130756 (X'00114104') DJX1131269 (X'00114305')

DJX1130756 (X'00114104') The Classic Connect kernel (subsystem) is not active.

Cause: The subsystem with the name specified in the SUBSYSTEM NAME parameter in the client configuration file is not active. Clients cannot be started before the kernel is started and completely initialized.

Action: Make sure that the SUBSYSTEM NAME parameter specifies the correct name. If the subsystem is not active, ask your system operator to start it. Wait for the confirmation message DJXSI INIT SUCCESSFUL. Then start your applications.

DJX1131008 (X'00114200') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1131009 (X'00114201') Access to Vector Table failed.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1131010 (X'00114202') An internal error occurred.

Cause: While retrieving information from MVS system control blocks, an inconsistency was found. This problem occurs only when those control blocks are overlaid. This is a fatal situation and cannot be recovered.

Action: Restart the application. If the problem recurs, make sure that the software calling the client didn't overlay system storage, especially the ASCB and related areas.

DJX1131264 (X'00114300') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1131265 (X'00114301') Error during load of VHS vector table.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1131266 (X'00114302') The RACF request was not processed.

Cause: There are several possible reasons for this error:

- RACF is active, but no profile exists for the specified user.
- RACF is not active.
- The MVS router is not active.

Action: For the first two cases, see your systems programmer and forward this information. The most likely reason is that the USERID specified by the client is not known by RACF.

DJX1131267 (X'00114303') The return code passed by RACINIT is invalid.

Cause: This is an internal error.

Action: Forward this message to an authorized IBM service representative.

DJX1131268 (X'00114304') The user profile is not defined to RACF.

Cause: No profile exists for the specified user. The most likely reason is that the USERID specified by the client is not known by RACF.

Action: See your systems programmer or RACF administrator.

DJX1131269 (X'00114305') The PASSWORD is not authorized.

Cause: The PASSWORD specified is not known by RACF.

Action: See your systems programmer or RACF administrator.

DJX1131270 (X'00114306') DJX1131520 (X'00114400')

DJX1131270 (X'00114306') The specified PASSWORD expired.

Cause: The password is valid only for an installation-defined length of time. That time interval expired.

Action: See your systems programmer or RACF administrator to have your access authorization renewed.

DJX1131271 (X'00114307') The new PASSWORD is invalid.

Cause: The format of the new password is invalid for RACF use.

Action: See your systems programmer or RACF administrator for an explanation of valid password formats.

DJX1131272 (X'00114308') The user is not defined to the group.

Cause: The client is trying to access a group that they are not part of.

Action: See your systems programmer or RACF administrator for proper group authorization.

DJX1131273 (X'00114309') The installation exit routine failed the RACINIT request.

Cause: The verification request for the specified user (client) failed inside the installation exit routine.

Action: See your systems programmer or RACF administrator and provide this error information.

DJX1131280 (X'00114310') The user access was revoked.

Cause: The access of the specified user to the system was revoked.

Action: See your systems programmer or RACF administrator, and provide this error information.

DJX1131281 (X'00114311') RACF is not active.

Cause: At the time of this request, RACF was not active.

Action: See your systems programmer, console operator, or RACF administrator.

DJX1131282 (X'00114312') The access to the group was revoked.

Cause: Your RACF group access was revoked.

Action: See your systems programmer, console operator, or RACF administrator.

DJX1131283 (X'00114313') An OIDCARD is required but not supplied.

Cause: Use of this terminal requires an operator ID card. No card was provided.

Action: See your systems programmer or RACF administrator.

DJX1131284 (X'00114314') The OIDCARD is invalid for specified user.

Cause: The operator ID card is not authorized for this user.

Action: See your systems programmer or RACF administrator.

DJX1131285 (X'00114315') The current user is not authorized to use the terminal.

Cause: The use of this terminal is restricted to certain users.

Action: See your systems programmer or RACF administrator.

DJX1131286 (X'00114316') The user is not authorized to use the application.

Cause: The use of this application is restricted to certain users.

Action: See your systems programmer or RACF administrator.

DJX1131520 (X'00114400') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1131521 (X'00114401') DJX1131778 (X'00114502')

DJX1131521 (X'00114401') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1131522 (X'00114402') The subsystem name in the configuration file is longer than four characters.

Cause: Subsystem names longer than four characters are invalid.

Action: Correct the subsystem name specified in the configuration file of the kernel.

DJX1131523 (X'00114403') The Virtual memory is exhausted.

Cause: One of the many memory allocations during kernel initialization failed. This is a fatal error at this time.

Action: Restart the kernel with a larger REGION size. Check the configuration parameters for excessive values for the tables. Check the configuration file for excessive size (comments are not read into memory).

DJX1131524 (X'00114404') The subsystem verification call failed.

Cause: The system call to verify that this is a valid subsystem from the operating system's point of view failed.

Action: This is an internal error. Check with an authorized IBM service representative to see if a new operating system version was installed, or if the operating system does not support or has changed the use of the IEFSSREQ macro. Report the problem to an authorized IBM service representative.

DJX1131525 (X'00114405') The subsystem name is invalid.

Cause: The system call to verify that this is a valid subsystem from the operating system's point of view failed.

Action: Check with an authorized IBM service representative to see if the name of the procedure for the kernel was put into the parameter library of the operating system as a valid subsystem name.

DJX1131526 (X'00114406') The return code in SSOBRETN is not zero.

Cause: After successful subsystem verification, an internal data field does not contain the expected value of zero. This is an internal error.

Action: Check with an authorized IBM service representative to see if a new operating system version was installed that changed the use of the IEFSSREQ macro. Report the problem to an authorized IBM service representative.

DJX1131527 (X'00114407') Another subsystem of the same name is already active.

Cause: A subsystem with the same name as the one you started is already active.

Action: Either stop the subsystem that is already active before starting one of the same name or use the subsystem that is already active. Before the start command for a Classic Connect subsystem can be issued, a running subsystem of the same name must issue the message DJXSI TERMINATION COMPLETE.

DJX1131776 (X'00114500') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1131777 (X'00114501') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1131778 (X'00114502') The pointer to the SSCVT is invalid.

Cause: This is a serious internal error. Either virtual storage is overwritten or an unauthorized code is executing.

Action: Restart the kernel. If the problem persists, restore the kernel's load library from the Classic Connect product distribution tape.

DJX1131779 (X'00114503') The CSA memory is full.

Cause: There is no room in the CSA for the 80-byte large SubSystem Block (SSB).

Action: Check with an authorized IBM service representative to see if the size of the CSA can be increased, which requires an IPL of the operating system. If that is not possible, wait until a time when there is less load on the operating system and restart the kernel.

DJX1131780 (X'00114504') Another subsystem of the same name initializes faster.

Cause: If two Classic Connect subsystems (kernels) are started at the same time, they are competing for the allocation of the SubSystem Block (SSB). The faster one wins and the second one cannot initialize. There can be only one kernel active with the same name at a time.

Action: Avoid starting a kernel of the same name twice.

DJX1131781 (X'00114505') Another subsystem of the same name is already active.

Cause: A subsystem with the same name as the one you started is already active.

Action: Either stop the subsystem that is already active before starting one of the same name, or use the subsystem that is already active. Before the start command for a Classic Connectsubsystem can be issued, a running subsystem of the same name must issue the message DJXSI TERMINATION COMPLETE.

DJX1131782 (X'00114506') The Virtual memory is exhausted.

Cause: One of the many memory allocations during kernel initialization failed. This is a fatal error at this time.

Action: Restart the kernel with a larger REGION size. Check the configuration parameters for excessive values for the tables. Check the configuration file for excessive size (comments are not read into memory).

DJX1131783 (X'00114507') The Entries Table (ET) cannot be built.

Cause: This is an internal error in SI. It can be due to storage overlay or corrupted code.

Action: Replace the kernel's load library with the modules from the Classic Connectproduct distribution tape. If the problem persists, report it to an authorized IBM service representative.

DJX1132032 (X'00114600') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

DJX1132033 (X'00114601') VHS vector table not found.

Cause: This is a fatal error. Access to the vector table failed.

Action: Contact an authorized IBM service representative.

DJX1132035 (X'00114603') The Session Table reset failed.

Cause: The cleanup for the Router Session Table failed.

Action: The kernel will shut down after this message and must be restarted.

DJX1132037 (X'00114605') The port handle could not be obtained.

Cause: The messaging system was unable to provide a new port handle for the router. For each restarted task there must be a port handle obtained to enable the messaging system to send and receive messages on behalf of that task. If the port handle cannot be obtained, the new task cannot be initialized.

Action: The kernel will shut down after this message and must be restarted.

DJX1132038 (X'00114606') The ATTACH for the router task failed.

Cause: The execution of the ATTACH macro call to the operating system failed. There are many possible reasons for this. The most likely is that the virtual storage for the kernel is exhausted or corrupted.

Action: The kernel will shut down after this message and must be restarted.

DJX1132288 (X'00114700') Normal completion of function.

Cause: This is an informational message identifying the function and confirming that it completed without error.

Action: No action is required.

Chapter 29. AIX (API-F) Error Messages

The following messages are related to AIX API-F errors.

DJX1179650 (X'00120002') Session with API was rejected/terminated.

Cause: This message occurs because either the API is not accepting service requests at this time, or the local Classic Connectkernel is in a shutdown state. There are several reasons why a kernel may not respond: when the kernel is shut down; when the APIB is not spawned due to system resources shortages; and, when the task table is full. You can change the MAX Number of Tasks parameter, when the task table is full, to eliminate that possibility.

Action: Wait for the Classic Connect kernel to come back on line, or contact an authorized IBM service representative.

DJX1179651 (X'00120003') Did not get client-router-service subsystem header.

Cause: A protocol error occurred during initialization of communications within the API.

Action: Contact an authorized IBM service representative.

DJX1179652 (X'00120004') Did not get API-DMSI message subsystem header.

Cause: A protocol error occurred during initialization of communications within the API.

Action: Contact an authorized IBM service representative.

DJX1179653 (X'00120005') Did not get session subsystem header.

Cause: A protocol error occurred during initialization of communications within the API.

Action: Contact an authorized IBM service representative.

DJX1179654 (X'00120006') API received an invalid message type.

Cause: A message was received by the API, which had an unknown message type.

Action: Contact an authorized IBM service representative.

DJX1179655 (X'00120007') An SQLDA was passed for an SQL statement that does not accept SQLDAs for an input parameters.

Cause: An invalid parameter list was passed for an embedded SQL statement. This is due to improper precompilation.

Action: Contact an authorized IBM service representative.

DJX1179656 (X'00120008') A HostVar was passed for an SQL statement that does not accept HostVar for input parameters.

Cause: An invalid parameter list was passed for an embedded SQL statement. This is due to improper precompilation.

Action: Contact an authorized IBM service representative.

DJX1179657 (X'00120009') Results failed to decode.

Cause: The results of the query failed to decode due to either a memory allocation error or an internal structural inconsistency.

Action: Attempt to reexecute the SQL statement. If the error persists, contact an authorized IBM service representative.

DJX1179658 (X'0012000A') CI rejected the message.

Cause: A transport layer error caused the message to be returned to the API.

Action: Attempt to reexecute the SQL statement. If the error persists, contact an authorized IBM service representative.

DJX1179666 (X'00120012') DJX1179667 (X'00120013')

DJX1179666 (X'00120012') Conversion between Hex and Binary strings resulted in an incorrect length.

Cause: Conversion between the Hex and Binary strings did not yield an appropriate value.

Action: Examine the application source that provided the hexadecimal string.

DJX1179667 (X'00120013') The maximum number of CICS transactions are currently active.

Cause: A Classic Connect-enabled CICS transaction was started when the maximum number of such transactions were already active. This number is set by the NUMTHRDS parameter on the DJXTHRDS macro.

Action: Restarting the transaction will resolve the problem if the system was temporarily saturated. If the message recurs, it indicates the need to increase the value of the NUMTHRDS parameter.

Chapter 30. Universal Coordinated Time Error Messages

The following messages are related to universal coordinated time errors.

DJX1245184 (X'00130000') Unable to convert calendar time to universal coordinated time.

Cause: Unable to get a system time stamp in ASN.1 format.

Action: Contact an authorized IBM service representative.

Chapter 31. SQL Syntax Error Messages

The following messages are related to SQL syntax errors.

DJX1376257 (X'00150001') Syntax error in SQL statement.

Cause: A syntax error was detected during the parsing of an SQL statement.

Action: Correct the syntax error in the SQL statement.

Chapter 32. Table Correlation Semantic Error Messages

The following messages are related to table correlation semantic errors.

DJX1507329 (X'00170001') Incorrect use of table name.

Cause: The same table name appears more than once in a FROM clause.

Action: Ensure that no table name appears more than once within a given FROM clause unless each occurrence of the name has a unique correlation name.

DJX1507330 (X'00170002') Table not defined.

Cause: A column reference has a qualifier that must be a table name but is not declared in the FROM clause of the same subquery where the reference occurs.

Action: Declare the table name in the FROM clause of the same subquery where the column reference occurs.

DJX1507331 (X'00170003') Qualifier not defined.

Cause: A qualifier of a column reference is not defined either as a table, view name, or correlation name.

Action: Ensure that a column qualifier is declared in the FROM clause of a syntactically visible subquery.

DJX1507332 (X'00170004') Identical table and correlation names.

Cause: Within a given FROM clause, a table name and a correlation name are the same.

Action: Ensure that no table names are the same as any correlation names within the same FROM clause.

DJX1507333 (X'00170005') Multiple use of same correlation name.

Cause: Two or more tables within a single FROM clause have identical correlation names.

Action: Ensure that all correlation references within a given FROM clause are unique.

Chapter 33. Cursor Subsystem Error Messages

The following messages are related to cursor subsystem errors.

DJX1572865 (X'00180001') Multiple declarations of a cursor.

Cause: An attempt was made to provide multiple declarations for a single cursor name.

Action: Reconsider the cursor naming strategy within the module.

DJX1572866 (X'00180002') Undeclared cursor in OPEN statement.

Cause: An attempt was made to open a cursor that was not declared.

Action: Ensure that the OPEN statement refers to a declared cursor.

DJX1572867 (X'00180003') Undeclared cursor in FETCH statement.

Cause: An attempt was made to fetch a cursor that was not declared.

Action: Ensure that the FETCH statement refers to a declared cursor.

DJX1572868 (X'00180004') Undeclared cursor in CLOSE statement.

Cause: An attempt was made to close a cursor that was not declared.

Action: Ensure that the CLOSE statement refers to a declared cursor.

Chapter 34. Memory Diagnostic Message

The following message is related to memory diagnosis.

DJX1638401 (X'00190001') Insufficient memory.

Cause: An attempt was made to dynamically allocate main memory, but none was available.

Action: Increase the memory region available to the application or reduce the data requirements.

Chapter 35. I/O Diagnostic Message for NL Catalog

The following message is related to I/O diagnosis
for the NL catalog.

DJX1703937 (X'001A0001') Cannot open I/O file.

Cause: An attempt was made to open an I/O file that
does not exist or cannot be opened.

Action: Check that the file name specified in the
application program exists, and correct any errors. If
valid files are correctly specified in the application, verify
that the application has authorization to access the files.

Chapter 36. ASN.1 Diagnostic Message for NL Catalog

The following message is related to ASN.1
diagnosis for the NL catalog.

DJX1769473 (X′001B0001′) ASN invalid encode.

Cause: This is an internal error.

Action: Contact an authorized IBM service
representative.

Chapter 37. Session Establishment Error Message

The following message is related to session establishment errors.

DJX2031616 (X'001F0000') Session establishment message timed out.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

439

Chapter 38. Communications Interface Error Messages

The following messages are related to communication interface errors.

DJX2097152 (X'00200000') CI operation is not yet implemented.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097153 (X'00200001') Internal error detected by CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097154 (X'00200002') Invalid memory location or size detected by CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097155 (X'00200003') Insufficient memory is available for CI to perform operation.

Cause: CI is unable to allocate sufficient memory for operation. There might be too many entries in the Node Address table for this CI.

Action: Try reducing the number of Node Address table entries. Contact an authorized IBM service representative.

DJX2097156 (X'00200004') Incorrect size of table detected in CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097157 (X'00200005') Initialization of CI failed.

Cause: CI is unable to perform some portion of its initialization process, and is therefore, unable to begin operation. This is a nonrecoverable error and will cause the CI task to exit.

There are generally three kinds of problems that cause the CI to fail initialization:

- CI is incorrectly configured.
- CI is unable to properly access the transport layer (in this case, the TCP/IP server).
- CI is unable to obtain the required resources, such as memory for internal control blocks.

This message is always associated with another message giving a more specific cause of the problem.

Action:

DJX2097158 (X'00200006') Time limit expired while CI was waiting for transmission to complete.

Cause: CI was unable to complete transmission of data. Data communications link night be down.

Action: Check status of data communications links and restart as necessary.

DJX2097159 (X'00200007') CI attempted to send or receive empty data packet.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097160 (X'00200008') Invalid trace value for CI found in System Configuration Table.

Cause: A CI trace value was outside of the range 1 ≤ TCP/IP CI TRACE LEVEL ≤ 20.

Action: Correct the system configuration file.

DJX2097161 (X'00200009') CI is incorrectly configured.

Cause: The configuration file is incorrect for this CI. This message should be accompanied by another message describing the problem in more detail.

Action:

DJX2097162 (X'0020000A') Incorrect time limit found for CI in System Configuration Table.

Cause: Unable to read TCP/IP_CI_TIME_LIMIT value from system configuration file.

Action: Correct the system configuration file.

DJX2097165 (X'0020000D') Invalid table reference detected by CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative

DJX2097167 (X'0020000F') CI is unable to send message or datagram.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097168 (X'00200010') CI is unable to switch message or datagram.

Cause: CI received a message from the network but was unable to forward it to a destination on the Classic Connect node. The destination might not be currently valid.

Action: Check the MVS operator console for abending task. If found, the user application, and possibly the Classic Connectnode, might need to be restarted.

DJX2097170 (X'00200012') CI attempted to transmit local message to local destination.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097172 (X'00200014') Invalid internal state detected by CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097173 (X'00200015') CI is unable to process message.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097175 (X'00200017') Invalid parameter detected by CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097176 (X'00200018') Invalid time value detected by CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097179 (X'0020001B') CI received a data packet for an unsupported protocol.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097180 (X'0020001C') CI received a data packet or message with an incomplete header.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097181 (X'0020001D') CI found an Event Timing Block with incomplete or incorrect data.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative

DJX2097184 (X'00200020') CI is unable to allocate a control block.

Cause: This is an informational message.

Action: No action is required.

DJX2097185 (X'00200021') DJX2097196 (X'0020002C')

DJX2097185 (X'00200021') CI found an unexpectedly empty data structure field.

Cause: This is an informational message.

Action: No action is required.

DJX2097186 (X'00200022') CI was unable to obtain requested resource from CI Resource File.

Cause: The CI requested message was not in a resource file. The message number will be displayed without text.

Action: Contact an authorized IBM service representative.

DJX2097187 (X'00200023') CI was unable to recognize provided specific return code value.

Cause: This is an informational message.

Action: No action is required.

DJX2097188 (X'00200024') CI detected end-of-file.

Cause: This is an informational message.

Action: No action is required.

DJX2097190 (X'00200026') CI detected an empty queue.

Cause: This is an informational message.

Action: No action is required.

DJX2097191 (X'00200027') CI unable to identify host submitting connection request.

Cause: The connection attempt was made from an unrecognized CI. This is probably a configuration file error.

Action: Check the Node Address Table entries for TCP/IP CIs, both locally and at all nodes to which CI will connect.

DJX2097192 (X'00200028') CI detected IO error while attempting to send or receive data.

Cause: This is an informational message.

Action: No action is required.

DJX2097193 (X'00200029') CI detected Node Address entry for same node.

Cause: The Classic Connect system configuration file has an entry for a TCP/IP CI at the same node as the Task Info Entry.

Action: Correct the configuration file and restart the Classic Connect node.

DJX2097194 (X'0020002A') CI found Node Address entries in Classic Connect configuration file.

Cause: The Classic Connect system configuration file has no entries for CI.

Action: Correct the configuration file and restart the Classic Connect node.

DJX2097195 (X'0020002B') CI found no Task Info entries in the Classic Connect configuration file.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX2097196 (X'0020002C') CI detected wrong number of fields specified in Node Address Entry.

Cause: The Classic Connect system configuration file has an incorrect Node Address Entry.

Action: Correct the configuration file, and restart the Classic Connect node.

Chapter 39. IMS Error Messages

The following messages are related to IMS errors.

DJX2228225 (X'00220001') PCB list error detected during DMSI initialization.

Cause: The PSB passed to the IMS DMSI does not have an I/O PCB.

Action: Reassemble the PSB with the CMPAT=YES parameter on the PSBGEN statement or use a PSB generated by the PSB Generator (PSBG). Contact an authorized IBM service representative for more information about PSBG.

DJX2228226 (X'00220002') All PCBs in use.

Cause: Too many cursors opened for the same database.

Action: Add PCBs to the PSB or use a PSB generated by the PSB Generator (PSBG) utility provided with the IMS DMSI.

DJX2228227 (X'00220003') Error converting IMS data to an SQL data type.

Cause: An error occurred while attempting to convert a constant in a WHERE clause from SQL to IMS format.

Action: Verify that the Projection Utility grammar syntax for the columns referenced in the WHERE clause is correct.

DJX2228228 (X'00220004') Cannot access a PCB for database requested.

Cause: Either all PCBs for the database mapped for the table being opened are in use or the PSB is defined incorrectly.

Action: Ensure that the PSB is correctly defined. Rerun the PSB Generator (PSBG) utility if necessary.

DJX2228229 (X'00220005') Cannot access a PCB with required segments.

Cause: At least one PCB was found for the correct database, but no PCB was sensitive to all segments mapped by the table being opened.

Action: Ensure that the PSB is correctly defined. Rerun the PSB Generator (PSBG) if necessary.

DJX2228230 (X'00220006') Requested storage not available.

Cause: A GETMAIN failed.

Action: Increase the region size as appropriate.

DJX2228231 (X'00220007') Unexpected status code was received.

Cause: A DL/I call issued by the IMS DMSI received an unexpected status code.

Action: Check the second four bytes of SQLEXT in the SQLCA for the IMS status code associated with the call. Examine the DMSI output for messages. Change the VSPEC parameter in the IMS DMSI to TR to activate DLITRACE. Examine the DFSTROUT data set output to identify the failing DL/I call.

DJX2228232 (X'00220008') Attempt to open 2nd cursor on same HSAM DB.

Cause: The application attempted to open more than one cursor on one or more tables mapping the same HSAM physical database. Because IMS does not support multiple PCBs and/or multiple positioning in HSAM databases, the Classic Connect DMSI is forced to reject the request.

Action: This restriction can be eliminated by converting the database to a different IMS database organization.

DJX2228233 (X'00220009') An invalid field token was detected.

Cause: The IMS DMSI specified an invalid token when attempting to retrieve a long string column from intermediate storage. This is an internal error.

Action: Save all relevant output associated with this failure and contact an authorized IBM service representative for technical assistance.

DJX2228234 (X'0022000A') More than one XDFLD in WHERE clause.

Cause: While parsing the WHERE clause, the IMS DMSI detected that more than one condition referenced an XDFLD.

Action: Refer to the section detailing support for secondary Indices in the *IMS Programming Supplement* for information about proper XDFLD usage in Classic Connect applications.

DJX2228235 (X'0022000B')   DJX2228235 (X'0022000B')

DJX2228235 (X'0022000B') No PCB with correct PROCSEQ.

Cause: The application specified an XDFLD column in its WHERE clause. However, none of the PCBs in the PSB specified in the IMS DMSI had a PROCSEQ parameter pointing to the Secondary Index DBD, which contains the XDFLD.

Action: Ensure that the PSB is correctly defined. Rerun the PSB Generator utility if necessary.

Chapter 40. VSAM System Error Messages

The following messages are related to VSAM errors.

DJX2293760 (X'00230000') Cannot establish position within Set.

Cause: The VSAM POINT macro failed.

Action: Check the second four bytes of SQLEXT in the SQLCA for the VSAM reason code associated with the call.

DJX2293761 (X'00230001') Cannot allocate memory below 16 MB.

Cause: A GETMAIN call failed when trying to acquire memory below 16 MB.

Action: Increase the region size.

DJX2293762 (X'00230002') Error accessing VSAM catalogs.

Cause: The SHOWCAT macro failed.

Action: Check the second four bytes of SQLEXT in the SQLCA for the return code returned from the SHOWCAT macro. Refer to IBM's *Catalog Administration Guide* for more information regarding return codes returned from the SHOWCAT macro.

DJX2293763 (X'00230003') Cannot allocate data set.

Cause: SVC 99 failed trying to allocate a data set.

Action: Check the second four bytes of SQLEXT in the SQLCA for the reason code associated with the SVCC 99 call and take appropriate action. Refer to IBM's *System Programming Library: System Macros & Facilities Volume 1* for more information regarding SVC99 reason codes.

DJX2293764 (X'00230004') Cannot acquire information from VSAM data set.

Cause: SVC 99 failed trying to acquire information about the status of the data set.

Action: Check the second four bytes of SQLEXT in the SQLCA for the reason code associated with the SVC 99 call and take appropriate action. Refer to IBM's *System Programming Library: System Macros & Facilities Volume 1* for more information regarding SVC 99 reason codes.

DJX2293765 (X'00230005') Cannot open VSAM data set.

Cause: The VSAM OPEN macro failed.

Action: Check the second four bytes of SQLEXT in the SLCA for the reason code associated with the OPEN failure and take appropriate action.

DJX2293766 (X'00230006') Retrieval of VSAM record failed.

Cause: The VSAM GET macro failed.

Action: Check the second four bytes of SQLEXT in the SQLCA for the reason code associated with the GET failure and take appropriate action.

Chapter 41. VRS Error Messages

The following messages are related to query processor interface errors.

DJX3342337 (X'00330001') Cannot open table

Cause: An attempt to open a table in the data dictionary failed.

Action: Check that the table name is specified correctly in your application program and that the specified table exists in the data dictionary.

DJX3342338 (X'00330002') Invalid column definition.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342339 (X'00330003') Insufficient memory.

Cause: An attempt was made to dynamically allocate main memory, but none was available.

Action: Increase the memory region available to the application, or reduce the data requirements.

DJX3342340 (X'00330004') Cannot read table.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342341 (X'00330005') Data dictionary not found.

Cause: An attempt to access a data dictionary failed because the data dictionary named does not exist.

Action: Correct the name of the data dictionary specified in your application program.

DJX3342342 (X'00330006') Cannot scan table.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342343 (X'00330007') Table descriptor not found.

Cause: An attempt to locate a table failed because a table descriptor was missing.

Action: Include an appropriate table descriptor in the application program.

DJX3342345 (X'00330009') Cannot scan table.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342346 (X'0033000A') Invalid database name.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342347 (X'0033000B') Database already open.

Cause: An attempt was made to open a database that was already open.

Action: Issue a CLOSE statement to free the database.

DJX3342348 (X'0033000C') Memory allocation error (VRSDA).

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not correct the error, contact an authorized IBM service representative.

DJX3342349 (X'0033000D') Memory allocation error (VRSVAR).

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not correct the error, contact an authorized IBM service representative.

DJX3342351 (X'0033000F') Cannot open temporary file.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342353 (X'00330011') Cannot close temporary file.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342354 (X'00330012') Table closed.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342355 (X'00330013') Cannot reset temporary file.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342356 (X'00330014') Invalid data area.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342357 (X'00330015') Empty data area.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342359 (X'00330017') Incomplete FETCH

Cause: A FETCH statement is not completely specified.

Action: Edit the statement to correct the error. Refer to the *DB2 SQL Reference* for MVS/ESA for FETCH statement syntax.

DJX3342360 (X'00330018') Invalid FETCH record.

Cause: A FETCH statement is incorrectly specified.

Action: Edit the statement to correct the error. Refer to the *DB2 SQL Reference* for MVS/ESA for FETCH statement syntax.

DJX3342361 (X'00330019') Invalid write record.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342362 (X'0033001A') Unsupported data type.

Cause: A specified data type is not supported by the Classic ConnectDMSI in use.

Action: Refer to the appropriate DMSI supplements for information about supported data types.

DJX3342363 (X'0033001B') Cannot allocate table

Cause: An attempt was made to dynamically allocate memory for a table, but none was available.

Action: Increase the memory region available to the application or reduce the data requirements.

DJX3342364 (X'0033001C') Cannot allocate data dictionary.

Cause: An attempt was made to dynamically allocate memory for a data dictionary, but none was available.

Action: Increase the memory region available to the application or reduce the data requirements.

DJX3342365 (X'0033001D') Cannot open file.

Cause: An attempt to open a file failed.

Action: Check that the file exists and that the file name is correctly specified. If the existing file was correctly specified, contact an authorized IBM service representative.

DJX3342366 (X'0033001E') Invalid I/O mode.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342367 (X'0033001F') DJX3342380 (X'0033002C')

DJX3342367 (X'0033001F') Missing VRSDA

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342368 (X'00330020') Data dictionary missing.

Cause: An attempt to open a data dictionary failed because the data dictionary was not found.

Action: Check that the data dictionary specified by the DMSI configuration file is the same as that created by the projection utility.

DJX3342369 (X'00330021') Data area missing.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342370 (X'00330022') Data area invalid.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342371 (X'00330023') Cannot allocate auxiliary table.

Cause: An attempt was made to dynamically allocate memory for an auxiliary table, but none was available.

Action: Increase the memory region available to the application or reduce the data requirements.

DJX3342372 (X'00330024') Cannot allocate cursor.

Cause: An attempt was made to dynamically allocate memory for a cursor, but none was available.

Action: Increase the memory region available to the application or reduce the data requirements.

DJX3342373 (X'00330025') Table not in data dictionary.

Cause: A table specified in the application program is not in the data dictionary.

Action: Edit the application to include current table names.

DJX3342374 (X'00330026') Unexpected information found in auxiliary column.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342375 (X'00330027') Cannot allocate auxiliary column.

Cause: An attempt was made to dynamically allocate memory for an auxiliary column, but none was available.

Action: Increase the memory region available to the application or reduce the data requirements.

DJX3342376 (X'00330028') Column not in data area.

Cause: A column specified in the application program is not in the designated data area.

Action: Edit the application to include current column descriptions.

DJX3342377 (X'00330029') Data dictionary table missing.

Cause: A data dictionary table cannot be found.

Action: Check that the data dictionary was created and correctly named in the application program.

DJX3342378 (X'0033002A') Data dictionary column missing.

Cause: A data dictionary column cannot be found.

Action: Check that the data dictionary was created and correctly named in the application program.

DJX3342379 (X'0033002B') Memory allocation error: internal.

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not correct the problem, contact an authorized IBM service representative.

DJX3342380 (X'0033002C') Memory allocation error: internal.

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not correct the problem, contact an authorized IBM service representative.

DJX3342381 (X'0033002D') Memory allocation error: internal.

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not correct the problem, contact an authorized IBM service representative.

DJX3342382 (X'0033002E') Bad record allocation.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342383 (X'0033002F') Incomplete record read.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342384 (X'00330030') Comparison not supported.

Cause: An invalid comparison occurred between operands.

Action: Correct the comparison in the program and retry.

DJX3342385 (X'00330031') Retrieval error: internal.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342386 (X'00330032') Memory allocation error: internal.

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not correct the error, contact an authorized IBM service representative

DJX3342387 (X'00330033') Sort error: internal.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342388 (X'00330034') Invalid sort record length.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3342389 (X'00330035') Duplicate column name.

Cause: The same column name appears more than once in a FROM clause.

Action: Ensure that no column name appears more than once within a given FROM clause unless each occurrence of the name has a unique correlation name.

DJX3342390 (X'00330036') Duplicate table name.

Cause: The same table name appears more than once in a FROM clause.

Action: Ensure that no table name appears more than once within a given FROM clause unless each occurrence of the name has a unique correlation name.

Chapter 42. Query Compiler Messages

The following messages are related to network errors.

DJX3407873 (X'00340001') Cannot open cursor.

Cause: An OPEN statement against a cursor failed.

Action: Check that the cursor name is correctly specified and correct any errors. Check that the cursor was not previously opened by the application without subsequently being closed. Verify that the application did not exceed the maximum number of simultaneously open cursors. For the limit on simultaneously-open cursors, see the appropriate cursors, and refer to the appropriate DMSI chapter in the *MVS Installation and Configuration Guide*

DJX3407874 (X'00340002') Cannot allocate FETCB.

Cause: An attempt to allocate memory failed.

Action: Increase the region size. If this does not correct the error, contact an authorized IBM service representative.

DJX3407875 (X'00340003') Cannot allocate SSCB.

Cause: An attempt to allocate memory failed.

Action: Increase the region size. If this does not correct the error, contact an authorized IBM service representative.

DJX3407876 (X'00340004') Invalid parse tree.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407877 (X'00340005') Ambiguous column reference.

Cause: A column name specified in an SQL statement is a valid column name on more than one table.

Action: Edit the SQL statement to include a table name or correlation name for the specified column.

DJX3407878 (X'00340006') Column reference does not exist.

Cause: One of the following conditions occurred:

- A column name specified in the select list of a SELECT statement does not refer to any column on the tables specified in the FROM clause of the statement.
- A correlation reference exists in a GROUP BY clause.
- An unresolved qualifying reference exists in a HAVING clause.

Action:

DJX3407879 (X'00340007') Invalid column qualifier.

Cause: The column name specified in a SELECT statement does not match any column in the data dictionary.

Action: Edit the SELECT statement to include a valid column name.

DJX3407880 (X'00340008') Cannot open code segment.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407881 (X'00340009') Error writing to segment.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407882 (X'0034000A') Memory allocation error: internal.

Cause: An attempt to allocate memory or allocate temporary file space occurred.

Action: Either increase the region size of the DMSI or change the value to the LD TEMP SPACE configuration parameter in the DMSI configuration file.

DJX3407883 (X'0034000B') Memory allocation error: internal.

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not correct the error, contact an authorized IBM service representative.

DJX3407884 (X'0034000C') Cannot allocate a reference offset.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407885 (X'0034000D') Cannot allocate entry.

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not correct the error, contact an authorized IBM service representative.

DJX3407886 (X'0034000E') Duplicate SMBL definition.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407887 (X'0034000F') Invalid entry mode.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407888 (X'00340010') Unresolved reference.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407889 (X'00340011') Cannot open data map table.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407890 (X'00340012') Memory allocation error (object module).

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not correct the error, contact an authorized IBM service representative.

DJX3407891 (X'00340013') Invalid instruction.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407892 (X'00340014') Error reading from segment.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407893 (X'00340015') Cannot disassemble instruction.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407894 (X'00340016') Normal return from query.

Cause: This is an informational message.

Action: No action is required.

DJX3407895 (X'00340017') Cannot open data segment.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407897 (X'00340019') Invalid segment mode.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407898 (X'0034001A') DJX3407913 (X'00340029')

DJX3407898 (X'0034001A') Invalid column grouping.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407899 (X'0034001B') SQLDA DESCRIBE failed.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407900 (X'0034001C') Unsupported data type.

Cause: A specified data type is not supported by the Classic Connect DMSI in use.

Action: Refer to the appropriate DMSI supplement for a list of supported data types.

DJX3407901 (X'0034001D') Table named does not exist.

Cause: A table name specified in a SELECT statement is not present in the data dictionary.

Action: Edit the SELECT statement to include a valid table name.

DJX3407903 (X'0034001F') SQL operator does not exist.

Cause: An operator specified in an SQL statement is incorrect.

Action: Edit the statement to correct the error.

DJX3407904 (X'00340020') Operand is incorrect.

Cause: An operand in a statement is used incorrectly.

Action: Edit the statement to correct the error.

DJX3407905 (X'00340021') Invalid number of result columns in the select list.

Cause: The number of result columns in a subquery does not match the syntax.

Action:

DJX3407906 (X'00340022') Storage class is incorrect.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407907 (X'00340023') Data type not supported.

Cause: A specified data type is not supported by the Classic Connect DMSI in use.

Action: Refer to the appropriate DMSI supplement for a list of supported data types.

DJX3407908 (X'00340024') DB2 type not supported.

Cause: A specified DB2 data type is not supported by the Classic Connect DMSI in use.

Action: Refer to the appropriate DMSI supplement for a list of supported data types.

DJX3407910 (X'00340026') Incompatible data type.

Cause: The data types in the target database are incompatible with the corresponding data types specified in the relational description in the data dictionary.

Action: The database administrator must update the data dictionary with the corrected data types.

DJX3407911 (X'00340027') Invalid opcode.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407912 (X'00340028') Table type incorrect.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407913 (X'00340029') Internal data type not supported.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407914 (X'0034002A') Scalar function not supported.

Cause: The scalar function is not currently supported.

Action: Delete the scalar function from the program and retry.

DJX3407915 (X'0034002B') Scalar functions not supported.

Cause: Some scalar functions are not supported by the Classic Connect.

Action: Remove the function from the SQL statement.

DJX3407916 (X'0034002C') Invalid number of operands.

Cause: The number of operands in an SQL statement does not match the syntax for that statement.

Action: Change the number of operands to match the syntax.

DJX3407917 (X'0034002D') Invalid value class.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407919 (X'0034002F') Insufficient memory.

Cause: An attempt was made to dynamically allocate main memory, but none was available.

Action: Increase the memory region available to the application or reduce the data requirements.

DJX3407920 (X'00340030') Variables in UNION clause mismatched.

Cause: At least one pair of columns matched by the UNION clause has incompatible data types.

Action: Review the SELECT statements joined by the UNION clause and correct any errors. Refer to the DB2 *Application and Programming and SQL Guide* for MVS/ESA and to the *DB2 SQL Reference* for MVS/ESA for more information on the UNION clause.

DJX3407921 (X'00340031') Union clause error: data areas incorrectly sized.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407922 (X'00340032') Invalid data type combination.

Cause: Data types matched by the application program cannot be combined. This error can occur if the UNION clause joining SELECT statements attempts to match incompatible data types.

Action: Check the matched data types and correct any errors.

DJX3407923 (X'00340033') UNION clause error: decimal precision too large.

Cause: The precision specified for a decimal type is greater than 15.

Action: Correct the error by specifying a lower precision value. Refer to the DB2 *Application and Programming and SQL Guide* for MVS/ESA and to the *DB2 SQL Reference* for MVS/ESA for more information.

DJX3407924 (X'00340034') Negative decimal scale error.

Cause: The scale specified for a decimal data type is a negative value.

Action: Change the scale to a positive value. Refer to the *DB2 Application and Programming and SQL Guide* for MVS/ESA and to the *DB2 SQL Reference* for MVS/ESA for more information.

DJX3407927 (X'00340037') Invalid use of long column.

Cause: The long string column specification in a select or subselect is used in an incorrect manner.

Action: Correct the statement by specifying a string column that is less than 255 characters long. For the restrictions that apply to long string columns, refer to the *DB2 SQL Reference* for MVS/ESA.

DJX3407928 (X'00340038') DJX3407958 (X'00340056')

DJX3407928 (X'00340038') Invalid long varchar type not supported.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407932 (X'0034003C') Run time error: string length too long.

Cause: The length of a character string value exceeds the length of the column to which it is assigned.

Action: Reduce the length of the string or increase the size of the column to correct the error. For more information on character string values, refer to the *DB2 SQL Reference* for MVS/ESA.

DJX3407933 (X'0034000D') Run time error: position of string out of bounds.

Cause: The start parameter of the SUBSTR scalar function must not exceed the length parameter of that function.

Action: Correct the SUBSTR scalar function. For more information refer to the *DB2 SQL Reference* for MVS/ESA.

DJX3407942 (X'00340046.ehex.) Invalid column used for sorting.

Cause: A column specified in the SORT clause of a SELECT statement is invalid.

Action: Correct the statement.

DJX3407943 (X'00340047') Invalid number used for sorting.

Cause: The number used in the SOR clause of a SELECT statement is invalid.

Action: Correct the statement with a valid column name.

DJX3407952 (X'00340050') Declared cursor already open.

Cause: A cursor specified in an OPEN statement is already in use.

Action: Issue a CLOSE statement against the cursor to free it for future use.

DJX3407953 (X'00340051') Prepared statement not declared.

Cause: An attempt was made to issue a statement for which a PREPARE statement was issued without a DECLARE statement.

Action: Include an appropriate DECLARE statement in your application program.

DJX3407954 (X'00340052') Prepared statement not declared.

Cause: An attempt was made to open a cursor for which a PREPARE statement was issued without a DECLARE statement.

Action: Include an appropriate DECLARE statement in your application program.

DJX3407955 (X'00340053') DESCRIBE statement not declared.

Cause: The DESCRIBE statement was issued before a DECLARE statement was issued for it.

Action: Insert an appropriate DECLARE statement in the program before issuing DESCRIBE.

DJX3407956 (X'00340054') DESCRIBE statement does not identify a prepared statement.

Cause: The DESCRIBE statement was issued before a DECLARE statement was issued for it.

Action: Insert an appropriate DECLARE statement in the program before issuing DESCRIBE.

DJX3407957 (X'00340055') Describe statement already open.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3407958 (X'00340056') Undeclared cursor in OPEN statement.

Cause: An OPEN statement specified an unprepared cursor.

Action: Include a PREPARE statement for the cursor in your application program before OPEN is declared.

DJX3407959 (X'00340057') Unprepared cursor in OPEN statement.

Cause: An OPEN statement specified an unprepared cursor.

Action: Include a PREPARE statement for the cursor in your application program before OPEN is declared.

DJX3407960 (X'00340058') Cursor in open statement already open.

Cause: A cursor specified in an OPEN statement is already in use.

Action: Issue a CLOSE statement against the cursor to free it for future use.

DJX3407961 (X'00340059') Undeclared cursor in FETCH.

Cause: FETCH specified an undeclared cursor.

Action: Include a DECLARE statement against the cursor to free it for future use.

DJX3407962 (X'0034005A') Unprepared cursor in FETCH.

Cause: FETCH specified a closed cursor.

Action: Include an OPEN statement for the cursor in your application program before FETCH is issued.

DJX3407963 (X'0034005B') Unopened cursor in FETCH.

Cause: FETCH specified a closed cursor.

Action: Include an OPEN statement for the cursor in your application program before FETCH is issued.

DJX3407964 (X'0034005C') Undeclared cursor in CLOSE.

Cause: A CLOSE statement was issued against an undeclared cursor.

Action: Include a PREPARE statement for the cursor in your application.

DJX3407965 (X'0034005D') Unprepared cursor in CLOSE.

Cause: A CLOSE statement was issued against an unprepared cursor.

Action: Include a DECLARE statement for the cursor in your application.

DJX3407966 (X'0034005E') Unopened cursor in CLOSE.

Cause: A CLOSE statement was issued against an unprepared cursor.

Action: Include a PREPARE statement for the cursor in your application.

DJX3407992 (X'00340078') Invalid SQLID option.

Cause: Specify an SQLID option as a character string or user ID. (Host variables are not supported.)

Action: Correct the syntax of the SET CURRENT SQLID statement.

DJX3408002 (X'00340082') Cannot open spillfile.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3408003 (X'00340083') Invalid long varchar character.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3408004 (X'00340084') LVC location is not in spillfile.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3408005 (X'00340085') LVC Length is not in spillfile.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3408006 (X'00340086') DJX3408047 (X'003400AF')

DJX3408006 (X'00340086') LVC value is not in spillfile.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3408007 (X'00340087') Invalid LVC register.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3408008 (X'00340088') Register not available.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3408022 (X'00340096') Ambiguous host variable reference.

Cause:

Action: Ensure that one and only one host variable of the given name is declared within BEGIN and END declare sections prior to the point of reference.

DJX3408028 (X'0034009C') SQLDA size is too small.

Cause: There are more parameter markers in a query than the SQLDA can accept.

Action: Use a DESCRIBE statement for the query or increase the SQLDA size.

DJX3408033 (X'003400A1') Operands not in same group.

Cause: On a select statement with a WHERE clause, an invalid or illegal comparison is being performed (for example, the comparison of a character field and a numeric field).

Action: Check the select statement and ensure that comparisons are legal, i.e. character fields are compared to character fields and numeric fields are compared to numeric fields.

DJX3408037 (X'003400A5') Var qualified SSCB not defined.

Cause: This is an internal error.

Action: Contact your system administrator.

DJX3408047 (X'003400AF') MAX ROWS EXAMINED governor limit exceeded.

Cause:

Action: Check the governor to see if the maximum number of rows was exceeded. Change the MAX NUMBER OF ROWS parameter to increase the limit.

Chapter 43. Btree Error Messages

The following messages are related to Btree
errors.

DJX3538946 (X'00360002') Btree allocation error.

Cause: An attempt to allocate memory failed.

Action: Increase the region size and if this does not
correct the error, contact an authorized IBM service
representative.

**DJX3538947 (X'00360003') Btree was called with the
wrong parameter.**

Cause: This is an internal error.

Action: Contact an authorized IBM service
representative.

**DJX3538948 (X'00360004') Btree record was not
found.**

Cause: This is an internal error.

Action: Contact an authorized IBM service
representative.

**DJX3538949 (X'00360005') Btree offset overflow, the
file is too big.**

Cause: This is an internal error.

Action: Contact an authorized IBM service
representative.

**DJX3538950 (X'00360006') Btree VRtclose function
failed.**

Cause: This is an internal error.

Action: Contact an authorized IBM service
representative.

**DJX3538951 (X'00360007') Btree VRtopen function
failed.**

Cause: This is an internal error.

Action: Contact an authorized IBM service
representative.

**DJX3538952 (X'00360008') Btree VRfread function
failed.**

Cause: This is an internal error.

Action: Contact an authorized IBM service
representative.

**DJX3538953 (X'00360009') Btree VRfseek function
failed.**

Cause: This is an internal error.

Action: Contact an authorized IBM service
representative.

**DJX3538954 (X'0036000A') Btree VRfwrite function
failed.**

Cause: This is an internal error.

Action: Contact an authorized IBM service
representative.

**DJX3538955 (X'0036000B') Btree VRtmpfile function
failed.**

Cause:

Action: Verify that temporary storage space is
available on DASD. If storage space is available and the
error recurs, contact an authorized IBM service
representative.

**DJX3538956 (X'0036000C') Btree wrong header,
probably the Btree file format was
changed.**

Cause: The file header is incorrect because the data
dictionary is either out of date or obsolete.

Action: Re-create the data dictionary.

Chapter 44. Storage System Interface Error Messages

The following messages are related to storage system interface errors.

DJX3735553 (X'00390001') Cannot allocate enough memory.

Cause: Not enough memory available.

Action: Increase the region size.

DJX3735554 (X'00390002') User conversion exit error.

Cause: A field procedure for converting a column defined by the DMSI failed.

Action: Check the second four bytes for the reason code passed by the field procedure for further explanation of the error.

DJX3735555 (X'00390003') Could not load user column conversion routine.

Cause: Classic Connect could not load a field procedure for a column.

Action: Ensure that the field procedure resides in STEPLIB, a task library, or the system link list.

DJX3735556 (X'00390004') Invalid assignment of column length in user exit.

Cause: The column length assigned by a field procedure is invalid.

Action: Determine and correct the mistake in the field procedure.

DJX3735558 (X'00390006') Cannot create a temporary file.

Cause: Creation of a temporary file failed.

Action: Use the LD TEMP SPACE DMSI configuration parameter to specify the correct attributes for temporary file allocation.

DJX3735559 (X'00390007') Cannot write to a data set.

Cause: An attempted write to a data set failed.

Action: Ensure that data sets and DD statements are properly defined. If so, use the LD TEMP SPACE DMSI configuration parameter to specify correct attributes for temporary files.

DJX3735560 (X'00390008') Cannot determine position within a data set.

Cause: Determination of position within the data set failed.

Action: Ensure data sets and DD statements are properly defined. If so, use the LD TEMP SPACE DMSI configuration parameter to specify correct attributes for temporary files.

DJX3735561 (X'00390009') Cannot establish position within a data set.

Cause: An attempt to establish position within a data set failed.

Action: Ensure data sets and DD statements are properly defined. If so, use the LD TEMP SPACE DMSI configuration parameter to specify correct attributes for temporary files.

DJX3735562 (X'0039000A') Cannot read a data set.

Cause: An attempted read of a data set failed.

Action: Ensure that data sets and DD statements are properly defined. If so, use the LD TEMP SPACE DMSI configuration parameter to specify correct attributes for temporary files.

DJX3735563 (X'0039000B') Wrong VARCHAR length specified when defining table.

Cause: The length of a field mapped to data type VARCHAR is greater than the specified maximum length for VARCHAR fields.

Action: Redefine the maximum length for VARCHAR fields in the projection utility.

Chapter 45. TCP/IP Socket Error Messages

The following messages are related to socket errors.

DJX3932161 (X'003C0001') CI is unable to create socket.

Cause: CI is unable to obtain a TCP/IP socket for a datacom connection. This is probably due to the maximum number of sockets being exceeded.

Action: Try reducing the number of Node Address Entries for this CI.

DJX3932162 (X'003C0002') CI is unable to obtain Internet address from provided dot notation.

Cause: CI is unable to convert an Internet address from the Task Info or Node Address Entry into proper network format.

Action: Check the Classic Connect system configuration file to ensure that all Internet addresses are correctly specified.

DJX3932163 (X'003C0003') CI is unable to bind socket.

Cause: CI is unable to use the supplied Internet address from the Node Address Entry.

Action: Make sure that the Internet address and port number combination of each TCP/IP CI in the system configuration file is unique. This problem can also occur if a Classic Connectnode is restarted too quickly after being shut down.

DJX3932166 (X'003C0006') Socket nnn is busy.

Cause: This is an informational message.

Action: No action is required.

DJX3932167 (X'003C0007') Socket is unavailable.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932168 (X'003C0008') Socket nnn is not connected.

Cause: CI detected that a partner CI closed the datacom connection.

Action: Check the status of other CIs on the network, and restart as necessary.

DJX3932169 (X'003C0009') CI detected an invalid socket bitmap.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932170 (X'003C000A') CI detected an invalid socket descriptor nnn.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932171 (X'003C000B') CI detected an invalid socket address xxxxx.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932172 (X'003C000C') CI detected an unknown socket error.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932173 (X'003C000D') Protocol nnn is not supported by CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932174 (X'003C000E') DJX3932187 (X'003C001B')

DJX3932174 (X'003C000E') Host is not ready.

Cause: An attempt to connect to a partner CI failed because the partner is not ready. The connection attempt will be tried again. This is an informational message.

Action: No action is required.

DJX3932175 (X'003C000F') Transport layer did not provide error number to CI.

Cause: The communications layer returned an unknown error code to a Classic Connect component. The log file might contain the error code returned. This is usually an indication of some configuration error.

Action: Refer to the appropriate communications layer error codes documentation for further assistance.

DJX3932176 (X'003C0010') Bad file descriptor nnn.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932177 (X'003C0011') Invalid argument xxxxx.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932178 (X'003C0012') TCP/IP IUCV layer detected error in CI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932179 (X'003C0013') Protocol unsupported.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932180 (X'003C0014') Unable to assign address xxxxx.

Cause: An attempt to connect to a partner CI failed because the partner is not yet ready. A connection attempt will be tried again. This is an informational message.

Action: No action is required.

DJX3932181 (X'003C0015') Address family nnnn not supported.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932182 (X'003C0016') Address xxxxx already in use.

Cause: CI is unable to use the supplied Internet address from the Node Address Entry.

Action: Make sure that the Internet address and port number combination of each TCP/IP CI in the system configuration file is unique. This problem can also occur if a Classic Connect node is restarted too quickly after being shut down.

DJX3932183 (X'003C0017') Address xxxxx is invalid. Memory referenced might be wrong length.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932184 (X'003C0018') Operation on socket nnn is unsupported.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932186 (X'003C001A') Attempt to connect socket nnn refused by remote host. Remote host might not be operational.

Cause: An attempt to connect to a partner CI failed because the partner is not ready. A connection attempt will be tried again. This is an informational message.

Action: No action is required.

DJX3932187 (X'003C001B') Operation time out.

Cause: An attempt to connect to a partner CI failed because the partner is not ready. A connection attempt will be tried again. This is an informational message.

Action: No action is required.

DJX3932188 (X'003C001C') Operation is in progress.

Cause: This is an informational message.

Action: No action is required.

DJX3932189 (X'003C001D') Operation will be completed when resources become available.

Cause: This is an informational message.

Action: No action is required.

DJX3932190 (X'003C001E') Socket nnn reset.

Cause: This is an informational message.

Action: No action is required.

DJX3932191 (X'003C001F') Socket nnn is already connected.

Cause: The CI tried to connect to a location using a socket that is already in the process of connecting to a location.

Action:

DJX3932192 (X'003C0020') Network is unreachable.

Cause: The specified Internet address is invalid.

Action: Check the TCP/IP network to make sure it is correctly installed. Make sure that Internet addresses are specified correctly in the Classic Connect system configuration file.

DJX3932193 (X'003C0021') Network is down.

Cause: CI cannot connect to the specified partner because the datacom link is down.

Action: Check and restart the datacom link as necessary.

DJX3932194 (X'003C0022') TCP/IP does not have available buffers to complete request.

Cause: Data transfer is blocked until TCP/IP resources become available. This is an informational message.

Action: No action is required.

DJX3932196 (X'003C0024') CI attempted to use active socket for listening. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932197 (X'003C0025') CI failed to recognize error number *xxxxx* supplied by the transport layer.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932198 (X'003C0026') CI failed to recognize DJX error number *xxxxx*. Data field of return code contains xxxxxxxx.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932199 (X'003C0027') CI failed to recognize DJX error number *xxxxx*.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3932200 (X'003C0028') Partner CI dropped connection nnn.

Cause: CI detected that the partner CI closed the datacom connection.

Action: Check the status of other CIs on the network and restart as necessary.

DJX3932204 (X'003C002C') The bind call to the operating system failed because the socket is busy.

Cause: The internet address for the socket is *xxxxx.xxxxx.xxxxx.xxxxx*. This problem is usually caused by restarting a CI too soon after a CI terminated.

Action: Restart the CI in a few minutes.

Chapter 46. TCP/IP CI Error Messages

The following messages are related to TCP/IP CI errors.

DJX3997702 (X'003D0006') IUCV path ID is out of range.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997704 (X'003D0008') CI provided TCP/IP server with initial message of the wrong length.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997705 (X'003D0009') Initial message for TCP/IP server does not begin with IUCVAPI.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997706 (X'003D000A') Initial message for TCP/IP server is not X'0002' at offset 10 (HEX).

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997710 (X'003D000E') KILL error detected by TCP/IP server.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997712 (X'003D0010') No socket numbers available in Allocation Map.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997713 (X'003D0011') ERRNO was set during socket operation.

Cause: This is an informational message.

Action: Check the output for your socket number using network utilities.

DJX3997716 (X'003D0014') SEND rejected and LASTERRNO failed.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997719 (X'003D0017') IUCV CONNECT failed.

Cause: TCP/IP is not available or the Classic Connect configuration file has an incorrect name for the TCP/IP address space.

Action: Verify that the TCP/IP subsystem is available and that the ADDRESS SPACE NAME parameter is set correctly. Also, check the TCP/IP error logs and the MVS system console messages for other communications-related messages, and correct anyb problems noted.

DJX2997721 (X'003D0019') Socket descriptor is not in allocation map.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997723 (X'003D001B') CI is unable to allocate memory for RECEIVE operation.

Cause: Insufficient MVS main memory.

Action: Increase the Classic Connect region size and restart.

DJX3997724 (X'003D001C') Unrecognized code returned to CI from TCP/IP SERVER.

Cause: CI is unable to match the TCP/IP error code with a CI error code.

Action: Contact an authorized IBM service representative.

DJX3997725 (X'003D001D')   DJX3997728 (X'003D0020')

DJX3997725 (X'003D001D') CI detected an invalid number of sockets specified per IUCV path.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997726 (X'003D001E') CI specified too many IOVECTIRS during SEND operation.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997727 (X'003D001F') CI IUCV control block inconsistent with IUCV NCB.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX3997728 (X'003D0020') CI Path control block inconsistent with IUCV PCB.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

Chapter 47. APPC CI Error Messages

The following messages are related to APPC CI errors.

DJX4128776 (X'003F0008') Message does not have enough space for CI subsystem header which is used to keep track of the status of communication.

Cause: Classic Connect system error.

Action: Contact an authorized IBM service representative.

Chapter 48. AIX System Error Messages

The following messages are related to AIX errors.

**DJX4194304 (X'00400000') Classic Connect
initializing.**

Cause: This is an informational message.

Action: No response required.

**DJX4194305 (X'00400001') The DJX_CONFIG
variable was not found in the
environment.**

Cause: The DJX_CONFIG variable was included in the
environment.

Action: Set the DJX_CONFIG variable to the name of
the configuration file.

**DJX4194307 (X'00400003') The SUBSYSTEM ID
parameter was not found in the
environment.**

Cause: The SUBSYSTEM ID parameter must be
present in the configuration file.

Action: Edit the AIX kernel configuration file to include
the SUBSYSTEM ID parameter. For details about the
SUBSYSTEM ID refer to the *DataJoiner for AIX
Planning, Installation, and Configuration Guide*

**DJX4194308 (X'00400004') The SUBSYSTEM ID path
name must be a fully qualified path and
must begin with a slash character.**

Cause: The value of the SUBSYSTEM ID configuration
parameter does not conform to the requirements stated
in the *DataJoiner Planning, Installation, and
Configuration Guide*

Action: Reset the SUBSYSTEM ID path based on the
requirements for this path as stated in the *DataJoiner
Planning, Installation, and Configuration Guide*

**DJX4194312 (X'00400008') A Classic Connect kernel
with the same SUBSYSTEM ID appears
to be running on this machine already.
Each node on a single machine must**

DJX4194319 (X'0040000F') DJX4194331 (X'0040001B')

DJX4194319 (X'0040000F') A request for RPC data exceeds the length of the IPC message.

Cause: System-wide message resource limits may have been reached.

Action: Check the system message resources. Increase the limits for message resources.

DJX4194320 (X'00400010') Server can't get a port to the External Task that sent the request.

Cause: System-wide message resource limits may have been reached.

Action: Check the system message resources. Increase the limits for message resources.

DJX4194321 (X'00400011') Platform IPC end_buffer failed.

Cause: System-wide message resource limits may have been reached.

Action: Check the system message resources. Increase the limits for message resources.

DJX4194322 (X'00400012') A msgsnd call to the operating system failed.

Cause: The unexpected error number returned is *xxxxx* and the line number is *xxxxx*.

Action: Contact an authorized IBM service representative.

DJX4194323 (X'00400013') Server failed to respond to the RPC request within timeout or CheckMsg did not respond within timeout.

Cause: This is a warning level message. It usually indicates that the system is heavily loaded and that the Classic Connect kernel could not service a request as quickly as is usually expected. This situation might result from a service that was swapped out or because some other process is monopolizing CPU, IO, or real memory resources.

Another possible cause is that the kernel was not started.

Action: Ensure that the kernel was started; then, check other system processes and attempt the request again.

DJX4194324 (X'00400014') The msgrcv call to the operating system failed.

Cause: The message queue ID is *xxxxx*, the error number is *xxxxx*.

Action: Contact an authorized IBM service representative.

DJX4194325 (X'00400015') Unable to make a connection port for this task.

Cause: This message indicates that there might be a limit on message queues or message length.

Action: Contact an authorized IBM service representative.

DJX4194326 (X'00400016') Unable to connect to the kernel instance.

Cause: The msgget() operating system call failed.

Action: Contact an authorized IBM service representative.

DJX4194329 (X'00400019') Internal Logic Error

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4194330 (X'0040001A') Bad or extraneous characters were encountered while reading the MAXIMUM NUMBER OF TASKS parameter from the configuration file.

Cause: An error in the MAXIMUM NUMBER of TASKS parameter was encountered.

Action: Edit the kernel configuration file to correct the MAXIMUM NUMBER OF TASKS parameter. For detailed information on this parameter, refer to the *DataJoiner Planning, Installation and Configuration G;uide*

DJX4194331 (X'0040001B') The MAXIMUM NUMBER OF TASKS value given in the configuration file is below the minimum allowed size.

Cause: The value of the MAXIMUM NUMBER OF TASKS parameter is too small. The minimum allowed size will be used.

Action: Edit the kernel configuration file to correct the MAXIMUM NUMBER OF TASKS parameter. For

detailed information on this parameter, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide*

DJX4194332 (X'0040001C') Server did not respond to RPC request or CheckMsg did not respond.

Cause: The server failed to respond to an RPC request, or CheckMsg failed to respond within the timeout period.

Action: Check for adequate kernel resources. If kernel resources are adequate, contact an authorized IBM service representative.

DJX4194333 (X'0040001D') The MAXIMUM NUMBER OF TASKS parameter was not provided in the kernel configuration file.

Cause: No value for the MAXIMUM NUMBER OF TASKS parameter was provided. The default value will be used.

Action: Edit the kernel configuration file to correct the MAXIMUM NUMBER OF TASKS parameter. For detailed information on this parameter, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide*

DJX4194334 (X'0040001E') The NODEID parameter was not provided in the configuration file.

Cause: No value for the NODEID parameter was provided. The default value will be used.

Action: Edit the kernel configuration file to add the NODEID parameter. For detailed information on this parameter, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide.*

DJX4194335 (X'0040001F') Bad or extraneous characters were encountered while reading the SIMON TRACE LEVEL parameter from the configuration file. The value accepted is nnn.

Cause: The value of the SIMON TRACE LEVEL configuration parameter contains characters that are not digits.

Action: Correct the value of the SIMON TRACE LEVEL configuration parameter. Refer to the *DataJoiner*

for AIX Planning, Installation, and Configuration Guide for information about acceptable values.

DJX4194336 (X'00400020') The SIMON TRACE LEVEL parameter was not provided in the configuration file.

Cause: No value for the SIMON TRACE LEVEL parameter was provided. The default value will be used.

Action: Edit the kernel configuration file to add the SIMON TRACE LEVEL parameter. For information about the configuration file, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide* for the platform you are using.

DJX4194337 (X'00400021') The SIMON TRACE LEVEL value provided in the configuration file is below the minimum allowed value.

Cause: The value of the SIMON TRACE LEVEL parameter is too small. The minimum value will be used.

Action: Edit the kernel configuration file to correct the SIMON TRACE LEVEL parameter. For detailed information about this parameter, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide* for the platform you are using.

DJX4194338 (X'00400022') The SIMON TRACE LEVEL value provided in the configuration file exceeds the allowable maximum value.

Cause: The value of the SIMON TRACE LEVEL parameter is too large. The default will be used.

Action: Edit the kernel configuration file to correct the SIMON TRACE LEVEL parameter. For detailed information about this parameter, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide* for the platform you are using.

DJX4194340 (X'00400024') The NODEID value provided in the configuration file is below the minimum allowed value.

Cause: The value of the NODEID parameter is too small. The minimum value will be used.

Action: Edit the kernel configuration file to correct the NODEID parameter. For detailed information about this parameter, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide* for the platform you are using.

DJX4194341 (X'00400025') The NODEID value provided in the configuration file exceeds the maximum allowable value.

Cause: The value of the NODEID parameter is too large. The default will be used.

Action: Edit the kernel configuration file to correct the NODEID parameter. For detailed information about this parameter, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide* for the platform you are using.

DJX4194342 (X'00400026') The DJX_PATH variable was not found in the environment.

Cause: The DJX_PATH variable should be set to the directory containing the Classic Connectexecutables and service scripts. A Classic Connect shell script was incorrectly modified. Each Classic Connect program is invoked from a shell script that sets required environment variables, including DJX_PATH (which is usually set to /usr/bin/lpp, the directory into which the installation program places Classic Connectexecutable programs).

Action: Undo the incorrect change or reinstall the product.

DJX4194343 (X'00400027') The DJX_PATH variable must be a fully-qualified directory name, and it must begin with a slash character.

Cause: Incorrect changes were made in either /usr/bin/lpp/djxsimon or another shell script supplied with Classic Connect. The value of DJX_PATH must always be a complete path name starting from the root of the file system.

Action: Correct the error and restart Classic Connect.

DJX4194344 (X'00400028') The error occurred while processing the *xxxxx* configuration file.

Cause: There might be a syntax error in the configuration file.

Action: Check the configuration file and correct any syntax errors.

DJX4194347 (X'0040002B') The shmdt call to the operating system failed.

Cause: The shared memory address passed to the function is *xxxxx*.

Action: Contact an authorized IBM service representative.

DJX4194349 (X'0040002D') The execl call to the operating system failed.

Cause: The requested task cannot be started. System resources might be low, or the name of the executable in the configuration file is incorrect.

Action: Check and correct the executable name in the configuration file. If the name is correct, contact an authorized IBM service representative.

DJX4194353 (X'00400031') The getuserpw call to the operating system failed.

Cause: The requested task cannot be started.

Action: Refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide* for information about the getuserpw subroutine for possible reasons why the subroutine failed.

DJX4194359 (X'00400037') DJX4194380 (X'0040004C')

DJX4194359 (X'00400037') The getuserattr call to the operating system failed.

Cause: The user groups could not be obtained. The specified guarded service will not be started.

Action: Refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide* for information about getusrattr.

DJX4194362 (X'0040003A') The getgroupattr call to the operating system failed.

Cause: The user group ID could not be obtained. The specified guarded service will not be started.

Action: Refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide* for information about getusrattr.

DJX4194364 (X'0040003C') The setgidx call to the operating system failed.

Cause: The user login could not be set. The specified guarded service will not be started.

Action: Refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide* for information about getusrattr.

DJX4194365 (X'0040003D') A critical region could not be obtained within *xxxxx* attempts. The sleep time between attempts is *xxxxx* microseconds.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4194369 (X'00400041') The LLWait function timed out while waiting for an event.

Cause: The timeout time given is *xxxxx* milliseconds. This is an internal error.

Action: Check to see if one of the kernel processes is hanging. If this is not the case, contact an authorized IBM service representative.

DJX4194372 (X'00400044') The LLPost function could not post the specified event because the receiver (task number *xxxxx*, process number *xxxxx*) had already timed out.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4194376 (X'00400048') An attempt to add another entry to ATT failed.

Cause: The active task table (ATT) is full.

Action: Increase the MAXIMUM NUMBER OF TASKS parameter in the kernel configuration file.

DJX4194377 (X'00400049') An attempt to start a new task failed.

Cause: Might be caused by invalid task flags value in the TASK INFO ENTRY in the kernel configuration file. The function task flags value is *xxxxx*.

Action: Check the TASK INFO ENTRY for the task, and be sure that task flags are valid.

DJX4194379 (X'0040004B') The BUDDY PERCENTAGE value given in the kernel configuration file is below the minimum percentage size.

Cause: The minimum allowed percentage will be used.

Action: Edit the Classic Connect kernel configuration file to correct the value of the BUDDY PERCENTAGE parameter. For detailed information about this parameter, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide*

DJX4194380 (X'0040004C') The BUDDY PERCENTAGE value in the kernel configuration file exceeds the maximum allowable value.

Cause: The maximum allowed percentage will be used.

Action: Edit the Classic Connect kernel configuration file to correct the value of the BUDDY PERCENTAGE parameter. For detailed information about this parameter, refer to the *DataJoiner for AIX Planning, Installation, and Configuration Guide*

DJX4194381 (X'0040004D') An attempt to find a nonexistent task in the ATT failed.

Cause: The PID of the nonexistent task is *xxxxx*. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4194382 (X'0040004E') A memory request for *xxxxx* bytes failed.

Cause: The amount of memory currently available is *xxxxx* bytes.

Action: This is a system resource problem. Contact an authorized IBM service representative.

DJX4194383 (X'0040004F') The getlogin call to the operating system failed.

Cause: The error number returned is *xxxxx*.

Action: Contact an authorized IBM service representative, and report the error number.

DJX4194384 (X'00400050') The ulimit call to the operating system failed.

Cause: The error number returned is *xxxxx*.

Action: Contact an authorized IBM service representative, and report the error number.

DJX4194385 (X'00400051') The malloc call to the operating system failed.

Cause: The error number returned is *xxxxx*. The requested size is *xxxxx*.

Action: Contact an authorized IBM service representative, and report the error number and requested size.

DJX4194386 (X'00400052') The msgget for a private queue call to the operating system failed.

Cause: The message flags are *xxxxx*.

Action: Contact an authorized IBM service representative, and report the message flags value displayed by the error message.

DJX4194391 (X'00400057') The LLPost function could not post the specified event because the GTID index of *xxxxx* exceeds the maximum of *xxxxx*.

Cause: This is probably an internal error.

Action: Contact an authorized IBM service representative.

DJX4194393 (X'00400059') The LLPost function could not post the specified event because the task (*xxxxx*) does not allow Posts.

Cause: The ulFlags for this task is *xxxxx*. This is probably an internal error.

Action: Check the TASK INFO ENTRY for the task and be sure that the task flag setting allows posts. If the problem persists, contact an authorized IBM service representative.

DJX4194394 (X'0040005A') The LLWait function could not wait for the specified event because the ECB was in an unknown state.

Cause: The expected state is 0 or *xxxxx*, the current state is *xxxxx*.

Action: This is an internal error. Contact an authorized IBM service representative.

DJX4194395 (X'0040005B')
DJX4194396 (X'0040005C') The EXTERNAL TASK WAIT TIME parameter value of 0 provided in the configuration file is below the minimum allowed value of 1.

Cause: The minimum allowed value will be used.

Action: Correct the value assigned to the EXTERNAL TASK WAIT TIME parameter in the configuration file.

DJX4194397 (X'0040005D') The EXTERNAL TASK WAIT TIME value of 4294967295 provided in the configurations file exceeds the maximum allowed value of 900000.

Cause: The maximum allowed value will be used.

Action: Edit the kernel configuration file to add the EXTERNAL TASK WAIT TIME parameter. For detailed information about the configuration file, refer to the Classic Connect installation guide for the platform you are using.

DJX4194398 (X'0040005E') Bad or extraneous characters were encountered while reading the EXTERNAL TASK WAIT TIME parameter from the configuration file.

Cause: The value accepted is 6000.

Action: Correct the value assigned to the EXTERNAL WAIT TIME parameter in the configuration file.

DJX4194399 (X'0040005F') The EXTERNAL TASK WAIT TIME parameter was not provided in the kernel configuration file.

Cause: The default value will be used.

Action: Edit the kernel configuration file to add the EXTERNAL TASK WAIT TIME parameter. For detailed information about the configuration file, refer to the Classic Connectinstallation guide for the platform you are using.

DJX4194403 (X'00400063') Bad or extraneous characters were encountered while reading the 'SHARED MEMORY ADDRESS' parameter from the configuration file.

Cause: The value accepted is *xxxxx*. This is an informational message. The system attaches the shared memory segment at a virtual address chosen by the operating system.

Action: None.

DJX4194404 (X'00400064') A bad or no 'SHARED MEMORY ADDRESS' parameter was provided in the configuration file.

Cause: The calculated value of *xxxxx* will be used. This is an informatioal message. The system attaches the shared memory segment at a virtual address chosen by the operating system.

Action: None.

DJX4194405 (X'00400065') The 'SHARED MEMORY ADDRESS' parameter from the configuration file of *xxxxx* could not be used.

Cause: The error number returned from the shmat function is *xxxxx*. An attempt will be made to calculate a valid shared memory address. This is an informational message. The system attaches the shared memory segment at a virtual address chosen by the operating system.

Action: None.

DJX4194406 (X'00400066') A bad or no 'SHARED MEMORY ADDRESS' parameter was provided in the configuration file.

Cause: No valid address could be found between *xxxxx* and *xxxxx*. The Classic Connect kernel will now terminate.

Action: There is a problem attaching shared memory. Look for system resource limits for shared memory and correct the problem. Contact an authorized IBM service representative if this does not resolve the problem.

DJX4194408 (X'00400068') The task that could not be started is *xxxxx*.

Cause: A task did not start.

Action: Check that the TASK INFO ENTRY parameter contains the right task. Also check that the task exists and can be executed.

DJX4194409 (X'00400069') The ATT entry at *xxxxx* was freed previously.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4194416 (X'00400070') Configuration file variable is not present in start script.

Cause: The configuration file variable, which is required, is not present in the start script.

Action: Add a correct configuration file variable to the start script.

Chapter 49. CPI Communication Interface Messages

The following messages are related to the CPI communication interface.

DJX4587523 (X'00460003') Unrecognized return code value from lower level routine.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587524 (X'00460004') Conversation is not currently allocated.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587552 (X'00460020') CI transport layer detected no error.

Cause:

Action:

DJX4587553 (X'00460021') Invalid deallocate.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587554 (X'00460022') Invalid send type.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587555 (X'00460023') Attempt to open an ACB failed.

Cause: There might be a syntax error in the LU name, or one of the names might not be defined to VTAM, or one of the LU names might already be in use by another application.

Action: Check the Task Info Entry in the configuration file for the APPC CI. Correct the Task Info Entry.

DJX4587556 (X'00460024') The APPC Control Block is the wrong size.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587557 (X'00460025') The Access Control Block is the wrong size.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587558 (X'00460026') The COMM Control Block is the wrong size.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587559 (X'00460027') Unable to create an ACB, RPL, or EXIT control block.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587560 (X'00460028') CPIC return code value set.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587561 (X'00460029') Data ready count is < 0.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4587562 (X'0046002A') Pending readcount is < 0.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

Chapter 50. Log File Error Messages

The following messages are related to the Classic Connect log file.

DJX4718593 (X'00480001') An error occurred while attempting to open the log file.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718594 (X'00480002') This message text was expecting more parameters than were provided by error code information.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718595 (X'00480003') An error occurred while attempting to write to the log file.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718596 (X'00480004') The CTABLE cell is not initialized properly for this task or function.

Cause: This might be the result of improper platform initialization or random destruction of memory. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718597 (X'00480005') Bad or extraneous characters were encountered while reading the LOG FILE LINE LENGTH parameter from the configuration file.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718599 (X'00480007') This log message has a mismatch between the given data fields and the message field specification.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718600 (X'00480008') This log message contains an invalid flag value.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718601 (X'00480009') No message text would be obtained for error number *xxxxx*

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718602 (X'0048000A') No message text would be obtained for error number *xxxxx*

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718603 (X'0048000B') The log task port number was improperly initialized.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718604 (X'0048000C') The ANCH LOGINFOSTR cell was not initialized properly for this function. This might be the result of improper platform initialization or random destruction of memory.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718605 (X'0048000D') DJX4718620 (X'0048001C')

DJX4718605 (X'0048000D') The hActualLogPH cell is not initialized.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718606 (X'0048000E') A request was sent to the log task.

Cause: This is an informational message.

Action: No action is required.

DJX4718607 (X'0048000F') A call was made to the LLLogCmd function with a bad command parameter.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718611 (X'00480013') Toggle request was denied because no keyword was provided in the system configuration file.

Cause: This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718612 (X'00480014') The LOG FILE LENGTH in the configuration file exceeds the maximum allowed size.

Cause: The maximum allowable length will be used.

Action: Correct the LOG FILE LENGTH parameter.

DJX4718613 (X'00480015') The LOG FILE LENGTH in the configuration file is below the minimum allowed size.

Cause: The minimum allowable length will be used.

Action: Correct the LOG FILE LENGTH parameter.

DJX4718614 (X'00480016') No LOG FILE LENGTH was given in the configuration file.

Cause: The minimum allowable length will be used.

Action: Correct the LOG FILE LENGTH parameter

DJX4718615 (X'00480017') An error occurred while attempting to close the log file.

Cause: The error number obtained from the operating system is displayed. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718616 (X'00480018') An error occurred while attempting to format a log file message.

Cause: The log will be closed and the log task will terminate. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718617 (X'00480019') This message text is not expecting all the information provided.

Cause: The information provided by error form 08 is displayed. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718618 (X'0048001A') This message text is not expecting all the information provided.

Cause: The information provided by error 14 is displayed. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718619 (X'0048001B') This message text is not expecting all the information provided.

Cause: The information provided by error form 16 is displayed. This is an internal error.

Action: Contact an authorized IBM service representative.

DJX4718620 (X'0048001C') This message text is not expecting all the information provided.

Cause: The information provided by error form 14 is displayed. This is an internal error.

Action: Contact an authorized IBM service representative.

Chapter 51. User Exit Predefined Error Messages

The following messages are related to user exit predefined errors.

DJX5046272 (X'004D0000') Undefined

Cause: Return code 0 is used for normal completion.

Action: No action is required.

DJX5046273 (X'004D0001') Invalid TVA Name structure passed to User Exit Interface.

Cause: An error occurred during processing of the Table Names list that was passed to DJXACVAL.

Action: Report this error to an authorized IBM service representative.

DJX5046274 (X'004D0002') Service access rejected by user exit.

Cause: The Service Validation user exit denied access to the service requested.

Action: See an authorized IBM service representative for valid services for the Logon ID used.

DJX5046275 (X'004D0003') Table access rejected by user exit.

Cause: The Access Validation user exit denied access to one or more of the Table Names referenced by this SQL statement.

Action: See an authorized IBM service representative for Table Names that the SQL ID in use is permitted to access.

DJX5046278 (X'004D0006') The Service Validation User Exit environment was corrupted.

Cause: User exit error checking found a problem in the environment.

Action: Report the problem to an authorized IBM service representative.

DJX5046279 (X'004D0007') The Access Validation User Exit environment was corrupted.

Cause: User exit error checking found a problem in the environment.

Action: Report the problem to an authorized IBM service representative.

Chapter 52. Classic Connect Internal Error Messages

The following messages are related to Classic
Connect internal errors.

**DJX2576980377 (X'99999999') Classic Connect
INTERNAL ERROR program: <name>,
Line: <number>**

Cause: This is an internal error. This message is
always issued in English.

Action: Save the error message and any other error
messages displayed, and give the material to an
authorized IBM service representative.

Section 4. Appendixes

Appendix A. Communications Errors

If your application returns a sqlcode of -30081, it means that a communications error has been detected. The actual error encountered by the communication subsystem is returned in the list of error tokens for the 30081 error message. The following lists possible communications errors that can be returned.

TCP/IP

Listed below are the *errnos* that the user might encounter most frequently when using TCP/IP under AIX. (The *errno* numbers are in parentheses.) Not ALL *errnos* that the user can encounter are listed. *Errnos* can be found in the file **/usr/include/sys/errno.h**.

- EMFILE (24): The per-process file descriptor table is full.
- EPIPE (32): Broken pipe.
- EADDRINUSE (67): The specified address is already in use.
- ENETDOWN (69): The network is down.
- ENETUNREACH (70): No route to the network or host is available.
- ENETRESET (71): The network has dropped the connection on reset.
- ECONNRESET (73): Connection has been reset by the partner.
- ENOBUFS (74): Insufficient resources were available in the system to complete the call.
- ENOTCONN (76): The socket is not connected.
- ETIMEDOUT (78): The establishment of a connection timed out before a connection was made.
- ECONNREFUSED (79): The connection has been refused. If you are trying to connect to the database, check that the database manager and TCP/IP protocol support at the server has been started successfully.

  If using SOCKS protocol support, also ensure that TCP/IP protocol support at the SOCKS server has been started successfully.
- EHOSTDOWN (80): The host is down.
- EHOSTUNREACH (81): No route to the host is available.

For more information on AIX TCP/IP communication errors, refer to *Calls and Subroutines Reference: Base Operating System*.

Listed below are the *errnos* that the user might encounter most frequently when using HP-UX under AIX. (The *errno* numbers are in parentheses.) Not ALL *errnos* that the user can encounter are listed. *Errnos* can be found in the file **/usr/include/sys/errno.h**. The *errno* number itself is in parentheses.

- EMFILE (24): The per-process file descriptor table is full.

- EPIPE (32): Broken pipe.

- EADDRINUSE (226): The specified address is already in use.

- ENETDOWN (228): The network is down.

- ENETUNREACH (229): No route to the network or host is available.

- ENETRESET (230): The network has dropped the connection on reset.

- ECONNRESET (232): Connection has been reset by the partner.

- ENOBUFS (233): Insufficient resources were available in the system to complete the call.

- ENOTCONN (235): The socket is not connected.

- ETIMEDOUT (238): The establishment of a connection timed out before a connection was made.

- ECONNREFUSED (239): The connection has been refused. If you are trying to connect to the database, check that the database manager and TCP/IP protocol support at the server has been started successfully.

  If using SOCKS protocol support, also ensure that TCP/IP protocol support at the SOCKS server has been started successfully.

- EHOSTDOWN (241): The host is down.

- EHOSTUNREACH (242): No route to the host is available.

For more information on HP-UX TCP/IP communication errors, refer to the HP-UX Reference Manuals.

Listed below are the *errnos* that the user might encounter most frequently when using TCP/IP under OS/2. (The *errno* numbers are in parentheses.) Not ALL *errnos* that the user might encounter are listed. *Errnos* can be found in the file **nerrno.h**.

- SOCEPIPE (10032): Broken pipe.

- EADDRINUSE (10048): The specified address is already in use.

- ENETDOWN (10050): The network is down.

- ENETUNREACH (10051): No route to the network or host is available.

- ENETRESET (10052): The network has dropped the connection on reset.

- ECONNRESET (10054): Connection has been reset by the partner.

- ENOBUFS (10055): No buffer space available.

- ENOTCONN (10057): The socket is not connected.

- ETIMEDOUT (10060): The establishment of a connection timed out before a connection was made.

- ECONNREFUSED (10061): The connection has been refused. If you are trying to connect to the database, check that the database manager and TCP/IP protocol support at the server has been started successfully.

  If using SOCKS protocol support, also ensure that TCP/IP protocol support at the SOCKS server has been started successfully.
- EHOSTDOWN (10064): The host is down.
- EHOSTUNREACH (10065): No route to the host is available.

For more information on OS/2 TCP/IP communication errors, refer to the OS/2 TCP/IP documentation.

Listed below are the *errnos* that the user might encounter most frequently when using a DOS TCP/IP product (IBM TCP/IP for DOS, FTP TCP/IP for DOS, Novell TCP/IP for DOS). (The *errno* numbers are in parentheses.) Not ALL *errnos* that the user might encounter are listed.

- EPIPE (32): Broken pipe.
- EADDRINUSE (48): The specified address is already in use.
- ENETDOWN (50): The network is down.
- ENETUNREACH (51): No route to the network or host is available.
- ENETRESET (52): The network has dropped the connection on reset.
- ECONNRESET (54): Connection has been reset by the partner.
- ENOBUFS (55): No buffer space available.
- ENOTCONN (57): The socket is not connected.
- ETIMEDOUT (60): The establishment of a connection timed out before a connection was made.
- ECONNREFUSED (61): The connection has been refused. If you are trying to connect to the database, check that the database manager and TCP/IP protocol support at the server has been started successfully.
- EHOSTDOWN (64): The host is down.
- EHOSTUNREACH (65): No route to the host is available.
- ENOTINSTLD (66): Protocol stack not installed.

For more information on DOS TCP/IP communication errors, refer to the documentation for your DOS TCP/IP product.

Listed below are the *errnos* that the user might encounter most frequently when using TCP/IP under Windows. (The *errno* numbers are in parentheses.) Not ALL *errnos* that the user can encounter are listed. Errors returned by WSAGetLastError() can be found in the file **winsock.h**.

- WSAEINPROGRESS (10036): A blocking Windows Sockets operation is in progress.

- WSAENOPROTOOPT (10042): The option is unknown or unsupported.

- WSAEADDRINUSE (10048): The specified address is already in use.

- WSAENETDOWN (10050): The network is down.

- WSAENETUNREACH (10051): No route to the network or host is available.

- WSAENETRESET (10052): The network has dropped the connection on reset.

- WSAECONNRESET (10054): Connection has been reset by the partner.

- WSAENOBUFS (10055): No buffer space available.

- WSAENOTCONN (10057): The socket is not connected.

- WSAETIMEDOUT (10060): The establishment of a connection timed out before a connection was made.

- WSAECONNREFUSED (10061): The connection has been refused. If you are trying to connect to the database, check that the database manager and TCP/IP protocol support at the server has been started successfully.

- WSAEHOSTDOWN (10064): The host is down.

- WSAEHOSTUNREACH (10065): No route to the host is available.

- WSASYSNOTREADY (10091): The underlying network subsystem is not ready for network communication.

- WSAVERNOTSUPPORTED (10092): The version of Windows Sockets API support requested is not provided by this particular Windows Sockets implementation.

For more information on TCP/IP communication errors under Windows, refer to the Windows Sockets documentation.

APPC

Below is a list of CPI-C function return codes that the user might encounter most frequently. It is NOT a complete list of return codes. The number in parentheses indicates the defined number corresponding to the return code.

- CM_ALLOCATE_FAILURE_NO_RETRY (1): Allocation failed because of a non-temporary condition. For example, the session cannot be activated due to a system definition error or session-activation protocol error. This return code is also returned when the session is deactivated because of a session protocol error before the conversation can be allocated.

- CM_ALLOCATE_FAILURE_RETRY (2): Allocation failed because of a temporary condition. For example, the session cannot be activated due to a temporary lack of resources at the local system or remote system.

- CM_CONVERSATION_TYPE_MISMATCH (3): Allocation failed because the remote program does not support the conversation type in the allocation request.

This is likely a problem with the TP at the server. Ensure that the TP at the server is configured to support a conversation type of *basic*.

- CM_TPN_NOT_RECOGNIZED (9): This error occurs when the allocate request is sent to the remote system. It indicates that the remote system does not recognize the transaction program name specified in the request. If you are not using global directory services, ensure that the TP name specified in the CPI-C Side Information profile at the client, matches the TP name specified at the server. If you are using global directory services, request your database administrator's help to ensure that the TP name specified in the global directory entry matches the TP name specified at the server.

- CM_TP_NOT_AVAILABLE_NO_RETRY (10): This error occurs when the allocate request is sent to the remote system. It indicates that the remote LU recognizes the TP name sent, but cannot start the program. If you are trying to connect to DB2 for AIX, ensure that the path to the executable, specified at the server, is correct. Also check that the userid specified in the TPN profile at the server is valid.

- CM_TP_NOT_AVAILABLE_RETRY (11): This error occurs when the allocate request is sent to the remote system. It indicates that the remote LU recognizes the TP name sent, but for some reason which may be temporary, it cannot start the program. If you are trying to connect to DB2 for OS/2, check that the database manager and APPC protocol support at the server has been started successfully.

- CM_DEALLOCATED_ABEND (17): This error occurs when the remote program deallocates the conversation. This can occur if the remote program has abnormally terminated, or has encountered a fatal error condition. If you are trying to connect to DB2 for AIX, check that the database manager and APPC protocol support at the server has been started successfully. For an AIX server, this error could also be caused by one of the following:

  - The database agent at the server was forced off by the system administrator.
  - A database agent could not be started at the server because the *maxagents* database manager configuration parameter has been exceeded. Check the First Failure Service Log (DB2DIAG.LOG) at the server to see if an error message has been logged.
  - The database agent at the server was terminated due to an abnormal termination of a key database manager process.

- CM_PRODUCT_SPECIFIC_ERROR (20): A product-specific error has been detected, and a description of the error has been stored in the product's system error log. Check that the local APPC subsystem has been started successfully. For AIX SNA Server/6000, to get more information about a product-specific error, it is necessary to check the value of the global variable *errno*. See the section below for more information on possible *errnos* that can be returned. OS/2 Communication.

- The database agent at the server was forced off by the system administrator.
- A database agent could not be started at the server because the *maxagents* database manager configuration parameter has been exceeded. Check the First Failure Service Log (DB2DIAG.LOG) at the server to see if an error message has been logged.
- The database agent at the server was terminated due to an abnormal termination of a key database manager process.

- CM_RESOURCE_FAILURE_RETRY (27): This error occurs when the conversation terminates prematurely (on either the remote or local end) for about the same reason as the NO_RETRY condition described above. The only difference is that the error may not be permanent.

For the most part, CPI Communications return codes are enough to deduce the cause of the error. But when CM_PRODUCT_SPECIFIC_ERROR is returned, additional information is provided.

For AIX SNA Server/6000, the *errno* provides additional information. Below is a list of some of the more common *errnos*. It is NOT a complete list. *Errnos* numbered 101 and higher can be found in the file **/usr/include/luxsna.h**, which contains the SNA Server/6000 specific *errnos*. Most of these *errnos* are translated to CPI-C return codes. Lower-numbered *errnos* are related to AIX problems and can be found in the file **/usr/include/sys/errno.h**. The *errno* number itself is in parentheses.

- EBADF (9): This is a "bad file descriptor" error. If this error occurs when trying to connect to the database, it usually means that the SNA subsystem at the server has not been started, or that there is some problem with the SNA configuration profiles. Check that the SNA subsystem at the server has been started. Check that the link station to the server node can be activated.

- EACCESS (13): This is a "permission denied" error. If this error occurs when trying to connect to the database, it usually means that there is some problem with the SNA configuration profiles.

   For more information on SNA Server/6000 problem resolution, including how to run traces to diagnose problems, refer to *AIX SNA Server/6000 Diagnosis Guide and Messages*.

On HP-UX, for SNAPlus, please refer to the /usr/include/sys/errno.h file for a description of the error.

For OS/2, when CPI Communications returns CM_PRODUCT_SPECIFIC_ERROR, an entry is created in the error log. Information in the error log entry identifies CPIC as the originator. If Communications Manager/2 is installed, Communications Manager records the error in the OS/2 system error log. Refer to the problem determination guide for the specific product, for a complete description of the error, and the recommended action to take.

For more information on CPI Communications errors, refer to *Systems Application Architecture Common Programming Interface Communications Reference*.

- On OS/2, you can view CPI-C return codes and APPC/APPN sense data, if you have installed the DB2 online documentation.
- On AIX, you can view CPI-C return codes and APPC/APPN sense data, by using the command:

    `xview DB2APPC`

NETBIOS

Below is a list of return codes (in hexadecimal) from calls to NetBIOS functions, which the user may encounter. It is NOT a complete list of return codes.

01   Invalid buffer length
03   Invalid command
05   Command timed out
06   Message incomplete
07   Data not received
08   Invalid local session number
09   No resource available
0A   Session closed

    This error could be caused by one of the following:
- The database agent at the server was forced off by the system administrator.
- A database agent could not be started at the server because the *maxagents* database manager configuration parameter has been exceeded. Check the First Failure Service Log (DB2DIAG.LOG) at the server to see if an error message has been logged.
- The database agent at the server was terminated due to an abnormal termination of a key database manager process.

0B   Command cancelled
0D   Duplicate name
0E   Name table full
0F   Command completed (name has active sessions and is now deregistered)
11   Local session table full
12   Session open rejected
13   Invalid name number
14   Remote name not found

    If this error occurs when trying to connect to the database, ensure that:
- The database manager has been successfully started at the server, and NetBIOS support has also been successfully started.
- The server nname specified in the NETBIOS node entry at the client, matches the workstation name specified in the database manager configuration file at the server.

15   Local name not found
16   Name in use on remote node
17   Name deleted
18   Session ended abnormally
19   Name conflict detected
21   Interface busy
22   Too many commands outstanding

23	Invalid adapter
24	Command already completed
26	Command invalid to cancel
30	Name defined by others
34	Environment not defined
36	Max applications exceeded
37	No SAP available
38	Inadequate resources
39	Invalid NCB address
3A	Invalid reset
3B	Invalid NCB DD ID
3C	Segment lock unsuccessful
3F	DD open error
40	OS error detected
4F	Perm ring status
F6	Unexpected CCB error
F8	Adapter open error
F9	Adapter handler error
FA	Adapter check
FB	Code not operational
FC	Open failures
FD	Unexpected close
FF	Command in process

IPX/SPX

Below is a list of *t_errnos* that the user might encounter most frequently. It is NOT a complete list of errors. *Error values* can be found in the file **tiuser.h**. The *t_errno* number itself is in parentheses.

- TBADF (4): Specified file handle does not refer to a transport endpoint.

- TNOADDR (5): Transport provider could not allocate an address.

- TOUTSTATE (6): A function was issued in the wrong sequence.

- TSYSERR (8): A system error occurred during execution of this function.

- TLOOK (9): An asynchronous event has occurred on this connection.

Below is a list of *events* that the user might encounter most frequently. It is NOT a complete list of *events*. *Events* can be found in the file **tiuser.h**. The *event* value is in parentheses.

- T_DISCONNECT (0x0010): Disconnect received.

    This error could be caused by one of the following:

    – The server was not started, or IPX/SPX support on the server was not started. Check the First Failure Service Log (DB2DIAG.LOG) at the server to see if messages have been logged.

- The node directory information specified on the client is incorrect.
- The client or server IPX/SPX communications stack was not installed and/or configured properly.
- The database agent at the server was forced off by the system administrator.
- A database agent could not be started at the server because the *maxagents* database manager configuration parameter has been exceeded. Check the First Failure Service Log (DB2DIAG.LOG) at the server to see if an error message has been logged.
- The database agent at the server was terminated due to an abnormal termination of a key database manager process.

Below is a list of disconnect *reason codes* that the user might encounter most frequently. It is NOT a complete list of *reason codes*. Disconnect *reason codes* can be found in the file **tispxipx.h**. The *reason code* value is in parentheses.

- TLI_SPX_CONNECTION_TERMINATED (0xEC): Disconnect received from remote.

- TLI_SPX_CONNECTION_FAILED (0xED): Connection failure.

For more information on TLI IPX/SPX communication errors, refer to the Transport Layer Interface documentation.

Listed below are the *return values* that the user might encounter most frequently when using DOS, OS/2, or Windows NetWare Connection and Bindery Services. Not ALL *return values* that the user might encounter are listed. *Return values* can be found in the file **nwerror.h**.

0x880F - This is a "NO CONNECTION TO SERVER" error. Ensure that the file server specified is up and running, and can be accessed by the client and server workstations.

0x88FF - If this error occurs when trying to connect to the database, and the function is *NWAttachToFileServer*, ensure that the IPX/SPX protocol support at the client has been installed correctly, and is started.

0x89EF - This is an "INVALID NAME" error.

0x89FC - This is an "UNKNOWN FILE SERVER" error. If either of these errors occur when running the REGISTER command, ensure that:

- An object name has been specified in the database manager configuration file, and the name contains valid characters.
- A file server name has been specified in the database manager configuration file, and the name contains valid characters.

0x89FC - This is a "NO SUCH OBJECT" error. If this error occurs when trying to connect to the database, and the function is *NWScanProperty*, ensure that:

- The network address of the database manager has been successfully registered at the NetWare file server. This is done by running the REGISTER command at the server.

- The object name specified in the IPXSPX node directory entry at the client, matches the object name specified in the database manager configuration file at the server.
- The file server name specified in the IPXSPX node directory entry at the client, matches the file server name specified in the database manager configuration file at the server.

0x89FF - This is a "NO SUCH OBJECT OR BAD PASSWORD" error. Ensure that the password you have specified is defined, and has not expired.

For more information on Connection and Bindery Services function errors, refer to the NetWare Client API for C documentation.

The following is a list of *error generators* that identify what generate errors encountered in the AIX NetWare Connection and Bindery Services environment.

- 0x00 - File server
- 0x10 - API
- 0x20 - API

Listed below are the *errnos* that the user might encounter most frequently when using AIX NetWare Connection and Bindery Services. It is NOT a complete list of errors. *Error codes* can be found in the file **nwerrors.h**.

0x02 - This is a "TRANSPORT OPEN*ired.

  For more i´/SP¸ comwunications

0xFF - This is a "NO RESPONSE FROM FILE SERVER". Ensure that the file server is up and running and accessible by the DB2 server.

For more information on Connection and Bindery Services function errors, refer to the NetWare C Interface for UNIX API reference.

Appendix B. SPM Protocol Violation Records

This section shows the Protocol Violation records generated by the DB2 SPM. These records will be directed to the first failure service log (db2diag.log) and will follow the corresponding message.

Resync Protocol Violation Record

```
/*******************************************************************************
* SQLCSPM_RESYNC_PROTOCOL_VIOLATION_206 - Resync Protocol error.              *
*                                                                             *
* This trace is produced when a protocol error occurs during a Resync Compare *
* States (CS) exchange with a remote LU. The following are examples of SNA    *
* Compare States protocol errors:                                            *
*    * an unknown STATE was found in the SNA CS message                       *
*    * an invalid data message was received from the remote site             *
*    * an invalid LU6.2 verb sequence was detected                           *
*                                                                             *
*******************************************************************************/
#define SQLCSPM_RPV_206_T "SQLCSPM_RESYNC_PROTOCOL_VIOLATION_206"  /* Title.*/
typedef struct sqlcspm_resync_protocol_violation_206
{                                       /*                                    */
  SQLCSPM_LUNAME  luname;               /* Site involved in violation.        */
  SQLCSPM_LUWID   luwid;                /* LUWID for the unit of work.        */
  SQLXA_XID       xid;                  /* XID of the thread.                 */
  char            role;                 /* Role of DB2 in LUW  -- C or P.     */
                                        /*   C for coordinator.               */
                                        /*   P for participant.               */
  char            status;               /* Local thread status -- C, A or I   */
                                        /*   C for COMMIT.                    */
                                        /*   A for ABORT.     .               */
                                        /*   I for INDOUBT.                   */
  SQLCSPS_TRACE   comm_trace;           /* Communication trace of violation.  */
} SQLCSPM_RESYNC_PROTOCOL_VIOLATION_206;
```

489

Syncpoint Protocol Violation Record

```
/*******************************************************************************
* SQLCSPM_2PHASE_PROTOCOL_VIOLATION_208  - SNA syncpoint protocol error.    *
*                                                                           *
* This trace is produced when a protocol error occurs during the SNA        *
* syncpoint process with a remote LU.   The following are examples of SNA   *
* syncpoint protocol errors:                                                *
*    * an unknown SNA PS header was received.                               *
*    * an SNA PS header was received in the wrong sequence.                 *
*    * an invalid LU6.2 verb sequence was detected.                         *
*    * the remote site issued DEALLOCATE TYPE(ABEND_SVC).                   *
*******************************************************************************/
#define SQLCSPM_2PPV_208_T "SQLCSPM_2PHASE_PROTOCOL_VIOLATION_208" /* Title.*/
typedef struct sqlcspm_2phase_protocol_violation_208
{                                   /*                                      */
  SQLCSPM_LUNAME  luname;           /*   Site involved in violation.        */
  SQLCSPM_LUWID   luwid;            /*   LUWID for the unit of work.        */
  SQLXA_XID       xid;              /*   XID of the thread.                 */
  char            role;            /*   Role of DB2 in LUW  -- C or P.      */
                                    /*    C for coordinator.                */
                                    /*    P for participant.                */
  char            local_status;     /*   Local thread status -- C, A or I   */
                                    /*    C for COMMIT.                     */
                                    /*    A for ABORT.      .               */
                                    /*    I for INDOUBT.                    */
  char            partner_status;   /*   Partner thread status -- C, A or I */
                                    /*    C for COMMIT.                     */
                                    /*    A for ABORT.      .               */
                                    /*    I for INDOUBT.                    */
  SQLCSPS_TRACE   comm_trace;       /*   Communication trace of violation.  */
} SQLCSPM_2PHASE_PROTOCOL_VIOLATION_208;
```

XLN Protocol Violation Record

```
/*******************************************************************************
* SQLCSPM_XLN_PROTOCOL_VIOLATION_236   - SNA XLN protocol error.            *
*                                                                           *
* This trace is produced when a protocol error occurs during the SNA        *
* syncpoint process with a remote LU.   The following are examples of SNA   *
* syncpoint protocol errors:                                                *
*    * an unknown SNA PS header was received.                               *
*    * an SNA PS header was received in the wrong sequence.                 *
*    * an invalid LU6.2 verb sequence was detected.                         *
*    * the remote site issued DEALLOCATE TYPE(ABEND_SVC).                   *
*******************************************************************************/
#define SQLCSPM_XPV_236_T "SQLCSPM_XLN_PROTOCOL_VIOLATION_236"     /* Title.*/
typedef struct sqlcspm_xln_protocol_violation_236
{                                   /*                                      */
```

```
   SQLCSPM_LUNAME  luname;            /*  Site involved in violation.        */
   SQLCSPX_XLN_INF xln_inf;           /*  Current XLN information
                                            in LUB.     */
   SQLCSPX_XLN_INF txln_inf;          /*  Temporary XLN information
                                            in LUB.     */
   SQLCSPM_GDSAREA xln;               /*  Current XLN GDS being
                                            processed. */
   SQLCSPS_TRACE   comm_trace;        /*  Communication trace of violation.  */
} SQLCSPM_XLN_PROTOCOL_VIOLATION_236;
```

Appendix C. Incompatibilities between Versions of DataJoiner

This appendix identifies the incompatibilities between DataJoiner version 1 and
DataJoiner version 2. We will discuss changes in system catalog tables and in
functionality, the symptoms that result when you try to use tables and functionality that
have changed, and ways to avoid or counteract these symptoms.

Definition of Incompatibility

Some tables and processes in DataJoiner version 2 are incompatible with their version
1 counterparts. Thus, if an existing application uses any of them, results might be
different than expected, an error might occur, or performance might be reduced. In this
context, the term *application* applies to a broad range of things; for example:

- Application program code
- Third-party utilities
- Interactive SQL queries
- Command and/or API invocation

This appendix does not describe:

- Operations that in version 2 are less likely to generate an error than they did in
  version 1. Such operations will have only a positive impact on existing applications.

- Incompatibilities common to DataJoiner and DB2. For these incompatibilities, see
  the *DB2 SQL Reference.*

System Catalog Tables

This section discusses problems that you would encounter if you use applications
based on version 1 to query and modify version 2 system catalog tables.

Columns and Values in System Catalog Tables

Change

Changes have been made to columns, column attributes, and the acceptable values in
several DataJoiner system catalog tables. This section discusses changes that could
cause problems for applications designed to use the version 1 form of the tables. To
ascertain other changes, you might query the version 1 tables and the version 2 system
catalog views that correspond to them, or you could read descriptions of these tables
and views. The tables are described in the version 1.2 edition of the *Application
Programming and SQL Reference Supplement.* The views are described in the version
2 edition of this book.

SYSCOLUMNS

> **HIGH2KEY:** Non-character values are now in printable format rather
> than binary format.

LOW2KEY: Non-character values are now in printable format rather than binary format.

NULLS: The value D (not null with default) has been changed to N (not nullable).

REMOTE_TYPE: In version 1, values denoted data types of columns of data source tables that DataJoiner referenced by nickname. In version 2, these values are stored in REMOTE_TYPENAME.

SYSINDEXES

CLUSTERRATIO: In version 1, the value in this column was -1 if statistics were not gathered. In version 2, the value is -1 either if statistics are not gathered or if detailed index statistics are gathered. In the latter case, an appropriate value is added to the CLUSTERFACTOR column.

SYSSERVERS

COLSEQ: Deleted from SYSSERVERS. In version 2, this server option is denoted by a value (colseq) in the OPTION column of the SYSCAT.SERVER_OPTIONS catalog view.

CONNECTSTRING: Deleted from SYSSERVERS. In version 2, this server option is denoted by a value (connectstring) in the OPTION column of the SYSCAT.SERVER_OPTIONS catalog view.

CPURATIO: Data type changed from DOUBLE to FLOAT.

DATEFORMAT: Deleted from SYSSERVERS. In version 2, this server option is denoted by a value (DATEFORMAT) in the OPTION column of the SYSCAT.SERVER_OPTIONS catalog view.

FOLDID: Deleted from SYSSERVERS. In version 2, this server option is denoted by a value (fold_id) in the OPTION column of the SYSCAT.SERVER_OPTIONS catalog view.

IORATIO: Data type changed from DOUBLE to FLOAT.

PASSWORD: Deleted from SYSSERVERS. In version 2, this server option is denoted by a value (password) in the OPTION column of the SYSCAT.SERVER_OPTIONS catalog view.

TIMEFORMAT: Deleted from SYSSERVERS. In version 2, this server option is denoted by a value (TIMEFORMAT) in the OPTION column of the SYSCAT.SERVER_OPTIONS catalog view.

<dl>
<dt>TIMESTAMPFORMAT:</dt>
<dd>Deleted from SYSSERVERS. In version 2, this server option is denoted by a value (TIMESTAMPFORMAT) in the OPTION column of the SYSCAT.SERVER_OPTIONS catalog view.</dd>
</dl>

SYSREMOTEUSERS

AUTHID Data type changed from CHAR to VARCHAR.

SYSTABLES

PACKED_DESC: Data type changed from LONGVARCHAR to BLOB.

REL_DESC: Data type changed from LONGVARCHAR to BLOB.

VIEW_DESC: Data type changed from LONGVARCHAR to BLOB.

Symptom

Obviously, a variety of symptoms could occur.

If an existing application does a qualified search on a column that takes a different value than it did before (for example, a search on NULLS in SYSIBM.SYSCOLUMNS for a value of D), the application might react differently than expected.

If an existing application accesses a column whose data type has changed (for example, CPURATIO in SYSIBM.SYSSERVERS), you might retrieve too little data or too much.

Resolution

Review the changes listed above to decide whether they affect your applications and, if so, what corrective action to take (for example, updating the application). So that any problems in accessing or maintaining catalog table data can be avoided, we strongly recommend that you query the version 2 catalog views instead of the tables.

If you need a rough approximation of the degree of clustering, select both CLUSTERRATIO and CLUSTERFACTOR in the SYSCAT.INDEXES catalog view and choose the greater of the two values that you retrieve.

How Users Modify System Catalog Tables

Change

For DataJoiner to perform operations on a specific data source, DataJoiner must associate an identifier (specifically, a server name) with that data source. In version 1, you could create such an association by INSERTing appropriate values into the table SYSIBM.SYSSERVERS. You could also modify an association by UPDATing SYSIBM.SYSSERVERS, and terminate an association by deleting a server name from SYSIBM.SYSSERVERS. In version 2, you use DDL to perform these same operations indirectly. Specifically, you create DataJoiner-to-data source associations with the CREATE SERVER MAPPING statement, modify them with the ALTER SERVER MAPPING statement, and terminate them with the DROP statement. These statements

operate on SYSCAT.SYSSERVERS, a catalog view derived from
SYSIBM.SYSSERVERS. The changes that you make to the view are propagated to
SYSIBM.SYSSERVERS.

For a user to access data sources from DataJoiner, DataJoiner must associate the ID
under which the user connects to DataJoiner with the IDs under which the user
connects to these data sources. In version 1, you could create such an association by
INSERTing appropriate values into the table SYSIBM.SYSREMOTEUSERS. You could
also modify an association by UPDATing SYSIBM.SYSREMOTEUSERS, and terminate
an association by deleting an ID from SYSIBM.REMOTEUSERS. In version 2, you use
DDL to perform these same operations indirectly. S. t.In version 1, ych an by den e, Hese IDs undeRSSô.SYS vSYSS

Appendix D. DataJoiner Education and Service

This section describes the customer education courses and the types of assistance available for DataJoiner.

DataJoiner Education

IBM offers customer education courses that teach you how to install, use, and maintain DataJoiner. The courses are described in this section.

For more information, or to enroll in any IBM class, call 1-800-IBM-TEACH (1-800-426-8322) and refer to the IBM US Course Code. For locations outside the United States, contact your IBM representative.

Course descriptions will also be maintained at the DataJoiner web site. The DataJoiner URL is:

`http://www.software.ibm.com/data/datajoiner/`

Using DataJoiner

IBM US Course Code U4253, World-Wide Code DW20

Duration   2 days

Format   Lecture with classroom exercises.

This course introduces the student to DataJoiner and its powerful multidatabase server capabilities. After completing this course, students should be able to effectively use DataJoiner to perform simple and complex distributed requests. They should also be able to monitor and tune SQL queries, accounting for the capabilities and characteristics of diverse DataJoiner data sources. Areas covered include:

- Global optimization
- Multi-vendor query considerations
- Nicknames
- Basic security
- An introduction to the DataJoiner catalog
- DataJoiner query performance
- The DataJoiner Explain tool
- The DataJoiner Database System Monitor

Who Should Take This Course

This course is appropriate for anyone who will be using, managing, installing, or maintaining a DataJoiner multiple database environment.

Prerequisite

SQL experience. You can obtain this experience by attending the "SQL Workshop," IBM US Course Code U4045.

DataJoiner Administration

IBM US Course Code U4254, World-Wide Code DW21

Duration   3 days

Format   Lecture with classroom exercises.

This course trains the student to install, configure, and manage a secure DataJoiner multidatabase server environment. Areas covered include:

- Installing DataJoiner
- Generating and managing the DataJoiner database
- Configuring DataJoiner
- Enabling DataJoiner client access to remote data sources
- DataJoiner security
- DataJoiner server performance

Who Should Take This Course

This course is appropriate for anyone who will be managing, installing, or maintaining a DataJoiner multiple database environment.

Prerequisite

DataJoiner knowledge or experience. You can obtain this experience by attending "Using DataJoiner," IBM US Course Code U4253.

DataJoiner Service Providers

IBM provides services for DataJoiner that include assistance with planning, installing, and configuring the product. The assistance is customized to your individual environment and takes place in two phases.

First Phase: Planning

The first phase helps you plan for the installation and configuration of DataJoiner, and configure network systems so DataJoiner can communicate optimally with all data sources and clients. It includes:

- Assessing general readiness
- Defining clients
- Defining data sources
- Assessing applications
- Defining backup and recovery strategies for DataJoiner
- Configuring DataJoiner database parameters
- Identifying test queries for system validation
- Defining security requirements

Second Phase: Implementation

The second phase focuses on implementation of the plan developed in the planning phase described above. It includes:

- Installing DataJoiner
- Configuring data sources
- Providing access to data source tables and views

- Installing and configuring remote clients
- Validating and documenting the environment
- Providing final turnover to the customer

At the end of this phase, active remote and local clients can access multiple data sources through DataJoiner.

DataJoiner services can be combined with replication services if you are interested in replicating data across a heterogeneous database environment. For more information about DataJoiner and replication services, contact your IBM representative or see the DataJoiner web page. The DataJoiner URL is:

`http://www.software.ibm.com/data/datajoiner/`

Index

-10002¢⩔

We'd Like to Hear from You

DB2 DataJoiner
Messages and Problem
Determination Guide
Version 2 Release 1

Publication No. SC26-9149-00

Please use one of the following ways to send us your comments about this book:

- Mail—Use the Readers' Comments form on the next page. If you are sending the form from a country other than the United States, give it to your local IBM branch office or IBM representative for mailing.

- Fax—Use the Readers' Comments form on the next page and fax it to this U.S. number: 800-426-7773 or (408) 463-4393.

- Electronic mail—Use the following network ID:

    – Internet: COMMENTS@VNET.IBM.COM

  Be sure to include the following with your comments:

    – Title and publication number of this book
    – Your name, address, and telephone number if you would like a reply

Your comments should pertain only to the information in this book and the way the information is presented. To request additional publications, or to comment on other IBM information or the function of IBM products, please give your comments to your IBM representative or to your IBM authorized remarketer.

IBM may use or distribute your comments without obligation.

Readers' Comments

DB2 DataJoiner
Messages and Problem
Determination Guide
Version 2 Release 1

Publication No. SC26-9149-00

How satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Technically accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐
Grammatically correct and consistent	☐	☐	☐	☐	☐
Graphically well designed	☐	☐	☐	☐	☐
Overall satisfaction	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

May we contact you to discuss your comments? ☐ Yes ☐ No

Name

Address

Company or Organization

Phone No.

Readers' Comments
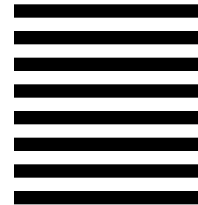SC26-9149-00

IBM®

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

International Business Machines Corporation
Department W92/H3
PO Box 49023
San Jose, CA 95161-9945

Fold and Tape   **Please do not staple**   Fold and Tape

SC26-9149-00

IBM ®

Spine information:

IBM   DB2 DataJoiner   Messages and Problem
Determination Guide   *Version 2 Release 1*