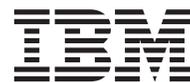IBM® DB2 Universal Database™

Message Reference
Volume 2

Version 8

IBM® DB2 Universal Database™

Message Reference
Volume 2

Version 8

Before using this information and the product it supports, be sure to read the general information under *Notices*.

Contents

About This Guide

The purpose of this book is to list the messages returned by various components of DB2.

Who Should Use This Guide

The *Message Reference* is designed to be used by anyone using DB2 and requiring more detailed information about returned messages.

How This Guide Is Structured

This guide lists the possible error messages returned by the various components of DB2.

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 English national language format. These may be different on your display, depending on the national language format used by your system.

Chapter 1. Introduction to Messages

It is assumed that you are familiar with the functions of the operating system where DB2 is installed. You can use the information contained in the following chapters to identify an error or problem and resolve the problem by using the appropriate recovery action. This information can also be used to understand where messages are generated and logged.

Message Structure

Message identifiers consist of a three character message prefix, followed by a four or five digit message number, followed by a single letter suffix. For example, *SQL1042C*. For a list of message prefixes, see "Information Available Online" and "Other DB2 Messages" on page 3. The single letter suffix describes the severity of the error message.

In general, message identifiers ending with a *C* are severe messages. Those ending with an *E* are urgent messages. Message identifiers ending with an *N* are error messages, those ending with a *W* are warning messages and an *I* indicates an informational message.

For ADM messages, message identifiers ending with a *C* are severe messages. Those ending with an *E* are urgent messages. Message identifiers ending with a *W* are important messages and an *I* are indicates an informational messages.

For SQL messages, 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.

Information Available Online

The following DB2 messages are accessible from the operating system command line:

Prefix Description

ASN   messages generated by DB2 Replication

CCA   messages generated by the Client Configuration Assistant

CLI   messages generated by Call Level Interface

DBA   messages generated by the Database Administration tools

DBI   messages generated by installation and configuration

DBT	messages generated by the Database tools
DB2	messages generated by the command line processor
DWC	messages generated by the Data Warehouse Center
GSE	messages generated by the DB2 Spatial Extender
ICC	messages generated by the Information Catalog Center
SAT	messages generated in a satellite environment
SPM	messages generated by the sync point manager
SQL	messages generated by the database manager when a warning or error condition has been detected.

As well, the message text associated with SQLSTATE values is available online.

Message identifiers consist of a three character message prefix (see above list), followed by a four or five digit message number. The single digit letter at the end which describes the severity of the error message is optional.

To access help on these error messages, enter the following 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, use the following command (on unix-based systems and others which support 'more'):

```
db2 "? XXXnnnnn" | more
```

Help can also 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 DB2 message help in this mode, type the following at the command prompt:

```
?
```

Note: If the message text exceeds the length of the screen, users with non-graphical workstations can pipe the output to the 'more' (on unix-based systems) program or redirect the output to a file which can then be browsed.

The message text associated with a given SQLSTATE value can be retrieved by issuing:

```
db2 "? nnnnn"
```

  or

```
db2 "? nn"
```

where *nnnnn* is a five digit SQLSTATE (alphanumeric) and *nn* is the two digit SQLSTATE class code (first two digits of the SQLSTATE value).

Other DB2 Messages

Some DB2 components return messages that are not available online or are not described in this manual. Some of the message prefixes may include:

ADM   messages generated by many DB2 components. These messages are written in the Administration Notification log file and are intended to provide additional information to System Administrators.

> **Note:** Although the ADM messages are available for reference, the messages are not available online.

AUD   messages generated by the DB2 Audit facility.

DIA   diagnostics messages generated by many DB2 components. These messages are written in the diagnostics log file db2diag.log, and are intended to provide additional information for users and DB2 service personnel when investigating errors.

GOV   messages generated by the DB2 governor utility.

In most cases, these messages provide sufficient information to determine the cause of the warning or error. For more information on the command or

utility that generated the messages, please refer to the appropriate manual where the command or utility is documented.

Other Message Sources

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

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

Chapter 2. 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 . Some SQLCODE values have associated SQLSTATE values. SQLSTATE values 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.

SQL0000 - SQL0099

SQL0000W   **Statement processing was successful.**

Explanation: The SQL statement executed successfully, unless a warning condition occurred.

User Response: 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 *Application Development Guide*.

sqlcode: 0

sqlstate: 00000, 01003, 01004, 01503, 01504, 01506, 1509, 01517

SQL0001N   **Binding or precompilation did not complete successfully.**

Explanation: For reasons specified in the previous messages, the bind or precompile request was unsuccessful.

No package is created.

User Response: 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.**

Explanation: For reasons specified in the previous messages, the bind file name cannot be used as specified.

No package is created.

User Response: 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.

Explanation: For reasons specified in the previous messages, the database name cannot be used as specified.

No package is created.

User Response: Refer to the messages in the message file. Resubmit the command.

SQL0004N   The password is not valid.

Explanation: The password contains characters that are not valid or the password is too long.

No package is created.

User Response: Resubmit the command with a valid password.

SQL0005N   The message file name is not valid.

Explanation: For reasons specified in the previous messages, the message file name cannot be used as specified.

No package is created.

User Response: 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.

Explanation: The value of the datetime format parameter is not in the valid range of 0 through 3.

No package is created.

User Response: Resubmit the command with a valid format parameter.

SQL0007N   The character *character* **following** *text* **is not valid.**

Explanation: 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.

Federated system users: some 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.

User Response: 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.**

Explanation: 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.

User Response: Examine the syntax of the host variable declaration.

SQL0009W   An attempt to override a precompiler option was ignored.

Explanation: An attempt was made to override a precompiler option.

The option is ignored.

User Response: Ensure that all precompiler options are specified correctly.

SQL0010N **The string constant beginning with** *string* **does not have an ending string delimiter.**

Explanation: The statement contains a string constant, beginning with *string*, that is not terminated properly.

The statement cannot be processed.

User Response: Examine the statement for missing apostrophes in the indicated string constant.

sqlcode: -10

sqlstate: 42603

SQL0011N **The comment is not terminated.**

Explanation: The comment is not terminated properly.

The statement cannot be processed.

User Response: Examine the statement for a missing or extra comment delimiter in the indicated comment.

SQL0012W **Correlation without qualification has occurred for the column** *column***.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Provide a valid cursor name, statement name, database name, or authorization ID.

SQL0014N **The source file name is not valid.**

Explanation: 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.

User Response: Ensure that the source file name is correct.

SQL0015N **The host variable data type** *token-1* **is not valid. Use** *token-2* **instead.**

Explanation: 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'.

User Response: Replace the current data type of the host variable with the data type specified in the message.

SQL0017N **A RETURN statement must be specified and executed in an SQL function or method.**

Explanation: The SQL function or method either does not contain a RETURN statement, or the function or method did not end with the execution of a RETURN statement.

User Response: Ensure the function or method

executes a RETURN statement.

sqlcode: -17

sqlstate: 42632

SQL0020W   **Bind or precompile option(s)**
(name or value) *option-name(s)* **are**
not supported by the target
database and will be ignored.

Explanation: This warning is returned in the
following situations:

- one or more options that were specified at
  precompile/bind time are not supported by
  the target DBMS
- the option value(s) of one or more options that
  were specified at precompile/bind time are not
  supported by the target DBMS

The unsupported options/values will be ignored.

Note that the option names in the message
correspond to terms used by DB2 UWO.

Enter ″db2 ? bind″ or ″db2 ? prep″ for a list of
options supported by DB2 UWO.

User Response: Ensure the specification of bind
or precompile *option-name(s)* or the associated
option value(s) was intended while connected to
this DBMS.

SQL0021W   **Invalid precompiler option** *option*
is ignored.

Explanation: The option specified in the
message is not a valid precompiler option.

The option is ignored.

User Response: Ensure that all precompiler
options are specified correctly.

SQL0022W   **Duplicate precompiler option**
option **is ignored.**

Explanation: The precompiler option *option* is a
duplicate.

The option is ignored.

User Response: Ensure that all precompiler
options are specified only once.

SQL0023N   **The database name is not valid.**

Explanation: The specified database name is not
a valid name.

Precompilation is terminated.

User Response: Ensure that the database name
is spelled correctly and follows the rules for
short identifiers.

SQL0024N   **A database name was not**
specified.

Explanation: No database name was provided
to proceed with precompilation.

Precompilation is terminated.

User Response: Provide a database name.

SQL0025W   **Binding or precompilation**
completed with warnings.

Explanation: Binding or precompilation
succeeded, but warnings were issued. A package
and/or bindfile was created as requested in the
command.

User Response: Refer to the messages in the
message file. If necessary, correct the problems
and resubmit the command.

SQL0026N   **The password is not valid.**

Explanation: The specified password is not a
valid password.

Precompilation is terminated.

User Response: Ensure that the password
provided follows the rules for valid passwords.

SQL0028C   **The release number of the bind**
file is not valid.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: A source file name was not provided to proceed with precompilation.

Precompilation is terminated.

User Response: Specify a source file name.

SQL0031C   File *name* could not be opened.

Explanation: The file *name* is required but could not be opened.

Precompilation is terminated.

User Response: 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*.

Explanation: While reading or writing file *name*, an error was encountered.

Precompilation is terminated.

User Response: Retry the precompilation.

SQL0033N   *name* is not a valid bind file.

Explanation: The bind file *name* specified is not a bind file.

Binding is terminated.

User Response: Ensure that the correct file name was specified.

SQL0034N   A bind file name was not specified.

Explanation: No bind file name was specified to proceed with binding.

Binding is terminated.

User Response: Provide a bind file name.

SQL0035N   The file *name* cannot be opened.

Explanation: The message file *name* could not be opened.

Binding or precompilation has been terminated.

User Response: Ensure that the system can access the file.

SQL0036N   The syntax of the file name *name* is not valid.

Explanation: 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.

User Response: Ensure that the specified file name is correct.

SQL0037W **The syntax of the message file** *name* **is not valid.**

Explanation: The message file name *name* is not syntactically correct for this function.

The system redirects the output to the standard output device.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Refer to the other messages which accompanied this one to resolve the errors which occurred. Resubmit the command.

SQL0051N **The space required to hold all SQL statements for a single program has exceeded its allowed maximum.**

Explanation: The space required by all SQL statements in the program cannot fit into the column SECT_INFO in SYSIBM.SYSPLAN.

Precompilation is terminated.

User Response: Either simplify the program, or split the program into smaller, separate programs, or do both.

SQL0053W **No SQL statements were found in the program.**

Explanation: The specified source file contains no SQL statements.

If binding, an empty package is created.

User Response: Ensure that you are precompiling or binding the correct program.

SQL0055N **Source input file is empty.**

Explanation: The program source input file contained no data.

Precompilation is terminated.

User Response: Ensure that the correct input file was specified.

SQL0056N **An SQLSTATE or SQLCODE variable declaration is in a nested compound statement.**

Explanation: An SQLSTATE or SQLCODE variable declaration is in a nested compound statement instead of the outermost compound statement in the SQL routine.

The statement cannot be processed.

User Response: Declare the SQLSTATE and SQLCODE variables only in the outermost compound statement of the SQL routine.

sqlcode: -56

sqlstate: 42630

SQL0057N **A RETURN statement in an SQL function or method must include a return value.**

Explanation: A RETURN statement is specified in the SQL function or method without specifying a value to return.

The statement cannot be processed.

User Response: Specify a value on the RETURN statement.

sqlcode: -57

sqlstate: 42631

SQL0058N **The data type of the RETURN statement value in an SQL procedure must be INTEGER.**

Explanation: A RETURN statement is specified in the SQL procedure with a value or expression that is not of the INTEGER data type.

The statement cannot be processed.

User Response: Specify a value on the RETURN statement that has a data type of INTEGER.

sqlcode: -58

sqlstate: 428F2

SQL0060W **The *name* precompiler is in progress.**

Explanation: 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.

User Response: No action is required.

SQL0061W **The binder is in progress.**

Explanation: This message is written to the standard output device when the binder begins processing.

Processing continues.

User Response: No action is required.

SQL0062W   Starting INCLUDE of file *name*.

Explanation: An INCLUDE statement has been encountered. The precompiler is now processing the INCLUDE file.

User Response: No action is required.

SQL0063W   Completed INCLUDE of file *name*.

Explanation: The precompiler has completed processing the INCLUDE file. It will resume processing the file which contained the INCLUDE statement.

User Response: No action is required.

SQL0064N   File *name* **directly or indirectly INCLUDEs itself.**

Explanation: 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.

User Response: Check the nesting of the INCLUDE files, and remove any cycles.

SQL0065N   An unexpected end-of-line was encountered in a host variable declaration.

Explanation: A host variable declaration has an invalid syntax. An end-of-line was encountered before the declaration was complete.

The statement cannot be processed.

User Response: Examine the syntax of the host variable declaration.

SQL0078N   Parameter names must be specified for routine *routine-name*.

Explanation: The routine *routine-name* does not have parameter names specified for all parameters. When the routine is defined with LANGUAGE SQL or SQLMACRO a parameter

name is required for each parameter defined.

The statement cannot be processed.

User Response: Ensure that each defined parameter includes a parameter name.

sqlcode: -78

sqlstate: 42629

SQL0079N   The Schema name for the declared global temporary table or index *name* **must be SESSION, not** *schema-name*.

Explanation: The schema name *name* for a declared temporary table or an index on a declared global temporary table must be SESSION. The statement is specifying an explicit schema-name of *schema-name* for the declared global temporary table or index on a declared global temporary table. This is not allowed.

The statement cannot be processed.

User Response: Change the statement in one of the following ways:

* Change the schema name to SESSION.
* Remove the schema name and let DB2 use the default value, SESSION.

sqlcode: -79

sqlstate: 428EK

SQL0081N   SQLCODE *sqlcode* **returned during precompilation/bind.**

Explanation: While precompiling or binding the program, an unexpected SQLCODE *sqlcode* was returned from the database manager.

The statement cannot be processed.

User Response: Inspect the SQLCODE to determine the problem and take the appropriate action.

SQL0082C   **An error has occurred which has terminated processing.**

Explanation: A previous, non-SQL error has caused processing to terminate.

Precompile/bind/rebind is terminated. No package was created.

User Response: Correct the situation that caused the previous error and retry the operation.

SQL0083C   **A memory allocation error has occurred.**

Explanation: During processing, there was not enough memory to continue processing.

User Response: Possible solutions include:

- Ensure that your system has sufficient real and virtual memory.
- Remove background processes.

If you continue to receive this message after attempting the suggested solutions, contact IBM Customer Support.

SQL0084N   **An EXECUTE IMMEDIATE statement contains a SELECT or VALUES statement.**

Explanation: A SELECT or VALUES statement was used in an EXECUTE IMMEDIATE statement.

The statement cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: Use a different name for the statement.

SQL0086C   **A memory allocation error has occurred.**

Explanation: During processing, there was not enough memory to continue processing.

User Response: 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.**

Explanation: 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.

User Response: Replace the structure reference with an atomic host variable, or with a fully qualified structure field name.

SQL0088N   **Host variable** *name* **is ambiguous.**

Explanation: Host variable *name* cannot be uniquely identified. More than one host variable with the same qualification can be found.

The statement cannot be processed.

User Response: Qualify the host variable further, or rename it if it is already fully qualified.

SQL0089N   **Processing ended after 100 errors were found.**

Explanation: The precompiler or binder quits processing after 100 errors were found.

User Response: 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.**

Explanation: The precompilation or binding has terminated with the stated number of warnings and errors.

Precompilation or binding is terminated.

User Response: 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.**

Explanation: Because of previous errors, no package was created.

User Response: Correct the errors and retry the precompilation or bind.

SQL0093N **The end-of-input was encountered before the statement terminator for EXEC SQL.**

Explanation: While processing an SQL statement, the end of source was encountered before the statement terminated.

Precompilation is terminated.

User Response: Ensure that the SQL statement terminates properly.

SQL0094N **Binding was ended because of user interrupt request.**

Explanation: Binding was terminated because the user may have pressed the interrupt key sequence.

Processing is terminated. No package is created.

User Response: 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.**

Explanation: Because of previous errors, no bind file was created.

No bind file is created.

User Response: Correct the errors and retry the precompilation.

SQL0097N **Variables or parameters of LONG VARCHAR or LONG VARGRAPHIC data types are not supported in SQL routines.**

Explanation: SQL routines (procedures, functions or methods) do not support variables or parameters of LONG VARCHAR or LONG VARGRAPHIC data types.

User Response: Do not use variables or parameters of LONG VARCHAR or LONG VARGRAPHIC data types in SQL routines. For LONG VARCHAR, use VARCHAR with an explicit length. For LONG VARGRAPHIC, use VARGRAPHIC with an explicit length.

sqlcode: -97

sqlstate: 42601

SQL0100 - SQL0199

SQL0100W No row was found for FETCH, UPDATE or DELETE; or the result of a query is an empty table.

Explanation: 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.

User Response: No action is required. Processing can continue.

sqlcode: +100

sqlstate: 02000

SQL0101N The statement is too long or too complex.

Explanation: 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.

Federated system users should also check to see if the statement:

- Exceeds either a federated server 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 different codepages, the result of the conversion is exceeding the length limit.

User Response: 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.
- Federated system 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 the federated server, 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.

Explanation: 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 32672 bytes. Character strings with lengths greater than 32672 bytes or graphic strings with lengths greater than 16336 characters can be specified only through assignment from host variables. Note that other servers in the DB2 family of products may specify a different size limit for character strings. Consult the documentation for the appropriate DB2 product for details.
- Federated system 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 OS/390 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.

User Response: 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.

Federated system 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.**

Explanation: The indicated *literal* begins with a digit but is not a valid integer, decimal, or floating point literal.

Federated system users: a data source-specific

literal representation error has occurred in a pass-through session.

The statement cannot be processed.

User Response: Correct the invalid numeric literal. Federated system 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***.**

Explanation: 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.

User Response: 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.**

Explanation: The statement contains a string constant beginning with *string* that is not valid.

The statement cannot be processed.

User Response: Specify the correct format of the string constant. Check for graphic string, paired delimiters, and an even number of bytes within the string.

Federated system users, see the problem determination guide to determine which data source is causing the error.

sqlcode: -105

sqlstate: 42604

SQL0106N   SQL statement begins properly but is incomplete.

Explanation: The SQL statement was correct up to the point where no more input was found. This may be caused by failure to properly end a literal. String literals need a final quote mark.

Processing of this SQL statement has ended.

User Response: Check that the statement has all the necessary parts to complete the desired function and that all clauses are complete.

For PL/I: Check that the SQL statement is complete before the semicolon. For Assembler: Check that the continuation rules are properly followed. (A nonblank character must be in column 72, and continued lines must start in column 16 or beyond.)

For COBOL: Check that the SQL statement is complete before END-EXEC.

sqlcode: -106

sqlstate: 42601, 42603

SQL0107N   The name *name* **is too long. The maximum length is** *length***.**

Explanation: The name returned as *name* is too long. The maximum length permitted for names of that type is indicated by *length*.

The names for indexes and constraints can be a maximum length of 18 bytes. The names for columns can be a maximum length of 30 bytes. The names for savepoints, tables, views and aliases can be a maximum length of 128 bytes. (This does not include any escape characters, if present.)

SQL variable names, condition names and labels in SQL routines must not exceed 64 bytes in length.

A maximum of 30 bytes is permitted for a schema name (object qualifier), with the exception of user-defined types, which allow a maximum of 8 bytes for a schema name.

Host variable names must not exceed 255 bytes in length.

For the SQL CONNECT statement, an application server name of up to 18 characters in length will be accepted at precompilation 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.

A package version identifier must not exceed 64 bytes in length.

Federated system 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.

User Response: Choose a shorter name or correct the spelling of the object name.

Federated system 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.**

Explanation: 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.

User Response: Ensure that the name for the object is qualified correctly.

sqlcode: -108

sqlstate: 42601

SQL0109N The *clause* clause is not allowed.

Explanation: 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.

Federated system users: In a pass-through session, a data source-specific restriction might have been violated.

The statement cannot be processed.

User Response: Correct the SQL statement by removing the clause.

Federated system 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 restriction has been violated, and adjust the failing statement as needed.

sqlcode: -109

sqlstate: 42601

SQL0110N *string* is an invalid hexadecimal constant.

Explanation: 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.

User Response: Correct the constant and resubmit the statement.

sqlcode: -110

sqlstate: 42606

SQL0111N The column function *name* does not include a column name.

Explanation: 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.

User Response: 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* includes a column function, a scalar fullselect, or a subquery.

Explanation: The operand of a column function cannot include a:

- column function
- scalar fullselect
- subquery.

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.

User Response: Correct the use of the column function to eliminate the invalid expression and try again.

sqlcode: -112

sqlstate: 42607

SQL0113N *identifier* **contains a character that is not allowed or does not contain any characters.**

Explanation: An SQL-variable-name, parameter-name or condition-name *identifier* contains an invalid character. Only characters that are valid for an SQL ordinary identifier are allowed. Note that because the identifier is delimited, folding is not performed and uppercase and lowercase letters are treated as distinct from each other.

User Response: Correct the identifier and resubmit the statement.

sqlcode: -113

sqlstate: 42601

SQL0117N **The number of values assigned is not the same as the number of specified or implied columns.**

Explanation:
- 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.

User Response: 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.**

Explanation: 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 DB2 Connect.

User Response: 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.**

Explanation: 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.

If the NODENUMBER or PARTITION functions are specified in the HAVING clause, then all partitioning key columns of the underlying table are considered to be in the HAVING clause.

The statement cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

If the NODENUMBER or PARTITION functions are specified in the SELECT clause, then all

partitioning key columns of the underlying table are considered to be in the SELECT clause.

The statement cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: This error is only applicable to releases of DB2 prior to DB2 Version 2 and hosts accessed through DB2 Connect.

sqlcode: -127

sqlstate: 42905

SQL0129N **The statement contains too many table names (the maximum is 15).**

Explanation: 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.

User Response: 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 DB2 Connect.

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.**

Explanation: 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*.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: The use of the string *name* is not permitted.

An expression resulting in a string data type with a maximum length greater than 255 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.

Federated system users: in a pass-through session, a data source-specific restriction can cause this error. See the SQL Reference documentation for the failing data sources.

The statement cannot be processed.

User Response: The requested operation on the string is not supported.

Note: If it is unclear as to how the 255 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

SQL0135N   **The input for a long string column in an INSERT statement or UPDATE statement must be from a host variable or be the keyword NULL.**

Explanation: The UPDATE or INSERT is using constants, column names, or subqueries where it should be using NULL or a host variable.

A long string column is either a LONG VARCHAR, LONG VARGRAPHIC, VARCHAR(n) where n is greater than 254 but less than or equal to 32767, or VARGRAPHIC(n)

where n is greater than 127 but less than or equal to 16383.

User Response: Refer to the DB2 for VM Application Programming manual for information on the use of long strings. Correct the statement. Try again.

sqlcode: -135

sqlstate: 56033

SQL0137N   **The length resulting from** *operation* **is greater than** *maximum-value***.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: The clause in the column specification is redundant.

The statement was processed successfully, but the redundant clause was ignored.

User Response: Correct the column specification.

sqlcode: +139

sqlstate: 01589

SQL0142N   **The SQL statement is not supported.**

Explanation: An embedded SQL statement that may be valid for other IBM relational database products is not supported by the database manager.

Federated system 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.

User Response: Change the syntax of the SQL statement or remove the statement from the program.

Federated system 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.**

Explanation: 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.

User Response: Change the syntax of the SQL statement or remove the statement from the program.

SQL0150N   **The view, typed table, materialized query table, or staging table in the INSERT, DELETE, or UPDATE statement is a view, typed table, materialized query table, or staging table for which the requested operation is not permitted.**

Explanation: The view, typed table, materialized query table, or staging table named in the INSERT, UPDATE, or DELETE statement is defined so that the requested insert, update, or delete operation cannot be performed.

A 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).
- Federated system users: a data source specific limitation that causes the column not to be updatable

Note that these conditions do not apply to subqueries of the SELECT statement.

You cannot insert rows directly into a typed table defined on a not instantiable structured type. Subtables of this table may allow inserts.

In general, system maintained materialized query tables and staging tables do not allow the insert, update or delete operations.

The statement cannot be processed.

User Response: The requested function cannot be performed on the view, system maintained materialized query table, or staging table.

Federated system 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.**

Explanation: 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-updatable column of a system catalog, or a column explicitly marked as READ ONLY.

Federated system users should check to see if some other data source specific limitation prevents the column from being updated.

The statement cannot be processed.

User Response: If the specified column is derived from a scalar function, expression,

keyword, or non updatable column, omit the column from the set clause of the update or the column of the insert. For a list of updatable catalogs (and the updatable columns) see the *SQL Reference*.

Federated system users: if the reason is unknown, isolate the problem to the data source failing the request (see the problem determination guide) and examine the object definition and the update restrictions for that data source.

sqlcode: -151

sqlstate: 42808

SQL0152N **Constraint** *constraint-name* **is a** *actual-constraint-type* **constraint instead of a** *expected-constraint-type* **constraint.**

Explanation: An attempt was made to alter or drop constraint *constraint-name*, but it is defined as a *expected-constraint-type* constraint, not as the specified *actual-constraint-type* constraint.

User Response: Verify the name and type of constraint to be altered or dropped.

sqlcode: -152

sqlstate: 42809

SQL0153N **The CREATE VIEW statement or common table expression does not include a column list.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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. The RUNSTATS and LOAD utilities are also applicable only to tables, not to views.

Federated system users: Some utilities and statements are not supported in a federated environment. See the Administration Guide for more information.

The statement or utility cannot be processed.

User Response: 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.

Federated system users: Verify that the object is not a nickname.

sqlcode: -156

sqlstate: 42809

SQL0157N   *name* **is not allowed in a FOREIGN KEY clause because it does not identify a base table.**

Explanation: The object *name* was identified in a FOREIGN KEY clause of a CREATE or ALTER TABLE statement. A FOREIGN KEY clause must identify a base table.

The statement cannot be processed. The specified table is not created or altered.

User Response: Correct the statement to specify a base table name in the FOREIGN KEY clause.

If an alias name was supplied, ensure that the alias resolves to a base table.

sqlcode: -157

sqlstate: 42810

SQL0158N   **The number of columns specified for *name* is not the same as the number of columns in the result table.**

Explanation: The identifier *name* could identify:
- a view named in a CREATE VIEW statement
- a table name of a common table expression
- a correlation name of a nested table expression
- a materialized query table named in a CREATE TABLE or ALTER TABLE statement
- a function named in a CREATE FUNCTION statement
- a method named in a CREATE METHOD statement.
- a staging table name in a CREATE STAGING TABLE statement

The number of column names specified must be equal to the number of columns in the result table of the associated fullselect. If *name* is a staging table and the associated materialized query table has group by clause, the number of column names specified must be 2 more than the number of columns in the materialized query table for which the staging table is being defined.

If the associated materialized query table has no group by clause, the number of column names specified must be 3 more than the number of columns in the materialized query table.

The statement cannot be processed.

User Response: Correct the syntax so that the list of column names defined for *name* specify a name for each column of the result table.

sqlcode: -158

sqlstate: 42811

SQL0159N **The statement references** *object* **which identifies a(n)** *object-type* **rather than a(n)** *expected-object-type***.**

Explanation: The object *object* specified as part of the statement or command refers to an object of type *object-type* instead of the expected type *expected-object-type*.

The type of the object provided with the statement or command must match the type identified by *expected-object-type*. For example, if the statement is DROP ALIAS *PBIRD.T1*, then *PBIRD.T1* must be an alias name.

User Response: Change the statement or command to properly match the type of object identified by *expected-object-type*.

sqlcode: -159

sqlstate: 42809

SQL0160N **The WITH CHECK OPTION clause is not valid for the specified view.**

Explanation: The WITH CHECK OPTION clause 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).
- A view on which the specified view definition is dependent has an INSTEAD OF trigger defined.
- A view on which the specified view definition is dependent contains a text search function.

Federated system users: the WITH CHECK option is not supported on views that reference updatable nicknames.

The statement cannot be processed. The specified view was not created.

User Response: Remove the WITH CHECK OPTION clause 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.**

Explanation: The WITH CHECK OPTION clause 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.

User Response: 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 updatable columns of catalogs.

Federated system users, if the reason is unknown, isolate the problem to the data source failing the request (see the problem determination guide) and examine the object definition and the view definition for that data source.

sqlcode: -161

sqlstate: 44000

SQL0170N   The number of arguments for function *name* **is incorrect.**

Explanation: The specified scalar function *name* has either too few or too many arguments.

The statement cannot be processed.

User Response: 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 routine** *name* **is incorrect.**

Explanation: The data type, length or value of argument *n* of routine *name* is incorrect.

The statement cannot be processed.

User Response: Ensure the arguments of the routine conform to the rules of the routine.

sqlcode: -171

sqlstate: 42815

SQL0172N   *name* **is not a valid function name.**

Explanation: The SQL statement includes an unknown scalar function.

The statement cannot be processed.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Federated system users: the problem might be due to a date/time representation problem at the data source. If the reason is unknown, isolate the problem to the data source failing the request Troubleshooting 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.**

Explanation: The string representation of a date, time or timestamp value contains a value that is out of range.

This error could be caused by accessing a datetime value from an application using a territory code whose datetime format is different than the one used by the application that created it. For example, a string datetime value stored in the format dd/mm/yyyy will not be valid when read by an application expecting its format to be mm/dd/yyyy.

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.

Federated system users: the problem might be due to a date/time representation problem at the data source. Refer to the data source's documentation for the ranges for date and time values at the data source. If the reason is unknown, isolate the problem to the data source failing the request (see the Troubleshooting Guide) and examine the date/time representation restrictions for that data source.

The statement cannot be processed.

User Response: Ensure that the value is within the valid range, and the application's datetime format is the same as the string value.

sqlcode: -181

sqlstate: 22007

SQL0182N   **An expression with a datetime value or a labeled duration is not valid.**

Explanation: The specified expression contains an improperly used date, time, or timestamp value or labeled duration.

The statement cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: An error indication from the operating system was detected while retrieving date/time information.

User Response: Ensure that the system TOD clock and timezone settings are correct.

sqlcode: -187

sqlstate: 22506

SQL0190N **ALTER TABLE** *table-name* **specified attributes for column** *column-name* **that are not compatible with the existing column.**

Explanation: The attributes specified in the ALTER COLUMN clause for the column *column-name* of the table *table-name* in an ALTER TABLE statement are not compatible with the attributes of the existing column. The error was returned for one of the following reasons.

- If a SET DATA TYPE clause is specified, the existing column:
  - is not of data type VARCHAR or
  - has a length that is greater than the length specified in the clause.
- If a SET EXPRESSION clause is specified, the existing column is not defined as generated using an expression.

The statement cannot be processed.

User Response: Make the attributes specified compatible with the existing column, remove the attribute specification, or specify a different column name.

sqlcode: -190

sqlstate: 42837

SQL0191N **Error occurred because of a fragmented MBCS character.**

Explanation: Possible causes:

1. User 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.

2. A scalar function, such as SUBSTR or POSSTR has truncated a multi-bytes string incorrectly. For these functions, start and length values must be correct in terms of bytes, in the context of the database code page. For a Unicode database, a common cause of this could be that the start or length for a UTF-8 string are incorrect.

3. A scalar function such as TRANSLATE has may have changed the multi-byte string.

Federated system users: this situation can also be detected by the data source.

User Response:

1. Correct the input data and try again.

2. Change the start and length values such that when characters are converted to the database code page, the multi-byte characters are not truncated incorrectly.

3. Correct the TRANSLATE in error.

Federated system users: if the data is correct, isolate the problem to the data source failing the request (see the Troubleshooting 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.

Explanation: 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.

User Response: 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 A qualified column name is not allowed in the ORDER BY clause.

Explanation: The ORDER BY clause of a fullselect that includes a set operator (UNION, EXCEPT, INTERSECT) cannot have qualified column names.

User Response: Ensure that all column names in the ORDER BY clause are unqualified.

sqlcode: -197

sqlstate: 42877

SQL0198N The statement string of the PREPARE or EXECUTE IMMEDIATE statement is blank or empty.

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: 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

SQL0200 - SQL0299

SQL0203N   A reference to column *name* **is ambiguous.**

Explanation: The column *name* is used in the statement and there is more than one possible column to which it could refer. This could be the result of:

- two tables specified in a FROM clause that have columns with the same name
- the ORDER BY clause refers to a name that applies to more than one column in the select list
- a reference to a column from the subject table in a CREATE TRIGGER statement does not use the correlation name to indicate if it refers to the old or new transition variable.

The column name needs further information to establish which of the possible table columns it is.

The statement cannot be processed.

User Response: Add a qualifier to the column name. The qualifier is the table name or correlation name. A column may need to be renamed in the select list.

sqlcode: -203

sqlstate: 42702

SQL0204N *name* **is an undefined name.**

Explanation: This error is caused by one of the following:

- The object identified by *name* is not defined in the database.
- A data type is being used. This error can occur for the following reasons:
  - If *name* is qualified, then a data type with this name does not exist in the database.
  - If *name* is unqualified, then the user's function path does not contain the schema to which the desired data type belongs.
  - The data type does not exist in the database with a create timestamp earlier than the time the package was bound (applies to static statements).

- If the data type is in the UNDER clause of a CREATE TYPE statement, the type name may be the same as the type being defined, which is not valid.

- A function is being referenced in one of:
  - a DROP FUNCTION statement
  - a COMMENT ON FUNCTION statement
  - the SOURCE clause of a CREATE FUNCTION statement

    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. Note that a function cannot be sourced on the COALESCE, NULLIF, or VALUE built-in functions.

This return code can be generated for any type of database object.

Federated system users: the object identified by *name* is not defined in the database or *name* is not a nickname in a DROP NICKNAME statement.

Some 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.

User Response: 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.

Federated system users: if the statement is DROP NICKNAME, make sure the object is actually a nickname. The object might not exist in the federated database or at the data source. Verify the existence of the federated database objects (if any) and the data source objects (if any).

sqlcode: -204

sqlstate: 42704

SQL0205N **Column or attribute** *name* **is not defined in** *object-name***.**

Explanation: If *object-name* is a table or view, then *name* is a column that is not defined in *object-name*. If *object-name* is a structured type, then *name* is an attribute that is not defined in *object-name*.

Federated system users: *object-name* could refer to a nickname.

The statement cannot be processed.

User Response: If *object-name* is a table or view, verify that the column and table or view names (including any required qualifiers) are specified correctly in the SQL statement. If *object-name* is a structured type, verify that the attribute and type 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 valid in the context where it is used.**

Explanation: 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, CREATE METHOD or CREATE FUNCTION statement:
  - The reference *name* does not resolve to the name of a column, local variable or transition variable.
  - The condition name *name* specified in the SIGNAL statement has not been declared.

- 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.

- For a CREATE FUNCTION statement with a PREDICATES clause:
  - The RETURN statement of the SQL function references a variable that is not a parameter or other variable that is in the scope of the RETURN statement.
  - The FILTER USING clause references a variable that is not a parameter name or an expression name in the WHEN clause.
  - The search target in an index exploitation rule does not match some parameter name of the function that is being created.
  - A search argument in an index exploitation rule does not match either an expression name in the EXPRESSION AS clause or a parameter name of the function being created.

- For a CREATE INDEX EXTENSION statement, the RANGE THROUGH clause or the FILTER USING clause references a variable that is not a parameter name that can be used in the clause.

The statement cannot be processed.

User Response: Verify that the 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   A column name is not allowed in the ORDER BY clause of a SELECT statement used with a set operator.

Explanation: 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.

User Response: 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 DB2 Connect.

sqlcode: -207

sqlstate: 42706

SQL0208N   The ORDER BY clause is not valid because column *name* **is not part of the result table.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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 in the ORDER BY clause in the following position, or starting with *expression-start-or-order-by-position* **in the** *clause-type* **clause is not valid. Reason code =** *reason-code***.**

Explanation: The expression identified by the first part of the expression *expression-start-or-order-by-position* 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.

5 The left side of a dereference operator in a GROUP BY clause cannot be a variant function. This reason code occurs only when *clause-type* is GROUP BY.

The statement cannot be processed.

User Response: 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.

5 Remove any variant function from the left side of the dereference operators in the GROUP BY clause.

sqlcode: -214

sqlstate: 42822

SQL0216N   The number of elements on each side of a predicate operator does not match. Predicate operator is *predicate-operator*.

Explanation: A predicate includes a list of elements on the right or left side (or both sides) of the predicate operator. The number of elements must be the same on both sides. These elements may appear in a list of expressions enclosed in parentheses or as elements of a select list in a fullselect.

The statement cannot be processed.

User Response: Correct the predicate with mismatched number of elements on each side of the predicate operator.

sqlcode: -216

sqlstate: 428C4

SQL0217W   The statement was not executed as only Explain information requests are being processed.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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***, column** *name2* **does not have the proper definition or is missing.**

Explanation: 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 (if *name2* is numeric)
- Incorrect data type assigned to columns (if *name2* is a column name).

User Response: 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

SQL0222N **Operation was attempted against a hole using cursor** *cursor-name***.**

Explanation: If the SQLSTATE is 24510, then an error occurred. A positioned update or delete was attempted with cursor *cursor-name*, which is defined as SENSITIVE STATIC, and the current row has been identified as either a delete hole or update hole. A hole occurred when DB2 tried to update or delete a row in the database corresponding to the current row of the result table for cursor *cursor-name* and the corresponding row of the underlying table no longer exists.

If the SQLSTATE is 02502, then this is a warning. A delete hole or update hole has been detected while processing a fetch for cursor *cursor-name*. A hole occurred when DB2 tried to refetch a row from the database corresponding to the current row of the result table for cursor *cursor-name* and the corresponding row of the underlying table no longer exists. No data is returned.

A delete hole occurs when the corresponding row of the underlying table has been deleted.

An update hole occurs when the corresponding row of the underlying table has been updated in the underlying table such that the updated row no longer satisfies the search condition specified in the SELECT statement of the cursor.

The statement cannot be processed. The cursor remains positioned on the hole.

User Response: Issue a FETCH statement to position the cursor on a row that is not a hole.

sqlcode: -222

sqlstate: 02502,24510

SQL0224N **The result table does not agree with the base table using cursor** *cursor-name***.**

Explanation: A positioned UPDATE or DELETE was attempted using cursor *cursor-name*, defined as SENSITIVE STATIC, on a row where the column values in the result table do not match the current values in the base table row. The rows do not match because the row of the base table was updated between the time it was fetched into the result table and the time the positioned UPDATE or DELETE was processed.

The statement cannot be processed. The cursor position is unchanged.

User Response: Change isolation levels so that the base table row cannot be updated again during the cursor operation, or change the application to do a FETCH INSENSITIVE and try

the positioned UPDATE or DELETE again.

sqlcode: -224

sqlstate: 24512

SQL0225N   **FETCH statement for cursor**
cursor-name **is not valid because**
the cursor is not defined as
SCROLL.

Explanation: A FETCH statement for
non-scrollable cursor *cursor-name* been specified
with one of the following scrollable cursor
keywords: PRIOR, FIRST, LAST, BEFORE,
AFTER, CURRENT, ABSOLUTE or RELATIVE.
Only NEXT can be specified for non-scrollable
cursors. No data is fetched.

The statement cannot be processed. The cursor
position is unchanged.

User Response: Change the FETCH statement
to remove the current fetch orientation keyword,
such as PRIOR or FIRST, and replace it with
NEXT. Alternatively, change the definition of the
cursor to be scrollable.

sqlcode: -225

sqlstate: 42872

SQL0227N   **FETCH NEXT, PRIOR, CURRENT**
or RELATIVE is not allowed
because cursor *cursor-name* **has an**
unknown position (*sqlcode***,**
*sqlstate***).**

Explanation: The cursor position for *cursor-name*
is unknown. A previous multiple row FETCH for
cursor *cursor-name* resulted in an error
(SQLCODE *sqlcode*, SQLSTATE *sqlstate*) in the
middle of processing multiple retrieved rows.
One or more of the requested rows could not be
returned to the program following the error,
leaving the position of the cursor unknown.

If an indicator structure had been provided on
the previous multiple row FETCH, a positive
SQLCODE would have been returned and all of
the retrieved rows could have been returned to
the application program.

The statement cannot be processed. The cursor
position is unchanged.

User Response: Close and reopen the cursor to
reset the position. For scrollable cursors, you can
change the FETCH statement to specify one of
the other fetch orientations, such as FIRST, LAST,
BEFORE, AFTER or ABSOLUTE, to establish a
valid cursor position and fetch a row of data.

sqlcode: -227

sqlstate: 24513

SQL0228N   **FOR UPDATE clause specified for**
read-only cursor *cursor-name*.

Explanation: Cursor *cursor-name* is defined as
INSENSITIVE SCROLL, but the corresponding
SELECT statement contains a FOR UPDATE
clause.

The statement cannot be processed.

User Response: To define a read-only cursor,
specify INSENSITIVE on DECLARE CURSOR,
but do not specify the FOR UPDATE clause as
part of the SELECT statement of the cursor.

sqlcode: -228

sqlstate: 42620

SQL0231W   **Current position of cursor**
cursor-name **is not valid for FETCH**
of the current row.

Explanation: A FETCH CURRENT or FETCH
RELATIVE 0 statement was issued for scrollable
cursor *cursor-name*. The operation is not valid,
because the cursor is not positioned on a row of
the result table. A FETCH of the current row is
not allowed following a FETCH BEFORE or
FETCH AFTER statement, or following a FETCH
statement that resulted in SQLCODE +100.

The statement cannot be processed. The cursor
position is unchanged.

User Response: Ensure that the cursor is
positioned on a row of the result table before
attempting to fetch the current row.

sqlcode: +231

sqlstate: 02000

SQL0236W   **SQLDA has only provided** *integer1* **SQLVAR entries.** *integer2* **SQLVAR entries are required for** *integer3* **columns. No SQLVAR entries have been set.**

Explanation: 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)).

User Response: 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.**

Explanation: 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 Base SQLVAR entries (and the SQLDOUBLED flag has been set off (i.e. to the space character).

User Response: 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 SQLDA 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 or structured type,** *integer2* **SQLVAR entries are required for** *integer3* **columns. No SQLVAR entries have been set.**

Explanation: Since at least one of the columns in the result set is a LOB or structured type, 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).

User Response: Increase the value of the SQLN field in the SQLDA to the value indicated in the message (after making sure that the SQLDA 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 or reference type,** *integer2* **SQLVAR entries are required for** *integer3* **columns. No SQLVAR entries have been set.**

Explanation: If any of the columns in the result set is a distinct type or reference 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).

User Response: If the distinct type or reference

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) or reference 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

SQL0242N   **The object named** *object-name* **of type** *object-type* **was specified more than once in the list of objects.**

Explanation: In a list of object names of type *object-type*, the object named *object-name* was specified more than once. The operation of the statement cannot be performed on the object more than once.

User Response: Correct the duplicated object in the list removing duplicate occurrences.

sqlcode: -242

sqlstate: 42713

SQL0243N   **SENSITIVE cursor** *cursor-name* **cannot be defined for the specified SELECT statement.**

Explanation: Cursor *cursor-name* is defined as SENSITIVE, but the content of the SELECT statement requires DB2 to build a temporary result table of the cursor, and DB2 cannot guarantee that changes made outside this cursor will be visible. This situation occurs when the content of the query makes the result table read-only. For example, if the query includes a join, the result table is read-only. In this case, the cursor must be defined as INSENSITIVE or ASENSITIVE.

The statement cannot be processed.

User Response: Either change the content of the

query to yield a result table that is not read-only, or change the type of the cursor to INSENSITIVE or ASENSITIVE.

sqlcode: -243

sqlstate: 36001

SQL0244N   **SENSITIVITY** *sensitivity* **specified on FETCH is not valid for cursor** *cursor-name*.

Explanation: The sensitivity option *sensitivity* specified on FETCH conflicts with the sensitivity option in effect for cursor *cursor-name*. The following list shows what can be specified on FETCH:

```
DECLARE CURSOR
FETCH Statement
INSENSITIVE          INSENSITIVE
SENSITIVE STATIC     SENSITIVE
                     or INSENSITIVE
SENSITIVE DYNAMIC    SENSITIVE
SENSITIVE            SENSITIVE
ASENSITIVE           INSENSITIVE
                     or SENSITIVE
                     (depending on the
                      effective
                      sensitivity of the
                      cursor)
```

In the case of a non-scrollable cursor, a sensitivity option cannot be specified.

The statement cannot be processed.

User Response: Change or remove the sensitivity option specified on FETCH.

sqlcode: -244

sqlstate: 428F4

SQL0257N   **Raw DEVICE containers are currently not supported on this platform.**

Explanation: An attempt was made to use DEVICE containers. These are not currently supported on this platform.

User Response: 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 pending or in progress.**

Explanation: One of the following conditions is true:

1. An ALTER TABLESPACE statement involving container operations has previously been issued by this application or another and has yet to be committed.

2. An ALTER TABLESPACE statement involving container operations has been issued and committed and the table space is currently being rebalanced.

User Response:

1. If possible, roll back the uncommitted unit of work and issue a single ALTER TABLESPACE statement to perform all of the container operations. Otherwise, wait until the rebalance has completed and then try the operation again. Note that pages cannot both be added to and removed from a table space in the same ALTER TABLESPACE statement.

2. 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.**

Explanation: 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.

User Response: 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

SQL0260N **Column** *column-name* **cannot belong to the partitioning key because it is either a LONG column, DATALINK column, or structured type column.**

Explanation: A partitioning key cannot have a LONG-type column, DATALINK column, or structured type column.

The statement cannot be processed.

User Response: Do not use LONG columns, DATALINK columns, or structured type columns for the partitioning key.

If the table only has LONG columns, DATALINK columns, or structured type columns, either add a column that can be used for the partitioning key to the table, or define the table without a partitioning key in a single-node nodegroup.

sqlcode: -260

sqlstate: 42962

SQL0262N **Table** *table-name* **cannot be created in the database partition group** *group-name* **because no column exists that can be used as the partitioning key.**

Explanation: The table *table-name* cannot be created in a database partition group. The table must have at least one column that can be used as the partitioning key. Columns of the following data types cannot be used in a partitioning key:

• LONG VARCHAR, LONG VARGRAPHIC, BLOB, CLOB, DBCLOB

• DATALINK

• Structured type

- A user-defined data type based on any of the above data types

The statement cannot be processed.

User Response: Either create the table with one or more columns that can be used in a partitioning key, or create the table in a single database partition group.

sqlcode: -262

sqlstate: 428A2

SQL0263N **Node range from** *node-number-1* **to** *node-number-2* **is not valid. Second node number must be greater than or equal to first node number.**

Explanation: The specified node range is not valid.

The statement cannot be processed.

User Response: Correct the node range in the statement, then try the request again.

sqlcode: -263

sqlstate: 428A9

SQL0264N **Partitioning key cannot be added or dropped because table resides in a table space defined on the multi-node nodegroup** *name***.**

Explanation: You can only add or drop a partitioning key on a table in a single-node nodegroup.

The statement cannot be processed.

User Response: Do one of the following and try the request again:
- Define an identical table with a partitioning key.
- Redistribute the nodegroup to a single-node nodegroup.

sqlcode: -264

sqlstate: 55037

SQL0265N **Node** *node-number* **is a duplicate node.**

Explanation: For the CREATE NODEGROUP statement, a node can only appear once in the ON NODES clause.

For the CREATE TABLESPACE and ALTER TABLESPACE statement, a node can appear only once and in only one ON NODES clause.

For the ALTER NODEGROUP statement or REDISTRIBUTE NODEGROUP command, one of the following occurred:
- The node appeared more than once in the ADD NODES or the DROP NODES clause.
- The node appeared in both the ADD NODES and the DROP NODES clause.
- The node to be added is already a member of the nodegroup.

The statement cannot be processed.

User Response: Ensure that the node names or node numbers in the ON NODES, ADD NODES, or DROP NODES clause are unique. For the CREATE TABLESPACE and ALTER TABLESPACE statements, ensure that a node appears in no more than one ON NODES clause.

In addition, for the ALTER NODEGROUP statement or REDISTRIBUTE NODEGROUP command:
- Do not specify a node in both the ADD NODES and the DROP NODES clause.
- Remove the node from the ADD NODES clause if the node is already defined in the nodegroup.

sqlcode: -265

sqlstate: 42728

SQL0266N **Node** *node-number* **is not defined.**

Explanation: The node *node-number* is not valid for one of the following reasons:
- the node number is not in the valid range of 0 to 999
- the node is not in the node configuration file

- the node is not part of the nodegroup, therefore the requested operation cannot be processed.

The statement cannot be processed.

User Response: Depending on the condition:

- issue the statement, command or API with node numbers in the valid range.
- follow the procedure to add a node to the system.
- remove the node from the specified nodes in the statement, command or API.

sqlcode: -266

sqlstate: 42729

SQL0268N *operation* **cannot be performed while nodegroup is being redistributed.**

Explanation: One of the following:

- The nodegroup is being redistributed. It cannot be altered, dropped or redistributed again until the current operation is complete.
- The partitioning key of a table cannot be dropped while the nodegroup of the table is being redistributed.

The statement cannot be processed.

User Response: Wait until the redistribution is complete, then try the request again.

sqlcode: -268

sqlstate: 55038

SQL0269N **Database contains maximum number of partitioning maps.**

Explanation: Because the database contains the maximum number of partitioning maps (32,768), you cannot create a new nodegroup, alter a nodegroup, or redistribute an existing one.

The statement cannot be processed.

User Response: Drop one or more nodegroups in the database.

Note: Dropping a nodegroup drops all database objects such as table spaces, tables and views that reside in the nodegroup.

sqlcode: -269

sqlstate: 54033

SQL0270N **Function not supported (Reason code =** *reason-code***).**

Explanation: The statement cannot be processed because it violates a restriction as indicated by the following reason code:

1 The primary key, each unique constraint, and each unique index must contain all partitioning columns of the table (columns may appear in any order).

2 Update of the partitioning key column value is not supported.

3 A foreign key cannot include any nullable partitioning key columns when defined with ON DELETE SET NULL. This is a special case of reason code 2 since defining such a constraint would result in attempting to update a partitioning key column.

4 A table defined using a multipartition nodegroup or a single-partition nodegroup on other than the catalog partition does not support DATA CAPTURE CHANGES.

5 View created with the WITH CHECK OPTION clause should not use functions (or reference views that use functions) that:

- are non-deterministic
- have side effects
- are related to the placement of data (for example nodenumber or partition functions).

These functions must also not be present within referenced views if the new view is created with the CASCADED check option.

6   A transform cannot be defined for a user-defined distinct type.

7   Long fields can only be defined using a table space with a page size that is 4K. A LARGE TABLESPACE can only be created using a 4K page size.

8   Structured types are not supported as columns of a table or structured type attribute data types prior to DB2 Version 7.1.

9   Triggers are not supported on typed tables.

10   A single default table space cannot be selected since the table has one or more LOB columns which must be placed in a table space with a 4K page size and the rowsize or number of columns in the table requires a table space with an 8K page size.

11   A typed table or typed view cannot be created using a structured type that has no attributes.

12   The type of a source key parameter must be a user-defined structured type or a distinct type that is not sourced on LOB, DATALINK, LONG VARCHAR, or LONG VARGRAPHIC.

13   Check constraints cannot be defined on a typed table or the WITH CHECK OPTION clause cannot be specified on a typed view.

14   Referential constraints cannot be defined on a typed table or to a parent table that is a typed table.

15   A default value cannot be defined for reference type columns.

16   A reference data type or structured data type cannot be used as a parameter data type or a returns data type of a user defined function prior to DB2 Version 7.1. Otherwise, a scoped reference data type cannot be used as a parameter data type or returns data type of a routine. A structured type cannot be used as a return column of a table or row function.

17   The SET CONSTRAINTS statement cannot be used for a typed table.

18   Column level UPDATE and REFERENCES privileges cannot be granted on a typed table or typed view.

19   A specific default value must be specified when defining a default for a column of a typed table.

20   ALTER TABLE is not supported for a materialized query table.

21   Column length cannot be altered on a table which is a base table for a materialized query table.

22   Materialized query tables cannot be defined in a CREATE SCHEMA statement.

23   REPLICATED can only be specified for a materialized query table defined with REFRESH DEFERRED.

24   The triggered-action in a BEFORE trigger cannot reference a materialized query table defined with REFRESH IMMEDIATE.

25   Only one materialized query table can be specified for a SET CONSTRAINTS statement.

26   The nodegroup being redistributed contains at least one replicated materialized query table.

27   Replicated materialized query tables cannot be defined on a table that does not have a unique index existing on one or more columns that make up the replicated materialized query table.

28   A typed table or materialized query table cannot be renamed.

29   FOR EXCEPTION clause cannot be specified with a materialized query table in the SET CONSTRAINTS statement.

30 Typed tables and typed views cannot be defined in a CREATE SCHEMA statement.

31 A partitioning key cannot be defined with more than 500 columns.

32 A table defined using a multipartition nodegroup or a single-partition nodegroup on other than the catalog partition does not support DATALINK columns defined with FILE LINK CONTROL.

33 An underlying table of a materialized query table defined with REFRESH IMMEDIATE cannot be the child of a referential constraint with cascading effect (i.e. with option ON DELETE CASCADE or ON DELETE SET NULL).

34 The underlying object relational feature is not supported in the current release.

35 A sequence or an identity column cannot be created in a multi-node database environment.

36 Database activation of a multi-node database is not allowed if sequences or identity columns exist.

38 An index using an index extension is not supported in a multiple partition nodegroup.

39 Nicknames or OLE DB table functions cannot be referenced directly or indirectly in the body of an SQL function or SQL method.

40 The function IDENTITY_VAL_LOCAL cannot be used in a trigger or SQL function.

41 An SQL variable statement assigns to both a local variable and a transition variable. This is not supported.

42 The execution of a trigger, method or function using SQL control statements and the execution of a dynamic compound statement in a multi-node database is not allowed.

43 One or more of the options you have requested is currently not supported.

44 The following EXPLAIN MODES are not supported in MPP, SMP, and Data Joiner:
- COUNT CARDINALITIES
- COMPARE CARDINALITIES
- ESTIMATE CARDINALITIES

45 APPEND mode is not supported for multi-dimensionally clustered (MDC) tables.

46 INPLACE table reorganization is not supported for multi-dimensionally clustered (MDC) tables.

47 Index extensions are not supported for multi-dimensionally clustered (MDC) tables.

48 Changes to the dimension specification of a multi-dimensionally clustered (MDC) table are not supported.

49 Clustering Indexes are not supported for multi-dimensionally clustered (MDC) tables.

50 A user defined temporary table cannot be a multi-dimensionally clustered (MDC) table.

51 CREATE PROCEDURE with LANGUAGE SQL from a non-catalog database partition is not supported.

52 The expression of a generated column cannot be modified if that column was used in the DIMENSIONS clause.

53 A column with a LONG VARCHAR, LONG VARGRAPHIC, DATALINK, LOB type, distinct type on any of these types, or structured type cannot be specified in the select-list of a scrollable cursor.

54 INPLACE table reorganization is not supported for the specified System Catalog table.

55 Federated database system support and concentrator feature cannot be active at the same time.

56 Online index reorganization is not supported for spatial indexes in ALLOW WRITE mode.

57 Online index reorganization is not supported on multi-dimensionally clustered (MDC) tables in ALLOW WRITE mode.

58 The XML data type can only be used as a transient data type and cannot be stored in the database nor returned to an application.

59 A function or method that contains SQL statements cannot be used in a parallel environment.

60 The ALTER TABLE ALTER COLUMN SET INLINE LENGTH statement is not allowed because there is an object of type VIEW, either typed or untyped, that depends on the typed table.

61 Text search function cannot be used in the expression for check constraints or generated columns.

62 The WITH CHECK OPTION clause cannot be used with views which reference text search functions directly or depend on other views referencing text search functions.

63 A column with a LOB type, distinct type on a LOB type, or structured type cannot be specified in the select-list of an insensitive scrollable cursor.

64 Federated processing is not supported on this platform.

65 Altering the nickname local type from the current type to the specified type is not allowed.

User Response: The action corresponding to the reason code is:

1 Correct the CREATE TABLE, ALTER TABLE or CREATE UNIQUE INDEX statement.

2 Do not attempt to update the partitioning key columns for a multipartition table or consider deleting and then inserting the row with the new values in the partitioning columns.

3 Make the partitioning key column not nullable, specify a different ON DELETE action, or change the partitioning key of the table so that the foreign does not include any columns of the partitioning key.

4 Either specify DATA CAPTURE NONE or ensure that the table is placed in a table space on a single-partition nodegroup that specifies the catalog partition.

5 Do not use the WITH CHECK OPTION clause or remove the function or view from the view definition.

6 Transforms are automatic for user-defined distinct types. Use the CREATE TRANSFORM statement for user-defined structured types only.

7 Use a table space with 4K page size for any table that includes long fields. If using DMS table spaces, long fields can be placed in a table space with 4K page size with other table or index data in table spaces with a different page size. When defining a LARGE TABLESPACE, use PAGESIZE 4K.

8 For servers prior to DB2 Version 7.1, ensure that no column data types are structured types in the CREATE TABLE statement or ALTER TYPE ADD COLUMN statement. Ensure that no attribute data types are structured types in the CREATE TYPE statement or ALTER TYPE ADD ATTRIBUTE statement.

9 Do not define triggers on typed tables.

10 Either reduce the row size or number of columns in the table or specify 2 table spaces such that the long data is in a

table space with a 4K page size and the base data is in a table space with an 8K page size.

11   When creating a typed table or typed view, specify a structured type that has at least one attribute defined.

12   For the type of a source key parameter, use only a user-defined structured type or a distinct type that is not sourced on LOB, DATALINK, LONG VARCHAR, or LONG VARGRAPHIC.

13   In a CREATE TABLE or ALTER TABLE statement for a typed table, do not specify check constraints. In a CREATE VIEW statement of a typed view, do not specify the WITH CHECK OPTION clause.

14   Do not specify referential constraints involving typed tables in a CREATE TABLE or ALTER TABLE statement.

15   Do not specify a DEFAULT clause for a column with a reference data type in a CREATE TABLE or ALTER TABLE statement.

16   For servers prior to DB2 Version 7.1, do not specify a structured type parameter or returns type when creating a user defined function. Otherwise, do not specify a scoped reference type as a parameter or returns type. Do not specify a structured type as a return column of a table or row function.

17   Do not specify a typed table in the SET CONSTRAINTS statement.

18   Do not include specific column names when granting REFERENCES or UPDATE privileges on a typed table or typed view.

19   Include a specific value when specifying the DEFAULT clause on a column of a typed table.

20   Drop the materialized query table and recreate it with the desired attributes.

21   Drop the materialized query table(s),

alter the column length of the base table and then recreate the materialized query table(s).

22   Issue the CREATE SUMMARY TABLE statement outside of the CREATE SCHEMA statement.

23   Either remove the REPLICATED specification or ensure that REFRESH DEFERRED is specified for the materialized query table definition.

24   Remove the reference to the materialized query table in the triggered-action in the BEFORE trigger.

25   Issue separate SET CONSTRAINTS IMMEDIATE CHECKED statements for each materialized query table.

26   Drop all replicated materialized query tables in the nodegroup and then issue the REDISTRIBUTE NODEGROUP command again. Recreate the replicated materialized query table(s).

27   Ensure that a subset of the columns defined for the materialized query table also are the set of columns that make up a unique index on the base table.

28   A typed table or materialized query table name can only be changed by dropping the table and creating it again with the new name. Dropping the table may have implications on other objects that depend on the table and the privileges on the table are lost.

29   Remove the FOR EXCEPTION clause from the SET CONSTRAINTS statement.

30   Issue the CREATE statement for the typed view or typed table outside of the CREATE SCHEMA statement.

31   Reduce the number of columns in the partitioning key.

32   Either specify NO LINK CONTROL for the DATALINK column or ensure that the table is placed in a table space on a single-partition nodegroup that specifies the catalog partition. If redistributing to

a multiple-partition nodegroup, the table needs to be dropped to continue with the redistribute.

33
- Do not define a referential constraint with cascading effect (i.e. with option ON DELETE CASCADE or ON DELETE SET NULL) with an underlying table of a materialized query table defined with REFRESH IMMEDIATE as the child, or
- do not define a REFRESH IMMEDIATE materialized query table whose underlying table is the child of a referential constraint with cascading effect (i.e. with option ON DELETE CASCADE or ON DELETE SET NULL).

34 The error can be corrected by removing the use of any unsupported object relational features.

35 Remove the "GENERATED [ALWAYS | BY DEFAULT] AS IDENTITY ..." attribute from the column or do not attempt to create the sequence.

36 Drop the new node or nodes to go back to a single node configuration. If more nodes are required, then the sequences or tables with identity columns must be dropped before new nodes are added.

38 An index using an index extension cannot be created on a table in a multiple partition nodegroup. A nodegroup cannot become a multiple partition nodegroup while an index using an index extension exists on a table in the nodegroup. Either drop any such indexes and add the partition to the nodegroup, in which case the indexes cannot be re-created, or leave the nodegroup unchanged.

39 Remove the reference to a nickname or OLE DB table function or remove the reference to the object that indirectly references one of these.

40 Remove the invocation of the

IDENTITY_VAL_LOCAL function from the trigger definition or the SQL function definition.

41 Split the assignment into two separate statements. One statement must only assign values to SQL variables, while the other statement must assign values only to transition variables.

42 Drop the new node or nodes to return to a single node configuration. If more nodes are required, then the triggers, functions or methods containing control statements must be dropped.

43 Please reissue the runstats command and set the unsupported option off.

44 These EXPLAIN modes cannot be used in SMP, MPP, and Data Joiner. If possible, try to run the query in serial mode. Otherwise, setting the EXPLAIN mode to YES or EXPLAIN will provide the same information except for the actual cardinalities.

45 Do not specify the APPEND clause in an ALTER TABLE statement for multi-dimensionally clustered (MDC) tables.

46 Reissue the REORG command without specifying the INPLACE option.

47 Do not specify the EXTENSION clause in a CREATE INDEX statement for multi-dimensionally clustered (MDC) tables.

48 Drop the multi-dimensionally clustered (MDC) table and recreate it with the modified dimension specification.

49 Do not specify the CLUSTER clause in a CREATE INDEX statement for multi-dimensionally clustered (MDC) tables.

50 Do not specify the DIMENSION clause in a CREATE TABLE statement for declared global temporary tables.

51   Issue the CREATE PROCEDURE with LANGUAGE SQL from the catalog database partition.

52   Do not modify the expression. To change the composition of the DIMENSIONS clause or the generating expression of any of its members that are generated columns, the table must be dropped and created again, then repopulated.

53   Modify the select-list of the scrollable cursor to not include a column with these types.

54   Reissue the REORG command without specifying the INPLACE option.

55   Either turn off concentrator or federated database system support. Turn off concentrator by setting the database manager parameter MAX_CONNECTIONS to be less than or equal to MAX_COORDAGENTS. Turn off federated database system support by setting the FEDERATED parameter in the database manager configuration to NO.

56   Reissue the REORG INDEXES command and specify ALLOW NONE or ALLOW READ.

57   Reissue the REORG INDEXES command and specify ALLOW NONE or ALLOW READ.

58   Input the XML data to one of the functions that accept XML input and store the output of the function in the database or return it to the application.

59   In a parallel environment, only use functions and methods defined with the NO SQL option.

60   Drop the views that depend on the typed table. Reissue the ALTER TABLE ALTER COLUMN SET INLINE LENGTH statement on the typed table and recreate the views that were dropped.

61   Do not use the text search function in the expression for check constraints or generated columns. Use the LIKE function instead of CONTAINS, if possible.

62   Do no specify the WITH CHECK OPTION clause on this view.

63   Modify the select-list of the scrollable cursor to not include a column with these types.

64   Do not attempt federated processing on this platform.

65   Do not attempt to alter the nickname local type from the current type to the specified type.

sqlcode: -270

sqlstate: 42997

SQL0271N   **Index file for table with fid** *fid* **is either missing, or is not valid.**

Explanation: The index file for the table with fid *fid* is required during processing. The file is either missing, or it is not valid.

The statement cannot be processed, and the application is still connected to the database. This condition does not affect other statements that do not use the index on this table.

User Response: Ensure that all users are disconnected from the database, then issue the RESTART DATABASE command on all nodes. Then try the request again.

The index (or indexes) is re-created when the database is restarted.

sqlcode: -271

sqlstate: 58004

SQL0276N   **Connection to database** *name* **cannot be made because it is in the restore pending state.**

Explanation: The database must be restored before a connection can be done.

No connection was made.

User Response: Restore the database, then issue the CONNECT statement again.

sqlcode: -276

sqlstate: 08004

SQL0279N **The database connection has been terminated during COMMIT processing. The transaction could be in doubt. Reason code =** *reason-code***.**

Explanation: Commit processing encountered an error. The transaction has been put in the commit state, but the commit processing may not have completed. The application's database connection has been terminated.

The cause of the error is indicated by the *reason-code*:

1 A node involved in the transaction has failed.

2 Commit was rejected on one for the nodes. Check the administration notification log for details.

User Response: Determine the cause of the error. It may be necessary to contact the system administrator for assistance since the most common cause of the error is node failure or connection failure. The RESTART DATABASE command will complete the commit processing for this transaction.

sqlcode: -279

sqlstate: 08007

SQL0280W **View, trigger or materialized query table** *name* **has replaced an existing inoperative view, trigger or materialized query table.**

Explanation: An existing inoperative view, trigger or materialized query table *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

• the new materialized query table definition as a result of a CREATE SUMMARY TABLE statement.

User Response: 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.**

Explanation: Additional containers cannot be added to a system managed table space. The exception to this is when a nodegroup was modified to add a node without table spaces, then containers can be added once on the new node using the ALTER TABLESPACE command. In general, the table space must be managed by the database in order to add additional containers.

The statement cannot be processed.

User Response: To add more containers to a system managed table space, drop and recreate the table space with more containers, ensuring that each container is of the same size and less than the container size limit, or change to a DMS table space.

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.**

Explanation: A table in the specified table space does not contain all of its parts in that table space. If more than one table spaces are specified, then a table in one of the specified table spaces does not contain all of its parts in the list. The base table, indexes, or long data may be in another table space, so dropping the table

space(s) will not completely drop the table. This would leave the table in an inconsistent state and therefore the table space(s) cannot be dropped.

User Response: Ensure that all objects contained in table space *tablespace-name* contain all their parts in this table space before attempting to drop it, or include those table spaces containing the parts in the list to be dropped.

This may require dropping the table *table-name* before dropping the table space.

sqlcode: -282

sqlstate: 55024

SQL0283N **System temporary table space** *tablespace-name* **cannot be dropped because it is the only system temporary table space with a** *page-size* **page size in the database.**

Explanation: A database must contain at least one system temporary table space with the same page size as the page size of the catalog tablespace. Dropping table space *tablespace-name* would remove the last system temporary tablespace with a *page-size* page size from the database.

User Response: Ensure there will be another system temporary table space with a *page-size* page size in the database before attempting to drop this table space.

sqlcode: -283

sqlstate: 55026

SQL0284N **The table was not created because the table space** *tablespace-name* **following the clause** *clause* **is a** *tablespace-type* **table space.**

Explanation: The CREATE TABLE or DECLARE GLOBAL TEMPORARY 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:

- for regular tables, *tablespace-name* was specified on the IN clause and the table space is not a REGULAR table space.
- for declared temporary tables, *tablespace-name* was specified on the IN clause and the table space is not a USER TEMPORARY table space.
- *tablespace-name* was specified on the LONG IN clause and the table space is not a LARGE, managed by database table space.
- *tablespace-name* was specified on the INDEX IN clause and the table space is not a REGULAR or LARGE, managed by database, table space.

User Response: 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.**

Explanation: 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.

User Response: 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 **A default table space could not be found with a pagesize of at least** *pagesize* **that authorization ID** *user-name* **is authorized to use.**

Explanation: The CREATE TABLE or DECLARE GLOBAL TEMPORARY TABLE statement did not specify a table space, and a table space of the

right type (USER TEMPORARY for declared temporary table) with sufficient page size (at least *pagesize*), over which authorization ID *user-name* has USE privilege, could not be found.

Sufficient page size for a table is determined by either the byte count of the row or the number of columns.

User Response: Ensure the existence of a table space of the correct type (REGULAR or USER TEMPORARY) with a page size of at least *pagesize* and that authorization ID *user-name* has USE privilege on this table space.

sqlcode: -286

sqlstate: 42727

SQL0287N   SYSCATSPACE cannot be used for user objects.

Explanation: The CREATE TABLE or GRANT USE OF TABLESPACE statement specified a table space named SYSCATSPACE which is reserved for catalog tables.

User Response: Specify a different table space name.

sqlcode: -287

sqlstate: 42838

SQL0288N   A large table space cannot be defined using MANAGED BY SYSTEM.

Explanation: 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 large table space cannot be defined to use system managed space.

User Response: Remove the keyword LARGE or change to MANAGED BY DATABASE in the CREATE TABLESPACE statement.

sqlcode: -288

sqlstate: 42613

SQL0289N   Unable to allocate new pages in table space *tablespace-name*.

Explanation: One of the following conditions is true:

1. One of the containers assigned to this SMS table space has reached the maximum file size. This is the likely cause of the error.
2. All the containers assigned to this DMS table space are full. This is the likely cause of the error.
3. The table space object table for this DMS table space is full.
4. A rebalance is in progress, but has not progressed far enough to enable the newly added space to be used.
5. A redirected restore is being done to containers that are too small.
6. A rollforward is being done following a redirected restore and all the containers assigned to this tablespace are full.
7. A rollforward skipping add containers is being done and all the containers assigned to this tablespace are full.
8. An attempt was made to create a table space with less than 5 usable extents.

User Response: Perform the action corresponding to the cause of the error:

1. either switch to a DMS TABLESPACE or recreate the SMS TABLESPACE with more directories (PATHs) such that: (number of directories) >= (max tablesize / maxfilesize). Note that maximum file size is operating system dependent.
2. add new container(s) to the DMS table space and try the operation again, after the rebalancer has made the new pages available for use.
3. drop unnecessary tables from this DMS table space.
4. wait for the rebalancer to make more progress.
5. perform the redirected restore again to larger containers.
6. perform the redirected restore again to larger containers.

7. perform the rollforward again allowing add containers, or perform a redirected restore to larger containers.

8. resubmit the CREATE TABLESPACE statement, ensuring that the table space has at least 5 usable extents.

sqlcode: -289

sqlstate: 57011

SQL0290N   Table space access is not allowed.

Explanation: 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 system or user temporary table space cannot be dropped which contains active system or declared temporary tables.

• The SET CONTAINER api cannot be used to set the container list unless the table space is in a "restore pending" state.

User Response: 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.

Refer to the Administration Guide for further information about the table space states.

sqlcode: -290

sqlstate: 55039

SQL0291N   State transition not allowed on table space.

Explanation: 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.

User Response: 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.

Explanation: An internal database file could not be created.

User Response: 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.

Explanation: This error may be caused by one of the following conditions:

• 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.

This error can be returned during database startup and during the processing of the ALTER TABLESPACE SQL statement.

User Response: Try the following actions:

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.

If the error was returned from the processing of an ALTER TABLESPACE SQL statement with the SWITCH ONLINE option, then re-issue the statement after correcting the problem as described above.

If the error persists, call your IBM service representative.

sqlcode: -293

sqlstate: 57048

SQL0294N   The container is already in use.

Explanation: Table space containers may not be shared. The possible causes of this error include the following.

- A CREATE TABLESPACE or ALTER TABLESPACE statement included a container that is already in use by another table space.

- A CREATE TABLESPACE or ALTER TABLESPACE statement included a container from a table space that has been dropped but the drop statement has not been committed.

- An ALTER NODEGROUP statement used to add a node used the containers of a LIKE node that is on the same physical node. These containers would therefore already be in use.

- A CREATE TABLESPACE or ALTER TABLESPACE statement is attempting to use the same container on more than one logical node on a single physical node. The same containers cannot be used for more than one node on the same physical node.

- An ADD NODE command or API used the containers from the system temporary table

spaces of a LIKE node that is on the same physical node. These containers would therefore already be in use.

- A CREATE TABLESPACE or ALTER TABLESPACE statement included a DMS container from another database that no longer exists but was not dropped properly. The container is not actually in use but it is tagged as being in use. Therefore, DB2 will not allow it to be used until it is untagged. However, it is very important to verify that the container is not in use by the same database or another database when untagging it. If the container is in use when you untag it, the database(s) involved will be corrupted.

User Response: Ensure the containers are unique.

- For the CREATE or ALTER TABLESPACE statement, specify a different container for the table space.

- For the CREATE or ALTER TABLESPACE statement including a container from a dropped table space, try again after the drop statement is committed or specify a different container.

- For the ALTER NODEGROUP statement, reissue the statement using the WITHOUT TABLESPACES clause, and then use the ALTER TABLESPACE statement to create unique containers for the new node.

- For CREATE or ALTER TABLESPACE statements where the environment includes more than one logical node on a physical node, ensure that the same containers are not specified for such logical nodes.

- For an ADD NODE command or API, reissue the statement using the WITHOUT TABLESPACES clause, and then use the ALTER TABLESPACE statement to create unique containers at the new node for the system temporary table spaces.

- If you are trying to use a DMS container that belonged to a database that no longer exists but was not dropped properly, then the db2untag utility can be used to remove the DB2 container tag from it. When this tag is removed DB2 considers the container to be free and the container can be used in a

CREATE TABLESPACE or ALTER TABLESPACE statement.

NOTE: Use extreme caution with db2untag. If you issue a db2untag command against a container that is still in use by a database, then both the database that originally used the container and the database that is now using the container will be corrupted.

sqlcode: -294

sqlstate: 42730

SQL0295N   The combined length for all container names for the table space is too long.

Explanation: The total space required to store the list of containers exceeds the space allotted for this table space in the table space file.

User Response: 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.

Explanation: This database contains the maximum number of table spaces. No more can be created.

User Response: 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.

Explanation: 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 administration notification log.

User Response: Shorten the path length.

sqlcode: -297

sqlstate: 54036

SQL0298N   Bad container path.

Explanation: 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.

- For EXTEND, REDUCE, RESIZE and DROP operations, the specified container path must exist.

- 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.

- Containers (files) for dropped database managed table spaces can only be reused as

containers (directories) for system managed table spaces after all agents terminate and vice versa.

- During a redirected restore, an SMS container was specified for a DMS table space or a DMS container was specified for an SMS table space.

- The specified type of the container for an EXTEND, REDUCE, RESIZE, or DROP operation does not match the type of the container (FILE or DEVICE) that was specified when the container was created.

This message will also be returned if any other unexpected error occurred which prevents DB2 from accessing the container.

User Response: 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.

Explanation: The container that you are attempting to add has already been assigned to the table space.

User Response: Choose another container and try again.

sqlcode: -299

sqlstate: 42731

SQL0300 - SQL0399

SQL0301N The value of input host variable or parameter number *number* **cannot be used because of its data type.**

Explanation: A host variable or parameter in position *number* 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. In the case of a user-defined structured type, the associated built-in type of the host variable or SQLTYPE might not be compatible with the parameter of the TO SQL transform function defined in the transform group for the statement. In the case of performing an implicit or explicit cast between character and graphic data types, this error indicates that such a cast was attempted with a non-Unicode character or graphic string.

The statement cannot be processed.

User Response: 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.

Explanation: 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.

- Federated system 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.

User Response: 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.

Federated system 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.**

Explanation: An imbedded SELECT or VALUES statement selects into a host variable, 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 a user-defined data type, the host variable might be defined with an associated built-in data type that is not compatible with the result type of the FROM SQL transform function defined in the transform group for the statement. 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.

User Response: Verify that the table definitions are current and that the host variable has the correct data type. For a user-defined data type,

verify that the associated built-in type of the host variable is compatible with the result type of the FROM SQL transform function defined in the transform group for the statement.

sqlcode: -303

sqlstate: 42806

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.**

Explanation: 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.

User Response: 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*.

Federated system users: for the ranges of data types that are returned from a data source, refer to the documentation for that data source.

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.**

Explanation: 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.

User Response: 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.**

Explanation: The host variable *name* is not declared in any DECLARE SECTION.

The statement cannot be processed.

User Response: Ensure that the host variable is declared and the name spelled correctly.

SQL0307N   **The host variable** *name* **is already defined.**

Explanation: The host variable *name* has already been declared in a DECLARE SECTION.

The definition is ignored. The previous definition is used.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The maximum number of host variables was exceeded in the statement.

The statement cannot be processed.

User Response: Ensure the statement has fewer host variables or is less complex.

SQL0311N   **The length of string host variable number** *var-number* **is negative or greater than the maximum.**

Explanation: When evaluated, the length specification for the string host variable, whose entry in the SQLDA is indicated by <var-number> (based at 1), was negative or greater than the maximum defined for that host variable.

The statement cannot be processed.

User Response: Correct the program to ensure that the lengths of all string host variables are not negative or that they are not greater than the maximum allowed length.

sqlcode: -311

sqlstate: 22501

SQL0312N   **The host variable** *host-name* **is used in a dynamic SQL statement, a view definition, or a trigger definition.**

Explanation: 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.

User Response: 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 values required.**

Explanation: The number of host variables specified in the EXECUTE or OPEN statement does not equal the number of host variables required for the parameter markers (?) appearing in the SQL statement.

User Response: Correct the application program so the number of host variables specified in the EXECUTE or OPEN statement is correct for the parameter markers in the SQL statement.

sqlcode: -313

sqlstate: 07001, 07004

SQL0314N **The host variable** *name* **is incorrectly declared.**

Explanation: 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.

User Response: Ensure that you correctly specify only the declarations the database manager supports.

SQL0315N **The host variable is incorrectly declared.**

Explanation: 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.

User Response: Ensure that you correctly specify only the declarations the database manager supports.

SQL0317N **No END DECLARE SECTION was found after a BEGIN DECLARE SECTION.**

Explanation: The end of input was reached during processing of a DECLARE SECTION.

Precompilation is terminated.

User Response: Add an END DECLARE SECTION statement to end a DECLARE SECTION.

SQL0318N **An END DECLARE SECTION was found without a previous BEGIN DECLARE SECTION.**

Explanation: An END DECLARE SECTION statement was found, but there was no previous BEGIN DECLARE SECTION.

The statement cannot be processed.

User Response: Enter a BEGIN DECLARE SECTION before an END DECLARE SECTION.

SQL0324N **The** *usage* **variable** *name* **is the wrong type.**

Explanation: 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.

User Response: 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*.

Explanation: 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.

- Federated system users: the data source does not support the specified code page conversion.

- DB2 Connect users: both "source code page" and "target code page" may refer to a CCSID on the host or AS/400 system.

- An error has occurred during conversion from the code page stored with the encrypted data to the target code page.

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.

User Response: 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 *Quick Beginnings*. If

DB2 Connect is being used, check the *DB2 Connect Quick Beginnings*.

- 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 *Quick Beginnings* or the *DB2 Connect Quick Beginnings*.

- 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 *Quick Beginnings* or the *DB2 Connect Quick Beginnings*.

- Change the unsupported code page(s) to one of the documented pairs.

AS/400 users should be aware that AS/400 CCSID 65535 is not supported. AS/400 data encoded using CCSID 65535 must be converted to a supported CCSID before it can be accessed using DB2 Connect.

Federated system users: for data source code page support, see the *Federated Systems 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*.

Explanation: 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.

User Response: 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,
- casting the encrypted data value to a VARCHAR string with a larger number of bytes before using it in a decryption function, 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

SQL0336N **The scale of the decimal number must be zero.**

Explanation: The decimal number is used in a context where the scale must be zero. This can occur when a decimal number is specified in a CREATE or ALTER SEQUENCE statement for START WITH, INCREMENT, MINVALUE, MAXVALUE or RESTART WITH.

The statement cannot be processed.

User Response: Change the decimal number to remove any non-zero digits from the right side of the decimal delimiter.

sqlcode: -336

sqlstate: 428FA

SQL0338N **An ON clause associated with a JOIN operator is not valid.**

Explanation: An ON clause associated with a JOIN operator is not valid for one of the following reasons.

- The join condition cannot include any subqueries.
- Column references in an ON clause must only reference columns of tables that are in the scope of the ON clause (included in the same joined-table clause as the ON clause).
- Scalar fullselects are not allowed in the expressions of an ON clause.
- A function referenced in an ON clause of a full outer join must be deterministic and have no external action.
- A dereference operation (->) cannot be used.
- A SQL function or SQL method cannot be used.

The statement cannot be processed.

User Response: Correct the ON clause to reference appropriate columns or delete any subqueries or scalar fullselects. Remove any dereference operations, SQL functions, or SQL methods from the ON clause.

If using full outer join ensure that all functions in

the ON clause are deterministic and have no external action.

sqlcode: -338

sqlstate: 42972

SQL0340N **The common table expression** *name* **has the same identifier as another occurrence of a common table expression definition within the same statement.**

Explanation: 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.

User Response: 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***.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

sqlcode: -342

sqlstate: 42925

SQL0343N **The column names are required for the recursive common table expression** *name***.**

Explanation: 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.

User Response: 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***.**

Explanation: 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.

User Response: 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, HAVING clause, ORDER BY clause, or an explicit join including an ON clause.**

Explanation: 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
* cannot include a HAVING clause
* cannot include an ORDER BY clause on the iterative fullselect
* and cannot include an explicit join with an ON clause.

The statement cannot be processed.

User Response: Change the common table expression by:
* making it a union of two or more fullselects
* removing any column functions, GROUP BY clause, HAVING clause, ORDER BY clause, or explicit JOIN including an ON 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: To prevent an infinite loop, include the expected syntax as described.

sqlcode: +347

sqlstate: 01605

SQL0348N *sequence-expression* **cannot be specified in this context.**

Explanation: The statement contains a NEXTVAL expression or PREVVAL expression in an invalid context. NEXTVAL expressions and PREVVAL expressions cannot be specified in the following contexts:

- join condition of a full outer join
- DEFAULT value for a column in a CREATE TABLE or ALTER TABLE statement
- generated column definition in a CREATE TABLE or ALTER TABLE statement
- condition of a CHECK constraint
- CREATE TRIGGER statement (a NEXTVAL expression may be specified, but a PREVVAL expression cannot be specified)
- CREATE VIEW statement, CREATE METHOD statement or CREATE FUNCTION statement

NEXTVAL expressions cannot be specified in the following contexts:

- CASE expression
- parameter list of an aggregate function

- subquery except in the fullselect of an INSERT, UPDATE or VALUES INTO statement
- SELECT statement for which the outer SELECT contains a DISTINCT operator
- SELECT statement for which the outer SELECT contains a GROUP BY clause
- join condition of a join
- SELECT statement for which the outer SELECT is combined with another SELECT statement using the UNION, INTERSECT or EXCEPT set operator
- nested table expression
- parameter list of a table function
- WHERE clause of the outer-most SELECT statement, DELETE or UPDATE statement
- ORDER BY clause of the outer-most SELECT statement
- select-cause of the fullselect of an expression, in the SET clause of an UPDATE statement
- IF, WHILE, DO...UNTIL, or CASE statement in an SQL routine

The statement cannot be executed.

User Response: Remove the reference to the sequence expression and resubmit the statement.

sqlcode: -348

sqlstate: 428F9

SQL0349N **The specification of the NEXTVAL expression for the column in position** *column-position* **must match the specification of all other expressions for the same column for all rows.**

Explanation: The expression specified for the column in position *column-position* of a VALUES clause of a multiple row INSERT statement or VALUES expression included a NEXTVAL expression. When an expression containing a NEXTVAL expression is used to specify the value of a column in one of these contexts, then that same expression must be specified for that column for all rows. For example, the following INSERT statement succeeds:

```
INSERT INTO T1
  VALUES(
    NEXTVAL FOR sequence1 + 5, 'a'
    ),
    (
    NEXTVAL FOR sequence1 + 5, 'b'
    ),
    (
    NEXTVAL FOR sequence1 + 5, 'c'
    )
```

However, the following INSERT statement fails:

```
INSERT INTO T1
  VALUES(
    NEXTVAL FOR sequence1 + 5, 'a'
    ),
    (
    NEXTVAL FOR sequence1 + 5, 'b'
    ),
    (
    NEXTVAL FOR sequence1 + 4, 'c'
    )
```

User Response: Correct the syntax and resubmit the statement.

sqlcode: -349

sqlstate: 560B7

SQL0350N   LOB, DATALINK, or structured type column *column-name* **cannot be used in an index, a key, a unique constraint, a generated column, or a declared temporary table.**

Explanation: Even in cases where it would not violate the maximum size of an index, key, or unique constraint, a LOB column, DATALINK column, or structured type column cannot be used in an index, key or unique constraint. These data types are also not supported as column types of a generated column or a declared temporary table. This restriction includes the use of distinct type column based on a LOB or DATALINK.

The statement cannot be processed.

User Response: Remove the LOB, DATALINK or structured type column from the specification of the index, key, unique constraint, generated column, or declared temporary table. An index

defined on a structured type column may be defined using an index extension.

sqlcode: -350

sqlstate: 42962

SQL0351N   An unsupported SQLTYPE was encountered in position *position-number* **of the output SQLDA (select list).**

Explanation: The element of the SQLDA at position *position-number* is for a data type that either the application requestor or the application server does not support. If the application is not using the SQLDA directly, *position-number* could represent the position of an element in the select list or a parameter of a CALL statement.

The statement cannot be processed.

User Response: Change the statement to exclude the unsupported data type. For a select statement, remove the names of any columns in the select-list with the unsupported data type or use a cast in the query to cast the column to a supported data type.

sqlcode: -351

sqlstate: 56084

SQL0352N   An unsupported SQLTYPE was encountered in position *position-number* **of the input list (SQLDA).**

Explanation: The element of the SQLDA at position *position-number* is for a data type that either the application requestor or the application server does not support. If the application is not using the SQLDA directly, *position-number* could represent the position of an input host variable, parameter marker, or a parameter of a CALL statement.

The statement cannot be processed.

User Response: Change the statement to exclude the unsupported data type.

sqlcode: -352

sqlstate: 56084

SQL0355N **The column** *column-name*, **as defined, is too large to be logged.**

Explanation: 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.

User Response: 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

sqlstate: 42993

SQL0357N **The DB2 Data Links Manager** *name* **is not currently available. Reason code =** *reason-code*.

Explanation:

The statement requires processing on the DB2 Data Links Manager *name*. The DB2 Data Links Manager is not currently available as indicated by the reason code.

01   The DB2 Data Links Manager is not available.

02   The database server, instance, or database from which the operation was attempted is not registered with the appropriate DB2 Data Links Manager.

03   Either access to the DB2 Data Links Manager is currently disallowed, or DB2 is unable to connect to the DB2 Data Links Manager.

04   The DB2 Data Links Manager is an unknown server.

05   An error occurred while communicating with the DB2 Data Links Manager.

06   The installation type of the DB2 Data Links Manager is not compatible with that registered to the database.

The statement cannot be processed.

User Response: The action depends on the reason code as follows.

01   The DB2 Data Links Manager or a communication link may be down. Wait a while and try again, or check with the DB2 Data Links Manager administrator. If the problem persists, disconnect the application from the database, and retry after connecting again.

02   Register the database server, instance, or database with the DB2 Data Links Manager.

03   Access to the DB2 Data Links Manager is disallowed until DB2 can ensure that it is in a consistent state. Wait until DB2 does this asynchronously. It is possible that DB2 is unable to do this because of conditions in other reason codes. So, if the problem persists, check the db2diag.log file of the instance to find out the condition and correct that condition. For more information, refer to the Administration Guide for information on crash recovery for DB2 Data Links Managers. Also check all the host settings for connections to this DB2 Data Links Manager.

04   Ensure that the DB2 Data Links Manager is available on the network.

05   Check that the DB2 Data Links Manager and the communication link are up. If the problem persists, disconnect the application from the database, and retry after connecting again.

06   If the DB2 Data Links Manager was installed for DFS, then it must added as a CELL to the database. If it was installed for native file systems, then it must be added as a NODE. Refer to the Command Reference for details on ADD DATALINKS MANAGER command.

sqlcode: -357

sqlstate: 57050

SQL0358N **Unable to access file referenced by a DATALINK value. Reason code =** *reason-code*.

Explanation: A DATALINK value could not be assigned. The possible reason codes are as follows:

21 DATALINK value data location format is not valid.

22 DATALINK value DB2 Data Links manager is not registered with the database.

23 DATALINK linktype value is not valid.

24 DATALINK value referenced file cannot be found by a DB2 Data Links Manager.

25 DATALINK value referenced file is already linked to a database.

26 DATALINK value referenced file cannot be accessed for linking. It may be a directory, a symbolic link or a file with permission bit for set user ID (SUID) on or set group ID (SGID) on, or a file located in a file system not defined as a DLFS (Data Links File System).

27 DATALINK value data location or comment is too long.

28 Existing registry on the DB2 Data Links Manager does not allow this file to be linked (No matching prefix on DLFM).

29 The DB2 Data Links Manager does not authorize the DB2 user to link this file.

30 A link is already in progress for the file by another application.

31 File copy by the DB2 Data Links Manager is not complete for the file to be unlinked. The statement cannot be processed.

32 DATALINK value does not contain a valid write token.

33 DATALINK value constructed by the DLPREVIOUSCOPY scalar function can be assigned only to a DATALINK column defined with WRITE PERMISSION ADMIN and RECOVERY YES.

34 DATALINK value constructed by the DLNEWCOPY or DLPREVIOUSCOPY scalar function does not match the value as already exists in the column.

35 DATALINK value constructed by the DLNEWCOPY or DLPREVIOUSCOPY scalar function cannot be used in an INSERT statement to assign a new value.

36 The write token embedded in DATALINK value data location does not match the token used to open the DATALINK value referenced file.

37 DATALINK value reference file is in the update-in-progress state.

38 DATALINK value with scheme DFS cannot be assigned to a DATALINK column defined with WRITE PERMISSION ADMIN.

39 DATALINK value constructed by the DLNEWCOPY scalar function cannot be assigned to a DATALINK column defined with WRITE PERMISSION BLOCKED.

40 The previous archive copy of DATALINK reference file cannot be located from the archive server.

41 The same DATALINK value constructed by the DLNEWCOPY or DLPREVIOUSCOPY scalar function cannot be assigned multiple times within the same transaction or the same datalink value cannot be unlinked while the assignment of the same datalink value constructed by DLNEWCOPY or DLPREVIOUSCOPY is not yet committed.

42 DATALINK value constructed by the DLREPLACECONTENT scalar function can be assigned to a DATALINK column defined with NO LINK CONTROL only

if the second argument (replacement file location) is a zero-length string or a null.

43 The unlink operation of the replacement file specified in the DLREPLACECONTENT scalar function has not committed.

44 The replacement file specified in the DLREPLACECONTENT scalar function is already used in another replacement process.

45 DATALINK referenced file is already specified as the replacement file in another operation.

46 The format of the replacement file specified in the DLREPLACECONTENT scalar function is not valid.

47 The format of the replacement file specified in the DLREPLACECONTENT scalar function cannot be a directory or a symbolic link.

48 The replacement file specified in the DLREPLACECONTENT scalar function is already linked to a database.

49 The replacement file specified in the DLREPLACECONTENT scalar function cannot be found by a Data Links File Manager.

50 DATALINK values that have a write token in the data location, as constructed by the DLNEWCOPY scalar function, can only be assigned to DATALINK columns defined with WRITE PERMISSION ADMIN.

User Response: The action is based on the reason code as follows.

21 Correct the data location format.

22 Ensure that the correct DB2 Data Links Manager is specified and if so have it registered to the database. Registered DB2 Data Links Managers are ignored unless the database manager configuration parameter Datalinks is set to YES.

23 Correct the linktype value.

24 Check that the correct file is specified and that the file exists.

25 Either unlink the existing reference to the file or do not specify the file in this statement.

26 Linking of directories is not allowed. Use the actual file name, not a symbolic link. If SUID or SGID is on, or if the file is located in a non-DLFS file system, this file cannot be linked using a DATALINK type.

27 Reduce the length of the data location value or the comment.

28 Contact the DB2 Data Links Manger administrator to add the registration necessary to link this file.

29 Contact the DB2 Data Links Manager administrator to get the necessary authorization.

30 Do not link this file or retry later.

31 Allow some time for the file copy to complete and retry later.

32 Provide the data location value with a valid write token. A write token is generated as part of the returned value from either the DLURLCOMPLETEWRITE or DLURLPATHWRITE scalar function in a SELECT statement on a DATALINK column defined with WRITE PERMISSION ADMIN.

33 Use DLVALUE scalar function instead.

34 Correct the data location to match the one as exists in the row.

35 If assigning a new DATALINK value, use DLVALUE scalar function instead.

36 Provide the data location embedded with the write token which is used to open and modify the file. If the token is lost, there is a set of Data Links File manager (DLFM) commands available to retrieve the write token. Refer to the

Data Links Manager Administration Guide and Reference for details about the DLM commands.

37 DATALINK value referenced file is being updated. First, issue a SQL UPDATE with the DLNEWCOPY or DLPREVIOUSCOPY scalar function on the DATALINK column to commit or backout the file changes. Then retry the same statement again.

38 It is a restriction that DATALINK value with DFS scheme cannot be assigned to a DATALINK column defined with WRITE PERMISSION ADMIN. Move the file to another type of file system and assign the DATALINK value with a different scheme.

39 Use DLVALUE scalar function instead.

40 Check whether the archive server is up and running. Then retry it again.

41 Retry the UPDATE operation again in a separate transaction.

42 Use either a zero-length string or a null as the replacement file location.

43 Commit the current transaction. The retry it again.

44 Use another name for the replacement file or retry it later.

45 Do not link this file or retry it later.

46 Correct the string format of the replacement file location.

47 Make sure a file is specified. Use the actual file name, not the symbolic link for the replacement file.

48 Unlink the file.

49 Check that the correct file is specified and that the file exists.

50 Remove the token from the data location value.

sqlcode: -358

sqlstate: 428D1

SQL0359N   The range of values for the identity column or sequence is exhausted.

Explanation: DB2 attempted to generate a value for an identity column or sequence object, however all allowable values have already been assigned.

The statement cannot be processed.

User Response: For an identity column, redefine the table with a larger range of values for the identity column. If a MAXVALUE or MINVALUE specification has been made that limits the range of values to be less than the range for the data type of the column, then the column can be altered to expand the range of valid values. Otherwise, the identity column must be recreated which requires that the table be recreated. First drop the existing table, and then recreate the table with a different data type for the identity column, specifying a data type that has a larger range of values than the current data type for the identity column.

For a sequence object, redefine the sequence with a larger range of values. If a MAXVALUE or MINVALUE clause limits the range of values to less than the range for the data type of the sequence object, then alter the sequence to expand the range of valid values. Otherwise, drop the sequence object and reissue the CREATE SEQUENCE statement specifying a data type that allows a larger range of values.

sqlcode: -359

sqlstate: 23522

SQL0360W   DATALINK values may not be valid because the table *table-name* **is in Datalink Reconcile Pending (DRP) or Datalink Reconcile Not Possible (DRNP) state.**

Explanation: DATALINK values in table *table-name* may not be valid because the table is either in Datalink Reconcile Pending (DRP) or Datalink Reconcile Not Possible (DRNP) state.

While in either of these states, control of the files on the DB2 Data Links Manager is not guaranteed.

Statement processing continues.

User Response: Refer to the Administration Guide for information on Datalink Reconcile Pending (DRP) and Datalink Reconcile Not Possible (DRNP) states to take appropriate action.

sqlcode: +360

sqlstate: 01627

SQL0368N   **The DB2 Data Links Manager** *dlm-name* **is not registered to the database.**

Explanation: The DB2 Data Links Manager *dlm-name* is not registered to the database. Registered DB2 Data Links Managers are ignored if the database manager configuration parameter DATALINKS is set to NO. The DB2 Data Links Manager may have been dropped using the DROP DATALINKS MANAGER command. It is possible that a new incarnation of the DB2 Data Links Manager with the same name is currently registered; in that case, the error pertains to one or more previously dropped incarnations of that DB2 Data Links Manager.

The statement cannot be processed.

User Response: Ensure that the database manager configuration parameter DATALINKS is set to YES. DATALINK values that are links to previously dropped DB2 Data Links Managers should be removed using the reconcile utility. For additional details, see the usage notes for the DROP DATALINKS MANAGER command in the *Command Reference*.

sqlcode: -368

sqlstate: 55022

SQL0370N   **The parameter in position** *n* **must be named in the CREATE FUNCTION statement for LANGUAGE SQL function** *name*.

Explanation: All parameters for functions defined with LANGUAGE SQL must have a *parameter-name* for each parameter.

The statement cannot be processed.

User Response: Include a parameter name for each parameter of the function.

sqlcode: -370

sqlstate: 42601

SQL0372N   **Only one IDENTITY or ROWID column is allowed in a table.**

Explanation: An attempt was made to do one of the following:

- Create a table with more than one IDENTITY column.
- Add an IDENTITY column to a table which already has one.
- Create a table with more than one ROWID column.
- Add a ROWID column to a table which already has one.

The statement cannot be processed.

User Response: For a CREATE TABLE statement, select only one column to have the ROWID data type or the IDENTITY attribute. For an ALTER TABLE statement, a ROWID column or IDENTITY column already exists for the table. Do not attempt to add another column with data type ROWID or with the IDENTITY attribute to the table.

sqlcode: -372

sqlstate: 428C1

SQL0373N **The DEFAULT clause cannot be specified for GENERATED column** *column-name*.

Explanation: A DEFAULT clause may not be specified for a column which has been identified as a GENERATED column.

The statement cannot be processed.

User Response: Remove the DEFAULT clause and resubmit the statement.

sqlcode: -373

sqlstate: 42623

SQL0374N **The** *clause* **clause has not been specified in the CREATE FUNCTION statement for LANGUAGE SQL function** *function-name* **but an examination of the function body reveals that it should be specified.**

Explanation: The following situations may be the cause of this error.

NOT DETERMINISTIC must be specified if either of the following conditions apply within the body of the function:

- a function that has the NOT DETERMINISTIC property is called,
- a special register is accessed.

READS SQL DATA must be specified if the body of the function defined with LANGUAGE SQL contains a subselect or if it calls a function that can read sql data.

EXTERNAL ACTION must be specified if the body of the function defined with LANGUAGE SQL calls a function that has the EXTERNAL ACTION property.

The statement cannot be processed.

User Response: Either specify the clause or change the function body.

sqlcode: -374

sqlstate: 428C2

SQL0385W **Assignment to an SQLSTATE or SQLCODE variable in an SQL routine may be over-written and does not activate any handler.**

Explanation: The SQL routine includes at least one statement that assigns a value to the SQLSTATE or SQLCODE special variables. These variables are assigned values by processing of SQL statements in the SQL routine. The value assigned may therefore be over-written as a result of the SQL statement processing. Furthermore, the assignment of a value to the SQLSTATE special variable does not activate any handlers.

The routine definition was processed successfully.

User Response: None required. Remove any assignments to the SQLSTATE or SQLCODE special variable to prevent the warning.

sqlcode: +385

sqlstate: 01643

SQL0388N **In a CREATE CAST statement for function** *function-name*, **either the source** *source-data-type-name* **and target** *target-data-type-name* **are both built-in types or they are the same type.**

Explanation: One of the data types must be a user-defined type. The source type and the target type cannot be the same data type.

The statement cannot be processed.

User Response: Change the data type of either the source or the target.

sqlcode: -388

sqlstate: 428DF

SQL0389N **The specific function instance** *specific-name* **identified in a CREATE CAST statement either has more than one parameter, has a parameter that does not match the source data type, or returns a data type that does not match the target.**

Explanation: A cast function must

- have exactly one parameter,
- the data type of the parameter must be the same as the source data type,
- the result data type must be the same as the target data type.

The statement cannot be processed.

User Response: Either chose a different function instance, change the source data type, or change the target data type.

sqlcode: -389

sqlstate: 428DG

SQL0390N **The function** *function-name* **resolved to specific function** *specific-name* **that is not valid in the context where it is used.**

Explanation: A function resolved to a specific function that is not valid in the context where it is used. If *specific-name* is an empty string, then the function resolved to the built-in function identified by *function-name*. Some of the possible situations include:

- the specific function is a table function where only a scalar, column, or row function is expected (such as creating a sourced scalar function).
- the specific function is a scalar, column, or row function where only a table function is expected (such as in the FROM clause of a query).
- the specific function is a row function where only a scalar or column function is expected.

The statement cannot be processed.

User Response: Ensure that the correct function name and arguments are specified and that the current path includes the schema where the correct function is defined. You may need to change the function name, the current path (using SET CURRENT FUNCTION PATH or the FUNCPATH bind option), or change the context in which the function is used.

sqlcode: -390

sqlstate: 42887

SQL0391N **Invalid use of the row based function** *function-name*.

Explanation: The statement uses a row based function *function_name* that cannot be used for one of the following reasons:

- The function is used in a GROUP BY or a HAVING clause but is not also included in the select list.
- The function cannot be used in this context because of the recursive nature of the statement.
- The function cannot be used in a check constraint.
- The function cannot be used in a generated column.
- The function cannot be used in a view definition where the WITH CHECK OPTION clause is specified or any view dependent on such a view having the WITH CHECK OPTION clause specified.
- The function has an argument that does not resolve to a row of a base table. This would include the situation involving a result column of an outer join where NULL producing rows are possible.
- The function cannot be used on rows from a replicated materialized query table.

The statement cannot be processed.

User Response: Remove *function-name* from the context where it is not allowed.

sqlcode: -391

sqlstate: 42881

SQL0392N SQLDA provided for cursor *cursor* **has been changed from the previous fetch.**

Explanation: The application is running with **DB2 rules**, and has requested that LOB data be returned as a LOB in one FETCH statement, and as a locator in another FETCH statement. This is not permitted.

User Response: The statement cannot be executed.

Either do not use **DB2 rules**, or change the application to not change the data type code from LOB to locator (or the reverse) in the SQLDA between successive fetches.

sqlcode: -392

sqlstate: 42855

SQL0396N *Object-type object-name* **(specific name** *specific-name***) attempted to execute an SQL statement during final call processing.**

Explanation: A routine *object-name* (specific name *specific-name*) attempted to execute an SQL statement (other than CLOSE cursor) during FINAL CALL (call-type = 255) processing. This is not allowed.

User Response: Change the routine to not issue SQL statements during FINAL CALL (call-type = 255) processing.

sqlcode: -396

sqlstate: 38505

SQL0400 - SQL0499

SQL0401N The data types of the operands for the operation *operator* **are not compatible.**

Explanation: The operation *operator* appearing within the SQL statement has a mixture of numeric and nonnumeric operands, or the operation operands are not compatible.

Federated system users: this data type violation can be at the data source or at the federated server.

Some 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.

User Response: 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.

Federated system 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.**

Explanation: A nonnumeric operand is specified for the arithmetic function or operator *operator*.

The statement cannot be processed.

User Response: Correct the SQL statement syntax so all specified function or operator operands are numeric.

Federated system 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

SQL0403W **The newly defined alias** *name* **resolved to the object** *name2* **which is currently undefined.**

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: 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

SQL0405N **The numeric literal** *literal* **is not valid because its value is out of range.**

Explanation: 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*.

Federated system users: this range violation can be at the data source or at the federated server. 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 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.

User Response: Reduce the literal value to the appropriate size.

Federated system 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.**

Explanation: 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.

User Response: 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 updatable catalogs.

sqlcode: -406

sqlstate: 22003

SQL0407N **Assignment of a NULL value to a NOT NULL column** *name* **is not allowed.**

Explanation: 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.

If the value for *name* is of the form "TBSPACEID=n1, TABLEID=n2, COLNO=n3", then the column name from the SQL statement was not available when the error was issued. The values provided identify the tablespace, table, and column number of the base table that does not allow NULL value.

Federated system users: this situation can be detected by the federated server or by the data source. Some 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).

User Response: 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.

If the value for *name* is of the form "TBSPACEID=n1, TABLEID=n2, COLNO=n3", you can determine the table name and column

name using the following query:

```
SELECT C.TABSCHEMA, C.TABNAME,
       C.COLNAME
   FROM SYSCAT.TABLES AS T,
        SYSCAT.COLUMNS AS C
   WHERE T.TBSPACEID = n1
   AND T.TABLEID = n2
   AND C.COLNO = n3
   AND C.TABSCHEMA = T.TABSCHEMA
   AND C.TABNAME = T.TABNAME
```

The table and column identified by this query may be the base table of a view for which the SQL statement failed.

Federated system 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   A value is not compatible with the data type of its assignment target. Target name is *name***.**

Explanation: The data type of the value to be assigned to the column, parameter, SQL variable, or transition variable by the SQL statement is incompatible with the declared data type of the assignment target. Both must be:

- Numeric
- Character
- Graphic
- Dates or character
- Times or character
- Timestamps or character
- Datalinks
- The same distinct types
- Reference types where the target type of the value is a subtype of the target type of the column.

- The same user-defined structured types. Or, the static type of the value must be a subtype of the static type (declared type) of the target. If a host variable is involved, the associated built-in type of the host variable must be compatible with the parameter of the TO SQL transform function defined in the transform group for the statement.

The statement cannot be processed.

User Response: Examine the statement and possibly the target table or view to determine the target data type. Ensure the variable, expression, or literal value assigned has the proper data type for the assignment target. For a user-defined structured type, also consider the parameter of the TO SQL transform function defined in the transform group for the statement as an assignment target.

sqlcode: -408

sqlstate: 42821

SQL0409N   The operand of a COUNT function is not valid.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Shorten the specified literal.

sqlcode: -410

sqlstate: 42820

SQL0412N **Multiple columns are returned from a subquery that is allowed only one column.**

Explanation: In the context of the SQL statement, a fullselect is specified that can have only one column as a result.

The statement cannot be processed.

User Response: Specify only one column when only a scalar fullselect is allowed.

sqlcode: -412

sqlstate: 42823

SQL0413N **Overflow occurred during numeric data type conversion.**

Explanation: 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.

Federated system users: numeric conversion can occur at the federated server, at data sources, or both.

The statement cannot be processed. No data was retrieved, updated, or deleted.

User Response: 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.

Federated system 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 are not compatible in a fullselect that includes a set operator or in the multiple rows of a VALUES clause of an INSERT or fullselect.**

Explanation: There are various statements where this error may occur.

- It may occur within a SELECT or VALUES statement that includes set operations (UNION, INTERSECT, or EXCEPT). The corresponding columns of the subselects or fullselects that make up the SELECT or VALUES statements are not compatible.

- It may occur within an INSERT statement that is inserting multiple rows. In this case, the corresponding columns of the rows specified in the VALUES clause are not compatible.

- It may occur within a SELECT or VALUES statement where the VALUES clause is used with multiple rows. In this case, the corresponding columns of the rows specified in the VALUES clause are not compatible.

The columns 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 character string constant.

The statement cannot be processed.

User Response: Correct the column names used in the SELECT statements or the expressions in the VALUES clause 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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 **A statement contains a use of a parameter marker that is not valid.**

Explanation: 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
- as a sort key in an ORDER BY clause

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
- in a query captured by DB2 Query Patroller

The statement cannot be processed.

User Response: 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.**

Explanation: 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.

Federated system users: Decimal division can occur at the federated server, at data sources, or

both. The specified decimal division results in an invalid scale for that data source.

The statement cannot be processed.

User Response: 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.

Federated system 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*.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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 **Locator variable** *variable-position* **does not currently represent any value.**

Explanation: A locator variable is in error. Either it has not had a valid result set locator or LOB locator variable value assigned to it, the locator associated with the variable has been freed, or the result set cursor has been closed.

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*.

Instead of an ordinal position, *variable-position* may have the value "function-name RETURNS", which means that the locator value returned from the user-defined function identified by function-name is in error.

User Response: Correct the program so that the locator variables used in the SQL statement have valid 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. Result set locator values are returned by the ASSOCIATE LOCATORS statements. Result set locator values are only valid as long as the underlying SQL cursor is open. If a commit or rollback operation closes an SQL cursor, the

result set locator associated with the cursor is no longer valid.

If this was a WITH RETURN cursor, ensure the cursor is opened before attempting to allocate it.

sqlcode: -423

sqlstate: 0F001

SQL0426N   Dynamic commit invalid for application execution environment.

Explanation: 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.

Federated system users: SQL dynamic COMMIT statements cannot be executed during a pass-through session.

User Response:

- 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.

Federated system users: Either comment out the COMMIT statement or code it as a static statement. Then resubmit your program.

sqlcode: -426

sqlstate: 2D528

SQL0427N   Dynamic rollback invalid for application execution environment.

Explanation: 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.

Federated system users: SQL dynamic ROLLBACK statements cannot be executed during a pass-through session.

User Response:

- 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.

Federated system users: Either comment out the ROLLBACK statement or code it as a static statement. Then resubmit your program.

sqlcode: -427

sqlstate: 2D529

SQL0428N   The SQL statement is only allowed as the first statement in a unit of work.

Explanation: The SQL statement that was issued is required to execute before any other SQL statement that initiates a unit of work. Following are possible situations:

- The SQL statement must be first in the unit of work and SQL has been issued against the connection within the unit of work.
- The SQL statement must be first in the unit of work and a WITH HOLD cursor is currently open for the connection.

Note that when the statement is DISCONNECT ALL, it causes the DISCONNECT to be directed against all connections, so the request will fail if any of the connections violate the above restrictions.

User Response: Issue a COMMIT or ROLLBACK prior to processing the SQL statement. If there are any WITH HOLD cursors, these will need to be closed. If the statement is SET INTEGRITY, remove the COMMIT THRESHOLD clause.

sqlcode: -428

sqlstate: 25001

SQL0429N   **The maximum number of concurrent LOB locators has been exceeded.**

Explanation: A maximum of 32,000 concurrent LOB locators per unit of work, is supported by DB2.

User Response: 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.**

Explanation: An abnormal termination has occurred while the named UDF was in control.

User Response: 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.**

Explanation: A user/client interrupt has occurred while the named UDF was in control.

User Response: 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 or reference target type name** *udt-name***.**

Explanation: A parameter marker in the statement has been determined as having the user-defined type *udt-name* or a reference type with the target type *udt-name* based on the context in which it is used. A parameter marker cannot have a user-defined type or reference 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 or reference data type using the CAST specification.

The statement cannot be processed.

User Response: Use an explicit cast to the user-defined distinct data type or reference data type for the parameter marker. An alternative is to cast the columns that are user-defined distinct data types to their corresponding source data type or columns that are reference data types to their corresponding representation type.

sqlcode: -432

sqlstate: 42841

SQL0433N   **Value** *value* **is too long.**

Explanation: 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.

User Response: 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***.**

Explanation: The value that was specified for clause *clause* is not supported and has been replaced with the identified supported value *value*.

User Response: 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.**

Explanation: The SQLSTATE specified in the RAISE_ERROR function does not conform to the rules for an application defined SQLSTATE.

User Response: 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.

Explanation: 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.

User Response: 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***.**

Explanation: 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.

User Response: 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 1,2,4)
- 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)
- Reissue RUNSTATS for both the tables involved in the query and their corresponding indexes, i.e. use the AND INDEXES ALL clause so that table and index statistics are consistent (Reason code 6)

sqlcode: +437

sqlstate: 01602

SQL0438N **Application raised error with diagnostic text:** *text*.

Explanation: This error or warning occurred as a result of execution of the RAISE_ERROR function or the SIGNAL SQLSTATE statement in a trigger. An SQLSTATE value that starts with '01' or '02' indicates a warning.

User Response: See application documentation.

sqlcode: -438, +438

sqlstate: application-defined

SQL0439N **User defined function** *function-name* **is indirectly implemented by function** *source-function* **which resulted in error** *sqlcode*.

Explanation: 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*.

User Response: 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 authorized routine named** *routine-name* **of type** *routine-type* **having compatible arguments was found.**

Explanation: This occurs in a reference to routine *routine-name*, when the database manager cannot find a routine it can use to implement the reference. There are several reasons why this could occur:

- *routine-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 SQL path does not contain the schema to which the desired function or method belongs, and an unqualified reference was used.
- The wrong number of arguments were included.
- The right number of arguments were included in the function or method reference, but the data types of one or more of the arguments is incorrect.
- The routine does not exist in the database with the same function id that was used when the package was bound (applies to static statements).
- The mutator method corresponding to an attribute assignment used in an UPDATE statement could not be found. The data type of the new value for the attribute is not a data type that is the same or promotable to the data type of the attribute.
- The routine invoker is not authorized to execute the routine.

User Response: Fix the problem and retry. This could involve catalog access, a change to the statement, a grant of the execute privilege to the routine invoker, the addition of new functions, and/or a change to the SQL path.

sqlcode: -440

sqlstate: 42884

SQL0441N **Invalid use of keyword DISTINCT or ALL with function** *function-name***.**

Explanation: There are several possible causes.

- The keyword DISTINCT or ALL 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 or ALL with a scalar function is invalid.
- The keyword DISTINCT is used in a column function for which it is not supported. These functions include COVARIANCE, CORRELATION and the linear regression functions (those that start with REGR).

- The function was expected to be a column function supporting the keyword ALL or DISTINCT but the function it resolved to was not a column function.

User Response:
- If a scalar function is being used, then remove the keyword DISTINCT or ALL. It is invalid for a scalar function.
- If the function is a column function that does not support the DISTINCT or ALL keywords, then remove the keywords.
- 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 routine** *routine-name***. The maximum number of allowable arguments (90) has been exceeded.**

Explanation: Too many arguments were specified in the reference to routine *routine-name*. The maximum allowable is 90.

User Response: Correct the statement by ensuring that the correct number of arguments has been used and try again.

sqlcode: -442

sqlstate: 54023

SQL0443N **Routine** *routine-name* **(specific name** *specific-name***) has returned an error SQLSTATE with diagnostic text** *text***.**

Explanation: An SQLSTATE was returned to DB2 by routine *routine-name* (specific name *specific-name*), along with message text *text*. The

routine could be a user-defined function or a user-defined method.

User Response: The user will need to understand the meaning of the error. See your Database Administrator, or the author of the routine.

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
12	Out of memory
13	Unexpected error

Errors that are detected by the IBM supplied routines in the SYSIBM or SYSPROC schemas and IBM supplied procedures in the SYSFUN schema all return the SQLSTATE 38553. The message text portion of the message contains a message number that may be an SQLCODE, (for example, SQL0572N), a DBA error message (for example, DBA4747), or some other indication from the routine as to what error was encountered. Note that if the message for the

message number would normally contain tokens, these token values are only available in the db2diag.log file.

sqlcode: -443

sqlstate: 38553

SQL0444N **Routine** *routine-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*.

Explanation: The DBMS is trying to access the body of the code that implements routine *routine-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 routine 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 routine 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 routine.)

User Response: Given for each reason code:

1 Path name *library-or-path* is longer than the maximum (255 bytes). Either the routine 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 routine creator or your database administrator. The routine

definition or the location of the routine itself will need to be corrected.

4 The file in *library-or-path* could not be found. See the routine creator or your database administrator. The routine definition or the location of the routine may need to be corrected, or the routine may need to be re-linked.

In OS/2 this reason code can arise if the UDF DLL name is longer than format (8.3). For example a name of "abcdefgh99.dll" has format (10.3) and would result in this message with reason code 4. The solution is to change the name to an acceptable form, for example "abcdef99.dll".

In addition to the above, this reason code can result if the routine requires a shared library or DLL, and the shared library cannot be located (using the concatenation of directories specified in the LIBPATH environment variable in UNIX-based systems, the PATH environment variable in INTEL systems). There can be multiple levels of this indirection involved, leading to this reason code. For example, routine body X can be found, and it needs shared library Y which can also be found. But Y needs Z, and Z cannot be located, and this will result in SQL0444N reason code 4.

5 There was insufficient memory to load the library containing the function or one or more symbols could not be resolved. Contact the routine creator or your database administrator to make sure that the library was correctly linked. All required libraries to resolve referenced symbols such as external functions must be available. If a lack of memory is determined then 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 routine creator or your database

administrator. The routine 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 routine creator or your database administrator. The routine 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 routine 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.

15 Access has been denied. This can happen in Windows NT environments if the EXTERNAL NAME specification in the routine definition statement does not specify a full path, or the function does not exist in the <db2 installation path>\function directory, necessitating a search using the PATH environment variable. For example, If the PATH

contains a LAN drive prior to the directory containing your function and the DB2 instance is running under the SYSTEM account, this reason code can result. Verify that the EXTERNAL NAME specification in the routine definition statement contains the full path, or the function exists in the <db2 installation path>\function directory. If either of the above is correct and the error occurs due to DB2 instance running under an account which cannot access a LAN drive prior to the directory containing your function in the PATH environment variable, you need to reconfigure the DB2 service to run under a user account that has access to all the network paths that are in the PATH environment variable. When assigning a user account to the DB2 service, make sure that the account is granted all the required advanced user rights as documented in the Quick Beginnings book.

other: An unidentified system failure has occurred. Note the code, and see your system administrator.

If unable to diagnose the error using the information in this message, it may be necessary to contact your system administrator for assistance.

Federated system users: if this user defined function is a function template (and thus no code is required to reside on the federated server), 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.**

Explanation: 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 routine (user-defined function (UDF) or method), and is being transformed because of a CAST FROM specification in the routine definition, or because the UDF is sourced on another function and the result needed to be transformed.

User Response: 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***.**

Explanation: The *clause* keyword was included more than once in the statement. This is a warning condition.

User Response: 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 defining routine** *routine-name***. The maximum number of allowable parameters (90 for user defined functions and methods, 32767 for stored procedures) has been exceeded.**

Explanation: Too many parameters were specified when defining routine *routine-name*. The routine definition statement may be a CREATE FUNCTION, CREATE PROCEDURE, CREATE TYPE (method definition), or ALTER TYPE (method definition).

User Response: Change the statement to include fewer parameters.

sqlcode: -448

SQL0449N   **The statement defining routine**
routine-name contains an invalidly
formatted library/function
identification in the EXTERNAL
NAME clause.

Explanation: An error was found in the
EXTERNAL NAME clause of the CREATE
statement for user defined function (UDF),
user-defined method, or stored procedure
routine-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 (for
  example, (on AIX) /u/slick/udfs/math or (on
  OS/2) d:\myfunc\math)

- a file name (for example, 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 routine.

User Response: 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   **Routine routine-name (specific**
name specific-name) has generated
a result value, SQLSTATE value,
message text, or scratchpad which
is too long.

Explanation: Upon return from routine
routine-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 routine
  definition). There are several possible causes:

  - Too many bytes were moved to the result
    buffer.

  - The data type is one where the data value
    must be delimited by a null, such as
    VARCHAR(n), and the delimiting null was
    not within the range of the defined size.

  - DB2 is expecting a two- or four- byte length
    value preceding the value, and this length
    exceeds the defined size of the result.

  - A LOB locator is returned by the routine,
    and the length of the LOB value associated
    with this locator exceeds the defined size of
    the result.

  The definition of the result argument in the
  Routine must conform to the requirement for
  the data type. Refer to the *Application
  Development Guide* for more information.

- the SQLSTATE value (6 bytes including the
  null terminator)

- the message text (71 bytes including the null
  terminator).

- the scratchpad content (length declared on
  CREATE FUNCTION).

This is not permitted.

This error will also be returned if the length field
of the scratchpad is altered by the routine.

User Response: See your Database
Administrator, or the author of the routine.

sqlcode: -450

sqlstate: 39501

SQL0451N   **The data-item definition, in the**
statement that defines routine
routine-name, contains a data type
type that is not appropriate for a
non-sourced routine written in the
given language.

Explanation: An error was made in the
data-item part of the statement defining routine
routine-name. The user's statement contained the
invalid type *type*, or it contained a user defined
type (UDT) which is based on the invalid type
type. The routine definition may be a CREATE

FUNCTION, CREATE PROCEDURE, CREATE TYPE (method definition), or ALTER TYPE (method definition).

data-item is a token that identifies the area of the problem in the statement. For example, "PARAMETER 2" or "RETURNS" or "CAST FROM".

User Response: Determine which situation has occurred, and take corrective action. Possible corrective actions include:

*   Changing the routine definition to a supported type; for example, from DECIMAL to FLOAT. This may involve a change in the routine body itself, and may also involve use of cast functions upon use of the routine.
*   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*.

Explanation: 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:

```
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.

User Response:

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 statement that defines the routine** *routine-name*.

Explanation: A problem casting the result of routine *routine-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.

User Response: 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 definition for routine** *routine-name* **matches the signature of some other routine that already exists in the schema or for the type.**

Explanation: The signature of a function consists of the 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).

The signature of a method consists of the method name, the subject type of the method, the number of parameters defined for the method, and an ordered list of the types of the parameters (without regard to any parameters of the types).

The signature of a procedure consists of the procedure name and the number of parameters defined for the procedure (data types are not considered).

In this case, either:

• there is a function or procedure (*routine-name*) already in the schema having the same signature as the function or procedure being created

• there is a method (*routine-name*) for the subject type having the same signature as the method specification being added or the method body being created

The statement cannot be processed.

User Response: Determine if the existing routine already provides the functionality desired. If not, then the signature of the routine will have to be changed; for example, by changing the routine name.

sqlcode: -454

sqlstate: 42723

SQL0455N   **In routine** *routine-name***, the schema name** *schema-name1* **provided for the SPECIFIC name does not match the schema name** *schema-name2* **of the routine.**

Explanation: 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 *routine-name*. Note that the *schema-name2* portion of *routine-name* may have been specified directly or it may have defaulted to the authorization ID of the statement. If the routine is a method, *schema-name* refers to the schema name of the subject type of the method.

User Response: Correct the statement and try again.

sqlcode: -455

sqlstate: 42882

SQL0456N   **In the definition of routine** *routine-name***, the SPECIFIC name** *specific-name* **already exists in the schema.**

Explanation: The user has supplied an explicit SPECIFIC name *specific-name* in the definition of routine *routine-name*, but this name already exists as the SPECIFIC name for a function, method, or procedure in the schema.

User Response: Choose a new SPECIFIC name and try again.

sqlcode: -456

sqlstate: 42710

SQL0457N   **A function, method, user-defined data type, or structured data type attribute cannot be called** *name* **since it is reserved for system use.**

Explanation: The user-defined function, method, user-defined data type, or structured data type cannot be created because a specified name is reserved for use by the system. Names that cannot be used for function names, distinct type names, structured type names, or attribute names include:

```
"=","<",">",">=","<=",
"&=","&>",,"&<",
"!=","!>","!<","<>",
SOME, ANY, ALL, NOT, AND, OR,
BETWEEN, NULL, LIKE, EXISTS, IN,
UNIQUE, OVERLAPS, SIMILAR, and MATCH.
```

User Response: Select a name for the function, method, user-defined data type, or structured data type attribute that is not reserved for system use.

sqlcode: -457

sqlstate: 42939

SQL0458N   **In a reference to routine** *routine-name* **by signature, a matching routine could not be found.**

Explanation: In a reference to function, method, or stored procedure *routine-name* by signature, no matching function, method, or stored procedure could be found.

If a data type is used that can accept a parameter, then the type parameter is optional. For example, for CHAR(12), you can either 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 tell the DBMS to ignore data type parameters when finding a matching function.

Note also that in the DROP FUNCTION/PROCEDURE and COMMENT ON FUNCTION/PROCEDURE 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.

A function cannot be sourced on the COALESCE, NULLIF, NODENUMBER, PARTITION, RAISE_ERROR, TYPE_ID, TYPE_NAME,

TYPE_SCHEMA, or VALUE built-in functions.

The statement cannot be processed.

User Response: 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 or procedure instead of a signature.

sqlcode: -458

sqlstate: 42883

SQL0461N   **A value with data type** *source-data-type* **cannot be CAST to type** *target-data-type***.**

Explanation: The statement contains a CAST with the first operand having a data type of *source-data-type* to be cast to the data type *target-data-type*. This cast is not supported.

User Response: Change the data type of either the source or target so that the cast is supported. For predefined data types these are documented in the SQL Reference. For a cast involving a user-defined distinct type, the cast can be between the base data type and the user-defined distinct type or from a data type that is promotable to the base data type to the user-defined distinct type.

sqlcode: -461

sqlstate: 42846

SQL0462W   **Routine** *routine-name* **(specific name** *specific-name***) has returned a warning SQLSTATE, with diagnostic text** *text***.**

Explanation: An SQLSTATE of the form 01Hxx was returned to DB2 by routine *routine-name* (with specific name *specific-name*), along with message text *text*.

User Response: The user will need to understand the meaning of the warning. See

your database administrator, or the author of the routine.

sqlcode: +462

sqlstate: 01Hxx

SQL0463N   **Routine** *routine-name* **(specific name** *specific-name***) has returned an invalid SQLSTATE** *state***, with diagnostic text** *text***.**

Explanation: The valid SQLSTATEs that a routine can return are 38xxx (error), 38502 (error) and 01Hxx (warning). This routine *routine-name* (specific name *specific-name*) returned an invalid SQLSTATE *state*, along with message text *text*. The routine is in error.

User Response: The routine will need to be corrected. See your database administrator, or the author of the routine. The application significance of the bad SQLSTATE can also be learned from the routine author.

sqlcode: -463

sqlstate: 39001

SQL0464W   **Procedure** *procedure-name* **returned** *generated-nbr-results* **query result sets, which exceeds the defined limit** *max-nbr-results***.**

Explanation: The stored procedure named by *procedure-name* completed normally. However, the stored procedure exceeded the defined limit on the number of query result sets the procedure can return.

generated-nbr-results
    identifies the number of query result sets returned by the stored procedure.

max-nbr-results
    identifies the defined limit on the number of query result sets for the stored procedure.

Only the first *max-nbr-results* query result sets are returned to the SQL program that issued the SQL CALL statement.

The possible causes are as follows: The stored procedure is unable to return *generated-nbr-results* result sets due to the DRDA limitations imposed by the client. The DRDA client establishes this limit with the MAXRSLCNT DDM code point.

User Response: The SQL statement is successful. The SQLWARN9 field is set to 'Z'.

sqlcode: +464

sqlstate: 0100E

SQL0465N   Unable to start, initialize, or communicate with fenced mode process. Reason code *code*.

Explanation: There is a system related problem involved with the running of a fenced mode routine (user-defined function or method). The exact nature of the problem is indicated by *code*. This is NOT a user problem. Possible reason codes are:

Routine process errors

21:   Failure to initialize internal data or application data.

22:   Failure to register signal handlers.

23:   Failure to grant agent process the access permission for the REQUEST QUEUE.

24:   Failure to connect to the routine process Shared Memory.

25:   Failure to open the REPLY QUEUE.

26:   Failure to write to the REPLY QUEUE.

27:   Failure to create the REQUEST QUEUE.

28:   Failure to read from the REQUEST QUEUE.

29:   Routine process dies.

30:   Routine process catches an USER INTERRUPT signal.

31:   Failure to unload routine module.

32:   Failure to allocate storage for control blocks used in module loading/unloading.

33:   Failure to send SIGINT from agent process to routine process.

34:   Failure to initialize OLE library.

35:   Failure to initialize OLE DB Initialization Service Component.

40:   Internal error occurred in routine process.

Agent process errors

41:   Failure to spawn routine process.

42:   Failure to create the REPLY QUEUE.

43:   Failure to read from the REPLY QUEUE.

44:   Failure to open the REQUEST QUEUE.

45:   Failure to write to the REQUEST QUEUE.

47:   Failure to grant the routine process the access permission to the UDFP shared memory set.

48:   Failure to grant the routine process the access permission to the REPLY QUEUE.

49:   Failure to allocate storage for control blocks used in module loading/unloading.

50:   Agent process died while running routine code or agent code.

51:   Agent process catches USER INTERRUPT while running un-fenced routine code.

60:   Internal error occurred in routine process.

User Response: Contact your Database or System administrator.

sqlcode: -465

sqlstate: 58032

SQL0466W   The procedure *procedure-name* **returns** *number-results* **result sets from the stored procedure.**

Explanation: This message is returned as a result of issuing a CALL SQL statement. It

indicates that the stored procedure *procedure-name* has *number-results* result sets associated with it.

The statement completed successfully.

User Response: None required.

sqlcode: +466

sqlstate: 0100C

SQL0467W **The procedure** *procedure-name* **includes another result set. There are a total of** *max-nbr-results* **result sets.**

Explanation: This message is returned as a result of closing a cursor. It indicates that another result set exists for stored procedure *procedure-name* and the cursor was re-opened on the next result set. There are a total of *max-nbr-results* result sets from the stored procedure.

The statement completed successfully.

User Response: None required. Fetches can proceed on to the next result set.

sqlcode: +467

sqlstate: 0100D

SQL0469N **The parameter mode (IN, OUT, or INOUT) is not valid for a parameter in procedure** *procedure-name* **with specific name** *specific-name* **(parameter number** *number*, **name** *parameter-name***).**

Explanation: One of the following errors occurred:

- a parameter in an SQL procedure is declared as OUT and is used as input in the procedure body
- a parameter in an SQL procedure is declared as IN and is modified in the procedure body.

User Response: Change the attribute of the parameter to INOUT, or change the use of the parameter within the procedure.

sqlcode: -469

sqlstate: 42886

SQL0470N **The user defined routine** *routine-name* **(specific name** *specific-name***) has a null value for argument** *argument* **that could not be passed.**

Explanation: A routine has an input argument with a null value but was defined with a parameter style that does not pass null indicators, or the datatype of this parameter does not support null values.

User Response: If the routine is to be called with null values, ensure the parameter style and input types are capable of accepting a null value. For functions, the function could also be created with ″RETURNS NULL ON NULL INPUT″.

sqlcode: -470

sqlstate: 39004

SQL0471N **Invocation of routine** *name* **failed due to reason** *reason-code***.**

Explanation: A routine *name* was invoked on a DB2 Universal Database for OS/390 server. The routine invocation failed because of the condition described by DB2 reason code *reason-code*.

The statement cannot be executed. A DSNX9xx message describing the error might be displayed on the MVS system console.

User Response: Consult the documentation for the DB2 Universal Database for OS/390 server and correct the condition described by the DB2 reason code.

sqlcode: -471

sqlstate: 55023

SQL0472N **One or more cursors were left open by function or method** *routine-name* **(specific name** *specific-name***).**

Explanation: The function or method *routine-name* (specific name *specific-name*) did not close all its open cursors before the invoking

statement completed. Functions and methods must close all cursors before the invoking statement completes.

User Response: Consult with the developer of the function or method. The function or method must be rewritten to ensure that all cursors are closed before the invoking statement completes.

sqlcode: -472

sqlstate: 24517

SQL0473N   A user defined data type having the same name as a system predefined type cannot be created.

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: 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 routine *function-name* was made without a signature, but the routine is not unique in its schema.

Explanation: References to a function or stored procedure without a signature is permitted, but the named function or store procedure *function-name* must be unique in its schema and is not. If the routine is a method, a reference without a signature is permitted but this named method must be unique for the data type.

Note that in the DROP FUNCTION/PROCEDURE and COMMENT ON FUNCTION/PROCEDURE 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.

Federated system users: For a pass-through session, 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.

User Response: Correct the reference by one of the following:

- completing the signature
- using the SPECIFIC name of the desired routine
- changing the SQL path

and try again.

sqlcode: -476

sqlstate: 42725

SQL0478N **DROP or REVOKE on object type** *object-type1* **cannot be processed because there is an object** *object-name*, **of type** *object-type2*, **which depends on it.**

Explanation: The requested DROP or REVOKE cannot be processed because a dependency exists on this *object-type1* There is a restrict dependency defined with objects of type *object-type2* (of which object *object-name* is an example).

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 or affected by the REVOKE.

For example:

- function F1 is sourced on function F2
- view V1 is defined using F1
- an attempt to drop F2 will fail due to the direct dependency of F1 on F2 and the indirect dependency of V1 on F2

User Response: Remove the dependencies on this object and then reissue the request.

sqlcode: -478

sqlstate: 42893

SQL0480N **The procedure** *procedure-name* **has not yet been called.**

Explanation: Either the procedure identified in an ASSOCIATE LOCATORS statement has not yet called within the application process, or the procedure has been called, but an explicit or implicit commit occurred before the statement.

User Response: Correct the statements so that the exact syntax used to specify the procedure name on the CALL statement is the same as that on the ASSOCIATE LOCATORS statement. If an unqualified name is used to CALL the procedure, the one-part name must also be used in the other statements. Reissue the statements.

sqlcode: -0480

sqlstate: 51030

SQL0481N **The GROUP BY clause contains** *element-1* **nested within** *element-2*.

Explanation: The following types of nesting are not allowed within a GROUP BY clause:

- CUBE within CUBE, ROLLUP, or GEL
- ROLLUP within CUBE, ROLLUP, or GEL
- () within CUBE, ROLLUP, or GEL
- GROUPING SET within GROUPING SET, CUBE, ROLLUP, or GEL

where GEL represents the element shown as grouping-expression-list in the syntax diagram of the GROUP BY clause.

In some instances the value "---" will be shown for *element 2*. In this case "---" represents one of CUBE, ROLLUP, GROUPING SET, or GEL.

The statement cannot be processed.

User Response: Modify the GROUP BY clause to remove the nesting.

sqlcode: -481

sqlstate: 428B0

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.**

Explanation: 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.

User Response: 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.**

Explanation: One or more of the data types in the statement is BOOLEAN. This is not supported in the current version of DB2.

User Response: Change the data type(s) then resubmit the statement.

sqlcode: -486

sqlstate: 42991

SQL0487N   **Routine** *routine-name* **(specific name** *specific-name***) attempted to execute an SQL statement.**

Explanation: The program used to implement the body of a routine is not allowed to execute SQL statements. This routine *routine-name* (specific name *specific-name*) contains SQL statements.

User Response: Remove any SQL statements then recompile the program. Investigate the level of SQL allowed as specified on the statement defining the routine.

sqlcode: -487

sqlstate: 38001

SQL0489N   **The function** *function-name* **in a SELECT or VALUES list item has produced a BOOLEAN result.**

Explanation: 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.

User Response: Correct the function name or remove the use of the function.

sqlcode: -489

sqlstate: 42844

SQL0490N   **Number** *number* **directly specified in an SQL statement is outside the range of allowable values in this context (***minval***,***maxval***)**

Explanation: A number (*number*) was specified that is not valid in the context in which it was specified. The minimum allowed value in this context is *minval*. The maximum allowed value in this context is *maxval*. n must be within the range specified by *minval* and *maxval* (*minval* =< n => *maxval*).

User Response: Change the value n to a valid value in the statement.

sqlcode: -490

sqlstate: 428B7

SQL0491N   **The definition of routine** *routine-name* **must have a RETURNS clause, and one of: the EXTERNAL clause (with other required keywords); the RETURN statement; or the SOURCE clause.**

Explanation: A required clause is missing in the definition of routine *routine-name*. If EXTERNAL was specified, one of the following clauses must also be specified: LANGUAGE, PARAMETER STYLE, DETERMINISTIC or NOT DETERMINISTIC, and EXTERNAL ACTION or NO EXTERNAL ACTION.

User Response: 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.**

Explanation: 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.

User Response: 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 **Routine** *routine-name* **(specific name** *specific-name***) has returned a date, time or timestamp value which is syntactically or numerically invalid.**

Explanation: The body of user defined function (UDF) or method *routine-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.

User Response: The routine needs fixing. See your DBA, or the author of the routine.

sqlcode: -493

sqlstate: 22007

SQL0495N **Estimated processor cost of** *estimate-amount1* **processor seconds (***estimate-amount2* **service units) in cost category** *cost-category* **exceeds a resource limit error threshold of** *limit-amount* **service units.**

Explanation:

The prepare of a dynamic INSERT, UPDATE, DELETE, or SELECT SQL statement resulted in a cost estimate that exceeded the error threshold value specified in the resource limit specification table (RLST).

This error is also issued if DB2's cost category value was "B", and the default action specified in the RLF_CATEGORY_B column in the RLST is to issue an error.

estimate_amount1
    The cost estimate (in processor seconds) if the prepared INSERT, UPDATE, DELETE or SELECT statement were to be executed.

estimate_amount2
    The cost estimate (in service units) if the prepared INSERT, UPDATE, DELETE or SELECT statement were to be executed.

cost-category
    DB2's cost-category for this SQL statement. The possible values are A or B.

limit-amount
    The error threshold (in service units) specified in the RLFASUERR column of the RLST.

The prepare of the dynamic INSERT, UPDATE, DELETE, or SELECT statement was unsuccessful.

User Response: If this SQLCODE was returned because the cost category value is "B", it might be that the statement is using parameter markers or that some statistics are not available for the referenced tables and columns. Make sure the administrator has run the utility RUNSTATS on the referenced tables. It might also be that UDFs

will be invoked when the statement is executed, or for INSERT, UPDATE, or DELETE statements that triggers are defined on the changed table. Check the DSN_STATEMNT_TABLE or the IFCID 22 record for this statement to find the reasons this SQL statement has been put in cost category "B". If the program cannot be changed, or if statistics cannot be obtained, ask the administrator to change the value in the RLF_CATEGORY_B column in the RLST to "Y" which allows the statement to execute or "W" which returns a warning instead of an error.

If the warning is caused by an SQL statement that is consuming too much processor resource, attempt to rewrite the statement to perform more efficiently. Another option is to ask the administrator to increase the error threshold value in the RLST.

sqlcode: -495

SQL0500 - SQL0599

SQL0501N	The cursor specified in a FETCH or CLOSE statement is not open.

Explanation: 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.

User Response: 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.

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.

Explanation: The program attempted to execute an OPEN statement for an open cursor.

sqlstate: 57051

SQL0499N	Cursor *cursor-name* has already been assigned to this or another result set from procedure *procedure-name*.

Explanation: An attempt was made to assign a cursor to a result set but multiple cursors have been allocated for procedure *procedure-name*.

User Response: 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

The statement cannot be processed. The cursor was unchanged.

User Response: 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.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The cursor name specified in the DECLARE statement has already been declared.

The statement cannot be processed.

User Response: Ensure the name is spelled correctly.

SQL0507N   **The cursor specified in the UPDATE or DELETE statement is not open.**

Explanation: 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.

User Response: 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.**

Explanation: 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. This includes any use of cursors within a savepoint when a ROLLBACK TO SAVEPOINT is performed.

Federated system 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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 includes :

- 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 cursor references a view that has an INSTEAD OF UPDATE (or DELETE) trigger.

The statement cannot be processed.

User Response: 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 updatable from its definition or context, rebind the program with either the BLOCKING NO or BLOCKING UNAMBIG bind option.

If the database manager is failing the statement and the cursor is based on a view with an INSTEAD OF UPDATE (or DELETE) trigger, use a searched UPDATE (or DELETE) statement.

Federated system 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.**

Explanation: 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
    - An OUTER clause with a typed table or typed 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.

User Response: Do not perform updates on the result table as specified.

Federated system 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.

Explanation: 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.

Federated system users: not all data sources report this warning condition. The federated server attempts to issue this warning whenever the condition exists, but there is no guarantee that the federated server can always detect this condition. Do not rely on this warning to preclude UPDATE/DELETE operations from affecting an entire table or view.

User Response: Ensure that you intend to modify the entire table or view.

SQL0514N   The cursor *name* is not in a prepared state.

Explanation: 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.

User Response: 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 reissued.

sqlcode: -514

sqlstate: 26501

SQL0516N   The DESCRIBE statement does not specify a prepared statement.

Explanation: The statement name in the DESCRIBE statement must specify a statement that was prepared in the same database transaction.

The statement cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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 reissued.

sqlcode: -518

sqlstate: 07003

SQL0519N **The PREPARE statement identifies the SELECT or VALUES statement of the open cursor** *name***.**

Explanation: 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.

User Response: 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

SQL0525N **The SQL statement cannot be executed because it was in error at bind time for section =** *section-number* **package =** *pkgschema.pkgname* **consistency token = 0X***contoken***.**

Explanation: One of the following:

- The statement was in error when the package was bound, but the error was ignored then because the option SQLERROR (CONTINUE) was used. Since the statement contains an error, it cannot be executed.

- The statement might not be an executable statement at this location, or might only be executable by a DB2 application requester

Note that *contoken* is given in hexadecimal.

The statement cannot be executed.

User Response: If the SQL statement is not supposed to execute at the indicated location, then correct the program so that the statement in error does not execute at that location. Precompile, compile, and bind replace the package. If the SQL statement is supposed to execute at the indicated location, correct the problem found and reissue PREP or BIND using ACTION(REPLACE). If multiple versions of the package have been bound, issue the following SELECT statement to determine which version

has the error: SELECT PKGVERSION FROM
SYSCAT.PACKAGES where
PKGSCHEMA='pkgschema' AND PKGNAME =
'pkgname' and HEX(UNIQUE_ID) = 'contoken'

sqlcode: -525

sqlstate: 51015

SQL0526N   **The requested function does not
apply to declared temporary
tables.**

Explanation: The SQL statement being executed
refers to a declared temporary table. A declared
temporary table cannot be used in the given
context.

The statement cannot be processed.

User Response: Modify the SQL statement to
ensure that the object references are not to
declared temporary tables.

sqlcode: -526

sqlstate: 42995

SQL0528N   **The table** *tablename* **already has a
unique constraint that is a
duplicate of constraint** *name*.

Explanation: A UNIQUE clause uses the same
column list as the PRIMARY KEY clause, another
UNIQUE clause or the PRIMARY KEY or a
UNIQUE constraint that already exists for the
table *tablename*. Duplicate unique constraints are
not allowed.

name is the constraint name, if one was specified
or exists. If a constraint name is not specified,
name is the first column name specified in the
column list of the UNIQUE clause followed by
three periods.

The statement cannot be processed.

User Response: Remove the duplicate UNIQUE
clause or change the column list to a set of
columns that is not already part of a unique
constraint.

sqlcode: -528

sqlstate: 42891

SQL0530N   **The insert or update value of the
FOREIGN KEY** *constraint-name* **is
not equal to any value of the
parent key of the parent table.**

Explanation: A value in a foreign key of the
object table is being set, but this value is not
equal to any value of the parent 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
some value of the parent 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 parent key of any
row of the parent table of the associated
relationship at the completion of the statement.

Federated system users: the constraint can exist
on the data source (if the child and parent tables
exist on the data source).

Some 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 statement could not be executed. The
contents of the object table are unchanged.

User Response: Examine the insert or update
value of the foreign key first, and then compare
it with each of the parent key values of the
parent table to determine and correct the
problem.

sqlcode: -530

sqlstate: 23503

SQL0531N   **The parent key in a parent row of
relationship** *constraint-name* **cannot
be updated.**

Explanation: An operation attempted to update
a parent key in the row of the parent table but
the parent key in the specified row has
dependent rows in the dependent table

associated with it in the *constraint-name* constraint.

When the update rule of constraint *constraint-name* is NO ACTION, the value of a parent key in a parent row cannot be updated if the parent row has any dependent rows at the completion of the statement.

When the update rule of constraint *constraint-name* is RESTRICT, the value of a parent key in a parent row cannot be updated if the parent row has any dependent rows at the time the update of the parent key is attempted.

Federated system users: the constraint can exist on the data source (if the child and parent tables exist on the data source).

Some 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 statement could not be executed. The contents of the parent table are unchanged.

User Response: Examine the parent key of the object table and the foreign key of the dependent table to determine if the value of the specified row of the parent 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.**

Explanation: An operation attempted to delete a specified row of the parent table but the parent key in the specified row has dependent rows in the referential constraint *constraint-name* and the delete rule of NO ACTION or RESTRICT is specified for the relationship.

When the delete rule of constraint *constraint-name*

is NO ACTION, a row of the parent table cannot be deleted if the dependent rows are still dependent on the parent key at the completion of the statement.

When the delete rule of constraint *constraint-name* is RESTRICT, a row of the parent table cannot be deleted if the parent row has any dependent rows at the time of the delete.

Note that a delete can cascade to delete other rows in dependent tables that have a delete rule of NO ACTION or RESTRICT. Thus the constraint *constraint-name* may be on a different table than the original delete operation.

Federated system users: the constraint can exist on the data source (if the child and parent tables exist on the data source).

Some 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 statement could not be executed. The contents of the table are unchanged.

User Response: 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.**

Explanation: 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.

Federated system users: the constraint can exist

on the data source (if the child and parent tables exist on the data source).

User Response: 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.

Explanation: 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.

Federated system users: the constraint can exist on federated server (if the child and parent tables exist as tables on federated server), or it can exist on the data source (if the child and parent tables exist on the data source).

User Response: 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.

Explanation: 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.

Federated system users: the constraint can exist on the data source (if the child and parent tables exist on the data source).

User Response: 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.

Explanation: 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.

Federated system users: the constraint can exist on the data source (if the child and parent tables exist on the data source).

Some 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.

User Response: 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 DB2 Connect.

sqlcode: -536

sqlstate: 42914

SQL0537N   **The PRIMARY KEY clause, a FOREIGN KEY clause, a UNIQUE clause, or a PARTITIONING KEY clause identifies column** *name* **more than once.**

Explanation: The column *name* appears more than once in a PRIMARY KEY clause, FOREIGN KEY clause, UNIQUE clause or PARTITIONING KEY clause of a CREATE or ALTER statement.

Federated system users: the constraint can exist on the data source (if the child and parent tables exist on the data source).

Some 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.

User Response: Specify unique names for each column.

sqlcode: -537

sqlstate: 42709

SQL0538N   **FOREIGN KEY** *name* **does not conform to the description of the parent key of table** *table-name*.

Explanation: The definition of the indicated foreign key does not conform to the parent key description of the table *table-name*.

Possible errors are:

- The number of columns in the foreign key column list does not match the number of columns in the parent key column list.
- The number of columns in the foreign key column list does not match the number of columns in the primary key of the parent table (a parent key column list is not specified).

- The descriptions of the corresponding columns 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).

name is the constraint name if specified in the FOREIGN KEY clause. If a constraint name was not specified, *name* is the first column name specified in the clause followed by three periods.

Federated system users: some 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 the data source (if the child and parent tables exist on the data source).

The statement cannot be processed.

User Response: Correct the statement so the foreign key description conforms to the parent key description.

sqlcode: -538

sqlstate: 42830

SQL0539N   **Table** *name* **does not have a** *key-type* **key.**

Explanation: 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.
- The ALTER TABLE statement attempted to drop the partitioning key for the table *name* but the table does not have a partitioning key.

Federated system users: the constraint can exist on the data source (if the child and parent tables exist on the data source).

Some 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.

User Response: When creating a referential constraint, specify the primary key before specifying any foreign keys (constraints).

sqlcode: -539

sqlstate: 42888

SQL0540N   **The definition of table** *table-name* **is incomplete because it lacks a primary index or a required unique index.**

Explanation: The table named was defined with a PRIMARY KEY clause or a UNIQUE clause. Its definition is incomplete, and it cannot be used until a unique index is defined for the primary key (the primary index) and for each set of columns in any UNIQUE clause (the required unique indexes). An attempt was made to use the table in a FOREIGN KEY clause or in an SQL manipulative statement.

The statement cannot be executed.

User Response: Define a primary index or a required unique index on the table before referencing it.

sqlcode: -540

sqlstate: 57001

SQL0541W   **The referential, primary key, or unique constraint** *name* **is ignored because it is a duplicate constraint.**

Explanation: If *name* refers to a referential constraint, then a FOREIGN KEY clause uses the same foreign key and parent table as another FOREIGN KEY clause.

If *name* refers to a primary key or unique constraint, then one of the following situations exists.

- A PRIMARY KEY clause uses the same set of columns as a UNIQUE clause in the statement.
- A UNIQUE clause uses the same set of columns as the PRIMARY KEY clause or another UNIQUE clause in the statement.
- A PRIMARY KEY or a UNIQUE constraint on the same set of columns already exists for the table *tablename*.

name is the constraint name, if specified. If a constraint name was not specified, *name* is the first column name specified in the column list of the FOREIGN KEY or UNIQUE clause followed by three periods.

Federated system users: the constraint can exist on the data source (if the child and parent tables exist on the data source).

Some 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 indicated referential constraint or unique constraint was not created. The statement was processed successfully.

User Response: No action is required. Processing can continue.

sqlcode: +541

sqlstate: 01543

SQL0542N   *name* **cannot be a column of a primary key or unique key because it can contain null values.**

Explanation: The column *name* identified in a PRIMARY KEY clause or UNIQUE clause is defined to allow null values.

Federated system users: some 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.

User Response: Correct the column, primary key or unique 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: 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.**

Explanation: 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 constraint definition contains a dereference operation or a DEREF function where the

scoped reference argument is other than the object identifier (OID) column.
- the constraint definition uses a TYPE predicate.
- the constraint definition includes a CAST specification with a SCOPE clause.

The statement cannot be processed.

User Response: 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: -546

sqlstate: 42621

SQL0548N **A check constraint or generated column that is defined with** *check-condition-element* **is invalid.**

Explanation: 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 function that is not deterministic
- the constraint definition contains a user defined function with external action
- the constraint definition contains a user defined function with the scratchpad option
- the definition contains a user defined function with the CONTAINS SQL or READS SQL DATA option

- the definition contains a reference to a generated column which is based on an expression
- 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 generated column definition contains a reference to itself
- the constraint definition contains a dereference operation or a DEREF function where the scoped reference argument is other than the object identifier (OID) column.
- the constraint definition uses a TYPE predicate.
- the constraint definition includes a CAST specification with a SCOPE clause.
- The constraint or generated column definition contains a text search function.

The token within the text of the error message lists the invalid item.

The statement cannot be processed.

User Response: The user response is one of the following, depending on the cause of the error:
- change the check constraint or generated column 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.

SQL0549N **The** *statement* **statement is not allowed for** *object-type1* *object-name1* **because the bind option DYNAMICRULES RUN is not in effect for the** *object-type2***.**

Explanation: A program attempted to issue the indicated SQL statement that is one of several SQL statements that can be dynamically prepared only in a package exhibiting run

behavior. Those SQL statements are:

- Dynamic GRANT statement
- Dynamic REVOKE statement
- Dynamic ALTER statement
- Dynamic CREATE statement
- Dynamic DROP statement
- Dynamic COMMENT ON statement
- Dynamic RENAME statement
- Dynamic SET INTEGRITY statement
- Dynamic SET EVENT MONITOR STATE statement

statement
> The SQL statement in error

object-type1
> PACKAGE or DBRM. DBRM is valid only in a DRDA connection.

object-name1
> If *object-type1* is PACKAGE, *object-name1* is the name of the package in the format 'location-id.collection-id.package-id'. If *object-type1* is DBRM, *object-name1* is the name of the DBRM in the format 'plan-name DBRM-name'.

object-type2
> PACKAGE or PLAN. PLAN is valid only in a DRDA connection. If *object-type1* is PACKAGE, *object-type2* can be either PACKAGE or PLAN (whichever is bound with DYNAMICRULES(BIND)). If *object-type1* is DBRM, *object-type2* is PLAN.

The SQL statement cannot be executed.

User Response: Do one of the following to correct the error:

- If the SQL statement is embedded, remove it, precompile and compile the application program again, and reissue the BIND command.
- If appropriate, use the SQL statement with a package or plan that is bound with DYNAMICRULES(RUN).

- Issue the REBIND command with the DYNAMICRULES(RUN) option for the plan or package to which the SQL statement is bound.

sqlcode: -549

sqlstate: 42509

SQL0551N *authorization-ID* **does not have the privilege to perform operation** *operation* **on object** *name*.

Explanation: 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.

If attempting to run a DB2 utility or CLI application, the DB2 utility programs may need to be rebound to the database because the userid who created the database no longer exists or no longer has the required privileges.

If this error occurs during invocation of a routine, then authorization ID *authorization-ID* does not have the EXECUTE privilege on any candidate routine in the SQL path. *name* is the name of a candidate routine in the SQL path.

Federated system users: if this message is returned when the user is changing the remote_pw column of the SYSCAT.USEROPTIONS 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.

User Response: Ensure that *authorization-ID* has the authorization necessary to perform the operation.

Federated system users: this authorization can be at the federated server, the data source, or both.

If the DB2 utility programs need to be rebound to the database, the database administrator can accomplish this by issuing one of the following CLP command from the bnd subdirectory of the instance, while connected to the database:

- "DB2 bind @db2ubind.lst blocking all grant public" for the DB2 utilities.
- "DB2 bind @db2cli.lst blocking all grant public" for CLI.

sqlcode: -551

sqlstate: 42501

SQL0552N *authorization-ID* **does not have the privilege to perform operation** *operation***.**

Explanation: Authorization ID *authorization-ID* attempted to perform the specified *operation* without the proper authorization.

Federated system users: some 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.

User Response: Ensure that *authorization-ID* has the authorization to perform the operation.

Federated system users: this authorization can be at the federated server, the data source, or both.

sqlcode: -552

sqlstate: 42502

SQL0553N **An object cannot be created with the schema name** *schema-name***.**

Explanation: 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, SYSFUN, 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, schemata, and aliases) cannot be created with any schema name that starts with the letters SYS.

The statement cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: Remove the authorization ID from the list.

sqlcode: -554

sqlstate: 42502

SQL0555N **An authorization ID cannot revoke a privilege from itself.**

Explanation: 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.

User Response: 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.**

Explanation: The privilege cannot be revoked because *authorization-ID* does not hold the privilege.

The statement cannot be processed.

User Response: 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.

Explanation: 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.

User Response: Correct and resubmit the statement.

sqlcode: -557

sqlstate: 42852

SQL0558N   An attempt to revoke a privilege from *authorization-ID* **was denied because** *authorization-ID* **would still hold** *control* **authority.**

Explanation: *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
- CREATE NOT FENCED ROUTINE

The statement cannot be executed. No privileges are revoked.

User Response: Revoke the *control* privilege, if desired.

sqlcode: -558

sqlstate: 42504

SQL0562N   The specified database privileges cannot be granted to PUBLIC (public).

Explanation: 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.

User Response: The implied function is not supported.

sqlcode: -562

sqlstate: 42508

SQL0567N *authorization-ID* **is not a valid authorization ID.**

Explanation: The authorization ID specified by *authorization-ID* is not valid for one of the following reasons:

- It begins with "SYS", "sys", "IBM", "ibm", "SQL", or "sql".
- 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.

User Response: Correct the invalid authorization ID.

sqlcode: -567

sqlstate: 42602

SQL0569N **The GRANT/REVOKE statement failed because** *authorization-name* **may identify both a user and a group in the system.**

Explanation: The GRANT or REVOKE statement specified an authorization name that may identify both a user and a group in the security namespace, and did not explicitly specify the USER or GROUP keyword in the statement. The statement is therefore ambiguous. Note that when using DCE security, the USER or GROUP keyword is always required.

User Response: Change the statement to explicitly specify the USER or GROUP keyword to uniquely identify the required authorization id.

sqlcode: -569

sqlstate: 56092

SQL0570W **Not all requested privileges on object** *object-name* **of type** *object-type* **were granted.**

Explanation: A GRANT operation was attempted on object *object-name* of type *object-type*, but some or all of the privileges were not granted. The authorization ID that issued the statement does not have all of the privileges to be granted with grant option or does not have DBADM authority.

All valid requested privileges were granted.

User Response: Obtain the required authority and try the the operation again.

sqlcode: +570

sqlstate: 01007

SQL0572N **Package** *pkgname* **is inoperative.**

Explanation: The package *pkgname* is marked as inoperative and must be explicitly rebound (without specifying RESOLVE CONSERVATIVE) before it can be used. This package cannot be used because one or more user-defined functions that it depends upon have been dropped.

User Response: Explicitly rebind the named package using either the REBIND (without specifying RESOLVE CONSERVATIVE) or the BIND command.

sqlcode: -572

sqlstate: 51028

SQL0573N **A column list specified in the references clause of constraint** *name* **does not identify a unique constraint of the parent table** *table-name***.**

Explanation: A list of column names was specified in the references clause for the constraint identified by *name* that does not match the column names of the primary key or any unique key of the referenced table *table-name*.

name is the constraint name, if specified. If a constraint name was not specified, *name* is the first column name specified in the column list of

the FOREIGN KEY clause followed by three periods.

The statement cannot be processed.

User Response: Correct the column list in the references clause or add the unique constraint to the referenced table.

sqlcode: -573

sqlstate: 42890

SQL0574N **DEFAULT value or IDENTITY attribute value is not valid for column** *column-name* **in table** *table-name*. **Reason code:** *reason-code*.

Explanation: The DEFAULT value or IDENTITY attribute 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 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.

10 A value with non-zero scale was specified for the START WITH or INCREMENT BY option of an identity column.

11 The CURRENT SCHEMA special register is specified and the length attribute of the character string data type is less than 128.

<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.

User Response: Correct the DEFAULT value or IDENTITY attribute value based on the reason code that was returned.

sqlcode: -574

sqlstate: 42894

SQL0575N **View or materialized query table** *name* **cannot be used because it has been marked inoperative.**

Explanation: The view or materialized query table *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 or DROP TABLE
- CREATE ALIAS
- CREATE VIEW or CREATE TABLE

The statement cannot be processed.

User Response: If *name* is a view, recreate the view by issuing a CREATE VIEW statement using the same view definition as the inoperative view. If *name* is a materialized query table, recreate the materialized query table by issuing the CREATE TABLE statement using the same materialized query table definition as in the inoperative materialized query table.

sqlcode: -575

sqlstate: 51024

SQL0576N Alias *name* cannot be created for *name2* **as it would result in a repetitive alias chain.**

Explanation: 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.

User Response: 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

SQL0577N User defined routine *routine-name* **(specific name** *specific-name*) **attempted to modify data but was not defined as MODIFIES SQL DATA.**

Explanation: The program used to implement the body of a routine is not allowed to modify SQL data.

User Response: Remove any SQL statements that modify data then recompile the program. Investigate the level of SQL allowed as specified when defining the routine.

sqlcode: -577

sqlstate: 38002

sqlstate: 42985

SQL0579N Routine *routine-name* **(specific name** *specific-name*) **attempted to read data but was not defined as READS SQL DATA or MODIFIES SQL DATA.**

Explanation: The program used to implement the body of a routine is not allowed to read SQL data.

User Response: Remove any SQL statements that read data then recompile the program. Investigate the level of SQL allowed as specified when defining the routine.

sqlcode: -579

sqlstate: 38004

sqlstate: 42985

SQL0580N The result-expressions of a CASE expression cannot all be NULL.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Correct the result-expressions so that they are compatible.

sqlcode: -581

sqlstate: 42804

SQL0582N **A CASE expression in a VALUES clause, IN predicate, GROUP BY clause, or ORDER BY clause cannot include a quantified predicate, an IN predicate using a fullselect, or an EXISTS predicate.**

Explanation: 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 VALUES clause
- an IN predicate
- a GROUP BY clause or
- an ORDER BY clause.

Such CASE expressions are not supported. The CASE expression may be part of a function written in SQL.

The statement cannot be processed.

User Response: Remove the use of the quantified predicate, IN predicate, or EXISTS predicate from the CASE expression. If the CASE expression is part of a function, the query may need to be written without the function that causes the error.

sqlcode: -582

sqlstate: 42625

SQL0583N **The use of routine** *routine-name* **is invalid because it is not deterministic or has an external action.**

Explanation: The routine (function or method) *routine-name* is defined as a non-deterministic routine or a routine with an external action. This type of routine 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)
- in a FILTER clause of a user-defined predicate specification or an index extension definition

The statement cannot be processed.

User Response: If the use of a non-deterministic or external action routine was not intended, substitute a routine without these characteristics. If the behavior associated with the non-deterministic or external action routine 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 routine would get specified in each search-condition.
- Remove the non-deterministic or external action routine from the GROUP BY clause. If grouping is desired on a column of the result that is based on a non-deterministic or external action routine 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 routine 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.
- Remove the non-deterministic or external action routine from the FILTER clause.

sqlcode: -583

sqlstate: 42845

SQL0584N   **Invalid use of NULL or DEFAULT.**

Explanation: 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.

Federated system users: DEFAULT can not be used in a VALUES clause of an INSERT statement where the object is a nickname.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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

SQL0590N   Name *name* **specified in context** *context-tag* **is not unique.**

Explanation: Name *name* is specified as a parameter, SQL variable, cursor, label, or condition in a context defined by *context-tag*. The name is not unique.

If *context-tag* is "BEGIN...END", then the context of the error is a dynamic SQL compound statement. Otherwise, the context of the error is a trigger or routine and *context-tag* is the trigger name or routine name containing the compound statement.

- If *name* is a parameter name, then it must be unique within the parameter list and the EXPRESSION AS clause of a routine.
- If *name* is an SQL variable name, cursor name or condition, then it must be unique within a compound statement .
- A label must be unique within a compound statement and different from the labels of the statements in which it is nested.

User Response: Change the name so that it is unique.

sqlcode: -590

sqlstate: 42734

SQL0595W   Isolation level *requested-level* **has been escalated to** *escalated-level*.

Explanation: The isolation level specified is not supported by DB2. It has been escalated to the next higher level of isolation supported by DB2.

User Response: 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

SQL0597N   Unable to retrieve a DATALINK value. Reason code = *reason-code*.

Explanation: A DATALINK value could not be retrieved. The possible reason codes are as follows:

01   The DB2 Data Links Manager does not authorize the DB2 user to retrieve a DATALINK value embedded with a write token for modifying the DATALINK value referenced file.

User Response: The action is based on the reason code as follows.

01   Contact the DB2 Data Links Manager administrator to grant the write access privilege to this file.

sqlcode: -0597

sqlstate: 42511

SQL0598W   Existing index *name* **is used as the index for the primary key or a unique key.**

Explanation: An index was required for an ALTER TABLE operation that defined a primary key or a unique key, and the indicated index matches the required index.

When creating a primary key or unique key index, an index description matches if it identifies the same set of columns (in any order) as the primary or unique key without regard to ascending or descending specifications, and is specified as unique.

The statement is processed successfully.

User Response: 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.

Explanation: 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.

User Response: No action is required.

sqlcode: +599

sqlstate: 01596

SQL0600 - SQL0699

SQL0600N   A routine *routine-name* **could not be generated because of a duplicate signature or because it would override an existing routine.**

Explanation: During the CREATE or ALTER operation, a system-generated cast function, observer method, mutator method, or constructor function could not be created because another function or method with the same name and signature already exists in the schema, or because the method or function would override an existing method.

User Response: Choose another name for the user-defined type, attribute, or cast function that causes the conflict, or drop the function or method with the same name as the function or method that could not be generated.

sqlcode: -600

sqlstate: 42710

SQL0601N   The name of the object to be created is identical to the existing name *name* **of type** *type***.**

Explanation: 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, UNIQUE, or CHECK CONSTRAINT, the *name* is the constraint name specified in the CREATE or

ALTER TABLE statement or generated by the system.

Federated system users: some 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.

User Response: Either drop the existing object or choose another name for the new object.

Federated system users: if the statement is a CREATE FUNCTION MAPPING or a CREATE TYPE MAPPING statement, the user can also consider not supplying a type mapping name and the system will automatically generate a unique name for this mapping.

sqlcode: -601

sqlstate: 42710

SQL0602N   The CREATE INDEX or CREATE INDEX EXTENSION statement specifies too many columns (16 is the maximum).

Explanation: The number of columns for the index that would be generated by the CREATE INDEX statement exceeds the maximum of 16 columns for the database manager. If the index is defined on a typed table, there is additional

overhead that reduces the maximum number of specified columns to 15.

For a CREATE INDEX EXTENSION statement, the GENERATE KEY function returns a number of columns that exceeds the maximum of 16 columns allowed in an index.

Federated system users: the limit for other data sources is different. That limit could have been exceeded. This problem may be detected on the federated server, or it may be detected on the data source.

The statement cannot be processed.

User Response: Change the index definition to conform to the column limit of 16. For a CREATE INDEX EXTENSION statement, specify a different GENERATE KEY function, or redefine the function to return fewer columns.

Federated system 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.

Explanation: 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.

Federated system users: this situation may also be detected by the data source.

The statement cannot be processed. The specified index is not created.

User Response: 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, structured type, attribute of structured type, function, or type mapping *data-item* is not valid.

Explanation: 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 or the data type containing an error. If the column data type is a structured type, then the explicit or implicit INLINE LENGTH value must be at least 292 and cannot exceed 32677.

• 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". In some cases, it may also be the data type containing the error.

• for a CREATE DISTINCT TYPE statement, *data-item* gives the name of the type being defined or the source data type containing the error.

• for a CREATE or ALTER TYPE statement, *data-item* gives the type of the attribute containing the error or the name of the structured type having an incorrect inline length value. The inline length cannot be smaller than 292 and the size returned by the constructor function for the structured type.

• for CAST(expression AS data-type), *data-item* is "CAST" or the data type containing the error.

Federated system 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.

User Response: Correct the syntax and try again.

sqlcode: -604

sqlstate: 42611

SQL0605W   The index was not created because the index *name* **already exists with the required description.**

Explanation: 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. Also, two index descriptions match if they identify the same columns in the same order with the same or reverse ascending or descending specifications, and at least one description includes the ALLOW REVERSE SCANS parameter.

The new index was not created.

User Response: No action is required unless the existing index *name* is not a suitable index. For

example, the existing index *name* is not a suitable index if it does not allow reverse scans, and the required one does (or vice versa). In this case, the index *name* must be dropped before the required index can be created.

sqlcode: +605

sqlstate: 01550

SQL0606N   The COMMENT ON or LABEL ON statement failed because the specified table or column is not owned by *owner***.**

Explanation: An attempt was made to comment or label on a table or column which does not exist or is not owned by the owner specified in the message text.

Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

sqlcode: -606

sqlstate: 42505

SQL0607N   *operation* **is not defined for system objects.**

Explanation: 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 updatable. For UPDATE operation on these catalog tables, updatable views in SYSSTAT schema should be used. For a description of updatable 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-updatable system table was identified in the FROM clause of a SELECT statement containing a FOR UPDATE clause. For a list of updatable system catalogs see the *SQL Reference*.
- DROP or ALTER of a system table space.
- DROP or ALTER of a system nodegroup.
- REDISTRIBUTE of the IBMCATGROUP or IBMTEMPGROUP nodegroups.

The statement cannot be processed.

User Response: Do not attempt to modify any system objects except for those columns of the system catalog tables which are updatable through updatable SYSSTAT views. See the *SQL Reference* for more information.

sqlcode: -607

sqlstate: 42832

SQL0612N   *name* **is a duplicate name.**

Explanation: A statement was issued with the same name appearing more than once where duplicates are not allowed. Where these names appear varies depending on the type of statement.

- CREATE TABLE statements cannot have the same column name defined for two columns.
- CREATE VIEW statements or common 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. Furthermore, a column name can only be referenced in one ADD or ALTER COLUMN clause in a single ALTER TABLE statement.
- CREATE INDEX cannot have a column name specified more than once as part of the index key or the INCLUDE columns of the index.

- CREATE TRIGGER cannot have a column name specified more than once in the list of columns that cause an update trigger to be activated.
- CREATE TABLE OF statements cannot have the same name defined for the REF IS column and any attribute of the structured type.
- CREATE TYPE statements cannot have the same name defined for two attributes. Attribute names must be unique with the type and all supertypes.
- ALTER TYPE statements cannot add an attribute to a structured type using the name of an attribute that already exists in the type or any of its subtypes or is the same as another attribute being added. Also, the name of the attribute may not be the same as the REF IS column in any table created from the structured type. Furthermore, an attribute name can only be referenced in one ADD or DROP ATTRIBUTE clause in a single ALTER TYPE statement.
- CREATE INDEX EXTENSION statements cannot have the same name defined for two parameters.

The statement cannot be processed.

User Response: Specify unique names as appropriate for the type of statement.

sqlcode: -612

sqlstate: 42711

SQL0613N   **The primary key or a unique key identified by** *name* **is too long or has too many columns.**

Explanation: The sum of the internal lengths of columns in the PRIMARY KEY clause or a UNIQUE clause identified by *name* exceeds 1024 or the number of columns exceeds the maximum of 16. Also, a primary key or a unique key cannot be defined using a LONG VARCHAR column. If the primary key or unique constraint is defined on a typed table, there is additional index overhead that reduces the maximum number of specified columns to 15 and limits the length to 1020.

name is the constraint name, if specified, of the primary key or unique constraint. If a constraint name was not specified, *name* is the first column name specified in the primary key or unique constraint clause followed by three periods.

The statement cannot be processed.

User Response: Modify the primary key or a unique 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 or index extension** *index-name* **cannot be created or altered because the combined length of the specified columns is too long.**

Explanation: The index could not be created or altered because the sum of the key column internal lengths would exceed 1024. Also, an index cannot be created using a LONG VARCHAR, LONG VARGRAPHIC, or LOB column. If the index is defined on a typed table, there is additional index overhead that reduces the maximum length by 4 bytes. The index may be altered by an ALTER TABLE statement that alters the data type of one or more columns.

An index extension could not be created because the sum of the columns returned by the GENERATE KEY function would exceed 1024.

The statement cannot be processed. The specified index or index extension was not created or the table could not be altered.

User Response: To modify the index definition or alter column, eliminate one or more key columns and reduce the key length to the permitted maximum. For an index extension definition, specify a different GENERATE KEY function, or redefine the function to reduce the length of the returned row.

sqlcode: -614

sqlstate: 54008

SQL0615N **The object** *object-name* **of type** *object-type* **cannot be dropped because it is currently in use by the same application process.**

Explanation: A DROP statement of an object cannot be issued when it is in use.

The statement cannot be processed. The object is not dropped.

User Response: Close any cursors that depend either directly or indirectly on the object *object-name* and resubmit the statement.

sqlcode: -615

sqlstate: 55006

SQL0620N **The CREATE TABLE statement failed because** *user-id* **has no private, recoverable dbspaces with less than 255 tables.**

Explanation: The dbspace name was not specified in the CREATE TABLE statement, so the database manager tried to find a private dbspace owned by *user-id*. This message will appear under one of the following conditions:

1. There were no private dbspaces found in the DB2 for VM database for the *user-id*.

2. There were one or more private dbspaces found for the *user-id*, but each contained 255 tables.

3. The private dbspaces were located in nonrecoverable storage pools. Only private dbspaces that exist in recoverable storage pools will be available when the CREATE TABLE statement does not specify the dbspace name.

Processing of the SQL statement has ended.

User Response: Suggestions for the three conditions above are:

1. Acquire a private dbspace in a recoverable storage pool. You may require a database administrator's assistance.

2. Drop a table in a private dbspace residing in a recoverable storage pool to free up an entry, or take the action indicated in (1) above.

3. If you want to create the table in a dbspace existing in a non-recoverable storage pool, then specify the dbspace name on the CREATE TABLE command. Otherwise, take the action indicated in (1) above.

Then rerun the CREATE TABLE statement.

If appropriate, acquire a private dbspace for the user.

sqlcode: -620

sqlstate: 57022

SQL0622N   **The clause** *clause* **is invalid for this database.**

Explanation: The clause indicated is invalid because it is incompatible with the defined characteristics for this database.

Possible reasons:

- CCSID ASCII cannot be specified when connected to a database created using a UNICODE code page.
- CCSID UNICODE cannot be specified when connected to a database created using a non-UNICODE code page.
- FOR SBCS DATA can only be specified when connected to a database created using a single byte code page.
- FOR MIXED DATA can only be specified when connected to a database created using a double byte or UNICODE code page.
- The IN *database-name.table-space-name* or IN DATABASE *database-name* clause was specified using a database name that does not match the name of the database to which the application is currently connected.

The statement cannot be processed.

User Response: Change or remove the clause and re-issue the SQL statement.

sqlcode: -622

sqlstate: 56031

SQL0623N   **A clustering index already exists on table** *name*.

Explanation: The CREATE INDEX statement would create a second clustering index on the specified table. A given table can have only one clustering index.

The statement cannot be executed.

User Response: Check to determine the identity and validity of the existing clustering index on table *name*. Consider creating the index without the CLUSTER attribute.

sqlcode: -623

sqlstate: 55012

SQL0624N   **Table** *name* **already has a** *key-type* **key.**

Explanation: A primary key or partitioning key cannot be defined in an ALTER TABLE statement because the indicated table already has a key of this type.

The statement cannot be executed.

User Response: A table cannot have more than one primary key or partitioning key.

sqlcode: -624

sqlstate: 42889

SQL0628N   **Multiple or conflicting keywords involving the** *clause-type* **clause are present.**

Explanation: 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.
- The keyword may require the specifying of other particular keywords in the same statement which were not specified.
- When ALTERING an object, a keyword was specified that conflicts with an existing property of the object.

User Response: 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.**

Explanation: 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, if specified, in the FOREIGN KEY clause. If a constraint name was not specified, *name* is the first column name specified in the column list of the FOREIGN KEY clause followed by three periods.

The statement cannot be processed.

User Response: 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.**

Explanation: The sum of the column internal lengths identified in the FOREIGN KEY clause in a CREATE TABLE statement exceeds 1024, 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, if specified, in the FOREIGN KEY clause. If a constraint name was

not specified, *name* is the first column name specified in the column list of the FOREIGN KEY clause followed by three periods.

The statement cannot be processed.

User Response: 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 **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***).**

Explanation: 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, if specified, in the FOREIGN KEY clause. If a constraint name was not specified, *name* is the first column name specified in the column list of the FOREIGN KEY clause followed by three periods.

The statement cannot be processed.

User Response: If possible, eliminate the particular FOREIGN KEY clause from the CREATE TABLE or ALTER TABLE statement.

sqlcode: -632

sqlstate: 42915

SQL0633N **The delete rule of FOREIGN KEY** *name* **must be** *delete-rule* **(reason code =** *reason-code***).**

Explanation: 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 required 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, if specified, in the FOREIGN KEY clause. If a constraint name was not specified, *name* is the first column name specified in the column list of the FOREIGN KEY clause followed by three periods.

The statement cannot be processed.

User Response: If possible, change the delete rule.

sqlcode: -633

sqlstate: 42915

SQL0634N **The delete rule of FOREIGN KEY** *name* **must not be CASCADE (reason-code =** *reason-code***).**

Explanation: 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, if specified, in the FOREIGN KEY clause. If a constraint name was not specified, *name* is the first column name specified in the column list of the FOREIGN KEY clause followed by three periods.

The statement cannot be processed.

User Response: If possible, change the delete rule.

sqlcode: -634

sqlstate: 42915

SQL0637N **Keyword** *keyword* **is not valid as used.**

Explanation: The SQL statement contains a duplicate or conflicting specification for a clause with the indicated keyword. For example:

- DEFAULT, UNIQUE, and PRIMARY can only be specified once in a column definition
- UNIQUE and PRIMARY cannot both be specified for the same column definition
- PRIMARY can only be specified once in a CREATE TABLE statement
- the attribute-string specified for the PREPARE statement cannot specify an option more than once, or there are conflicting options

The statement cannot be processed.

User Response: Correct the statement to

specification of duplicate or conflicting clauses.

sqlcode: -637

sqlstate: 42614

SQL0638N **Table** *name* **cannot be created because no column definitions were specified.**

Explanation: The CREATE TABLE statement does not contain any column definitions.

The statement cannot be processed.

User Response: Add one or more column definitions to the statement.

sqlcode: -638

sqlstate: 42601

SQL0644N **Invalid value specified for keyword** *keyword* **in statement** *statement-type***.**

Explanation: 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.

User Response: Determine from reference documentation for the *statement-type* what are valid values and make the appropriate change.

sqlcode: -644

sqlstate: 42615

SQL0647N **Bufferpool** *bufferpool-name* **is currently not active.**

Explanation: The bufferpool *bufferpool-name* is not active in the current database environment. An attempt was made to find another bufferpool with the same page size but there are no such bufferpools active in the current database environment. The bufferpool *bufferpool-name* was recently defined but has not yet been activated.

The statement cannot be processed.

User Response: The database must be stopped

and started again to activate the required bufferpool.

sqlcode: -647

sqlstate: 57003

SQL0648N **The foreign key cannot be defined because it will cause table** *owner1.table-name1* **to be delete-connected to table** *owner2.table-name2* **through multiple paths. Reason Code =** *reason-code***.**

Explanation: The foreign key cannot be defined because one of the following reason codes = *reason-code*:

01 The relationship will cause table *owner1.table-name1* to be delete-connected to table *owner2.table-name2* through multiple paths with the same delete rule of SET NULL.

02 The relationship will cause table *owner1.table-name1* to be delete-connected to table *owner2.table-name2* through multiple paths with different delete rules.

Processing of the SQL statement has ended.

User Response: Correct the statement. Try again.

sqlcode: -648

sqlstate: 42915

SQL0658N **The object** *name* **cannot be explicitly dropped or altered.**

Explanation: The identifier *name* identifies one of the following:

- a cast function or comparison function that cannot be dropped using the DROP statement because it was created by the system for use with a distinct type
- a method that cannot be dropped using the ALTER TYPE method because it was created by the system for use with a structured type

- a package that cannot be dropped using the DROP statement because it was created by the system for use with a SQL procedure. The consistency token (or unique_id) is given in hexadecimal as part of the package name in the form of "pkgschema.pkgname 0Xcontoken".
- a routine that cannot be altered because it is defined with LANGUAGE SQL

The statement cannot be processed.

User Response:
- The function *name* can only be dropped by dropping the distinct type or structured type that caused the function to be defined. The distinct type name corresponds to either the name of the function or the type of the parameters to the function.
- The method *name* can only be dropped by dropping the structured type that caused the method to be defined. The method name corresponds to the name of an attribute for the structured type.
- The package *name* can only be dropped by dropping the SQL procedure that caused the package to be defined. The specific name of the SQL procedure can be found by retrieving the DSCHEMA and DNAME from the SYSIBM.SYSDEPENDENCIES catalog table where the BSCHEMA, BNAME, and BUNIQUE_ID match the relevant parts of *name*, BTYPE is 'K' and DTYPE is 'L'. The consistency token (or unique_id) is needed to distinguish between package versions that share the same schema and name.
- The routine *name* can only be changed by dropping the routine and then redefining it.

sqlcode: -658

sqlstate: 42917

SQL0659N   **Maximum size of a table object has been exceeded.**

Explanation: 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.

User Response: 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 parent key of the parent table.**

Explanation: 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 parent key value in the parent table.

name is the constraint name, if specified. If a constraint name was not specified, *name* is the first column name specified in the column list of the FOREIGN KEY clause followed by three periods.

The statement cannot be processed. The specified table is not altered.

User Response: Remove the erroneous table rows and define the foreign key.

sqlcode: -667

sqlstate: 23520

SQL0668N   Operation not allowed for reason code *reason-code* **on table** *table-name*.

Explanation: Access to table *table-name* is restricted. The cause is based on the following reason codes *reason-code*:

1   The table is in Check Pending state. The integrity of the table is not enforced and the content of the table may be invalid. An operation on a parent table or an underlying table that is not in a check pending state may also receive this error if a dependent table is in a check pending state.

2   The table is in No Data Movement state. When in this state, operations that cause data movement are disallowed. Data movement operations include REDISTRIBUTE, update of partitioning key, update of multi-dimensional clustering key and REORG TABLE.

3   The table is in Load Pending state. A previous LOAD attempt on this table resulted in failure. No access to the table is allowed until the LOAD operation is restarted or terminated.

4   The table is Read Only state. This state can occur during on-line LOAD processing (LOAD INSERT with the READ ACCESS option), or after an on-line LOAD operation, but before all constraints have been validated in the newly appended portion of the table using the SET INTEGRITY statement. No update activity is allowed on this table.

5   The table is in Load In Progress state. The LOAD utility is currently operating on this table, no access is allowed until the LOAD is finished.

User Response:

1   Execute the SET INTEGRITY statement

with the IMMEDIATE CHECKED option on table *table-name* to bring the table out of the Check Pending state.

2   Execute REFRESH TABLE statement on the dependent immediate materialized query tables and staging tables of table *table-name*. The contents of these dependent immediate materialized query tables and staging tables can be incrementally maintained from the appended data of *table-name* through previous LOAD INSERT operations.

3   Restart or terminate the previously failed LOAD operation on this table by issuing LOAD with the RESTART or TERMINATE option respectively.

4   Issue the LOAD QUERY command to check whether the table is in the process of being loaded. If yes, wait until the LOAD utility has completed, or if necessary, restart or terminate previously failed LOAD operation. If LOAD is currently not in progress, issue the SET INTEGRITY command with the IMMEDIATE CHECKED option, to validate constraints in the newly loaded portion of the table.

5   Wait until the current LOAD operation has finished. You can use the LOAD QUERY command to monitor the progress of load.

sqlcode: -668

sqlstate: 57016

SQL0669N   A system required index cannot be dropped explicitly.

Explanation: The DROP INDEX statement attempted to drop an index required to:

- enforce the primary key constraint on a table
- enforce a unique constraint on a table
- enforce the uniqueness of the object identifier (OID) column of a typed table hierarchy
- maintain a replicated materialized query table.

A system required index cannot be dropped using the DROP INDEX statement.

The statement cannot be processed. The specified index is not dropped.

User Response: If you do not want to keep the primary or unique constraint, use the DROP PRIMARY KEY clause or the DROP CONSTRAINT clause of the ALTER TABLE statement to remove the primary key or unique constraint. If the index was created only for enforcing the primary or unique key, then the index will be dropped. If not, the DROP INDEX statement could then be processed.

The index for an OID column can only be dropped by dropping the table.

The index required to maintain a replicated materialized query table can only be dropped by first dropping the replicated materialized query table.

sqlcode: -669

sqlstate: 42917

SQL0670N **The row length of the table exceeded a limit of** *length* **bytes. (Table space** *tablespace-name***.)**

Explanation: The row length of a table in the database manager cannot exceed:

- 4005 bytes in a table space with a 4K page size
- 8101 bytes in a table space with an 8K page size
- 16293 bytes in a table space with an 16K page size
- 32677 bytes in a table space with an 32K page size

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 the following conditions can occur:

- The row length for the table defined in the CREATE TABLE or ALTER TABLE statement exceeds the limit for the page size of the table

space. The regular table space name *tablespace-name* identifies the table space from which the page size was used to determine the limit on the row length.
- The row length for the table defined in the DECLARE GLOBAL TEMPORARY TABLE statement exceeds the limit for the page size of the table space. The user temporary table space name *tablespace-name* identifies the table space from which the page size was used to determine the limit on the row length.

The statement cannot be processed.

User Response: Depending on the cause, do one of the following.

- In the case of CREATE TABLE, ALTER TABLE, or DECLARE GLOBAL TEMPORARY TABLE, specify a table space with a larger pagesize, if possible.
- Otherwise, reduce the row length by eliminating one or more columns or reducing the lengths of one or more columns.

sqlcode: -670

sqlstate: 54010

SQL0672N **Operation DROP not allowed on table** *table-name***.**

Explanation: The DROP operation failed for one of the following reasons:

- The table being dropped has the RESTRICT ON DROP attribute.
- The table space or nodegroup being dropped contains the specified table, which has the RESTRICT ON DROP attribute.

The DROP statement cannot be executed.

User Response: If the problem occurred during a DROP TABLE statement, issue an ALTER TABLE statement using the DROP RESTRICT ON DROP clause. Then re-issue the DROP TABLE statement.

If the problem occurred during a DROP TABLESPACE or DROP NODEGROUP statement, ensure that there are no other tables

within the table space or nodegroup with the RESTRICT ON DROP attribute. The following select statement can help identify the tables:

```
SELECT TB.TABNAME, TB.TABSCHEMA,
        TS.TBSPACE, TS.NGNAME
FROM SYSCAT.TABLES TB,
     SYSCAT.TABLESPACES TS
WHERE TYPE = 'T'
AND DROPRULE = 'R'
AND TB.TBSPACEID = TS.TBSPACEID
ORDER BY TS.NGNAME, TS.TBSPACE,
         TB.TABSCHEMA, TB.TABNAME;
```

After identifying tables with the RESTRICT ON DROP attribute, issue an ALTER TABLE statement using the DROP RESTRICT ON DROP clause for each table with the RESTRICT ON DROP attribute. Then re-issue the DROP TABLESPACE or DROP NODEGROUP statement.

sqlcode: -672

sqlstate: 55035

SQL0673N   **A primary or unique key index cannot be created because the table contains rows which are duplicates with respect to the values of the identified primary or unique key columns of constraint** *name***.**

Explanation: The primary or unique key definition of the constraint identified by *name* failed because the table being altered already contains rows with duplicate values for the PRIMARY KEY or UNIQUE clause columns.

name is the constraint name, if specified. If a constraint name was not specified, *name* is the first column name specified in the primary key or unique constraint clause followed by three periods.

The statement cannot be processed. The specified table is not altered.

User Response: Remove the erroneous rows from the table before attempting to define the primary or unique key.

sqlcode: -673

sqlstate: 23515

SQL0680N   **Too many columns are specified for a table, view, or table function.**

Explanation: The maximum number of columns permitted for each table is based on the page size of the table space and the data types of the columns. The limits for tables are:

- 4K page size allows a maximum of 500 columns
- 8K, 16K, and 32K page sizes allow a maximum of 1012 columns.

The actual number of columns for a table is determined by the following formula: Total Columns * 8 + Number of LOB Columns * 12 + Number of Datalink Columns * 28 <= row size limits for page size.

The maximum number of columns permitted for each view is 5000.

The maximum number of columns permitted for a table function is 255.

Federated system users: other data sources may have a different maximum column limit. This limit has been exceeded.

The statement cannot be processed.

User Response: Ensure that the number of table or view columns does not exceed the limit. If creating a table where a larger page size would allow the number of columns, specify a table space with the larger page size. Create separate tables or views, as required, to hold additional information beyond the limit.

Federated system 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, attribute, user-defined type or function *data-item* **contains incompatible clauses.**

Explanation: 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.
- for a CREATE or ALTER TYPE statement, *data-item* identifies the clause containing the error or gives the name of the attribute containing the error.

The statement cannot be processed.

User Response: 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***.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Remove any OLD or NEW correlation names from the REFERENCING clause or replace FOR EACH STATEMENT with FOR EACH ROW.

sqlcode: -697

sqlstate: 42899

SQL0700 - SQL0799

SQL0707N **The name** *name* **cannot be used because the specified identifier is reserved for system use.**

Explanation: The following lists reserved names:

- a table space name cannot start with 'SYS'
- a database partition group name cannot start with 'SYS' or 'IBM'
- a savepoint name cannot start with 'SYS'

User Response: Select a name that is not reserved.

sqlcode: -707

sqlstate: 42939

SQL0713N **The replacement value for** *special-register* **is invalid.**

Explanation: 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.

User Response: 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

SQL0719N **Bind error for user** *auth-id*. **Package** *package-name* **already exists.**

Explanation: An attempt is made, using the ACTION ADD option of PRECOMPILE or BIND to add a package that already exists. The combination of 'pkgschema.pgkname.pkgversion' must be unique in the SYSCAT.PACKAGES catalog view.

auth-id Authorization ID of the invoker of the BIND or PREP

package-name
Name of the package in the form 'pkgschema.pkgname.pkgversion'. If the package version is the empty string, then '.pkgversion' is omitted from the name.

The package is not created.

User Response: Possible ways to ensure that there is no attempt to add a duplicate entry include:

- Check the SYSCAT.PACKAGES catalog view for names of existing application packages. Re-invoke the PRECOMPILE or BIND with a 'pkgschema.pkgname.pkgversion' that is not in use.
- Reissue the PREP or BIND statement without the ACTION ADD option. This will replace the existing package.

sqlcode: -719

sqlstate: 42710

SQL0720N **Attempting to replace package** *pkgschema.pkgname* **WITH VERSION** *pkgversion*, **where this version already exists.**

Explanation: An attempt is made to create a version of a package that already exists using the ACTION REPLACE REPLVER option. The version specified in the REPLVER keyword is different from the version specified in the VERSION precompile option. The version specified in the VERSION precompile option already exists in the catalog. The combination of 'pkgschema.pkgname.version' must be unique in the SYSCAT.PACKAGES catalog view.

A common mistake is that the user may believe that the version that is being created is the one specified in the REPLVER keyword. This is not the case. The version specified in the REPLVER keyword is the name of the version being replaced. The version that will be created is the version that is given to the program in the VERSION option.

No package is created.

User Response: Ensure that the version being created does not already exist.

There are two approaches to solve this problem:

- Precompile the program again with a new version name and reissue the original PREP or BIND command.
- The other approach is to reissue the PREP or BIND command specifying the ACTION REPLACE clause without the REPLVER specification. In this case, the REPLACE will replace the version that matches the version specified in the VERSION option.

sqlcode: -720

sqlstate: 42710

SQL0721N   Package *pkgschema.pkgname* with consistency token = 0X*contoken* cannot be created since it is not unique.

Explanation: An attempt is made to add or replace a package with a consistency token that is not unique for that package. The combination of 'pkgschema.pkgname.contoken' already exists. This is most likely caused by using the LEVEL option of the PRECOMPILE to specify the consistency token.

The package is not created.

User Response: Check the SYSCAT.PACKAGES catalog table for names of existing application packages with the indicated consistency token. Reissue the PREP or BIND command such that the 'pkgschema.pkgname.contoken' is unique within the catalog. The following SQL statement can be used to query the catalog:

```
SELECT PKGSCHEMA,PKGNAME
FROM SYSCAT.PACKAGES
WHERE HEX(UNIQUE_ID) = 'contoken';
```

If the LEVEL option on the precompile was used, reissuing the precompile and specifying a different LEVEL value would change the consistency token. It is recommended that the LEVEL option is not used; not specifying the

LEVEL option of the precompile would ensure that the consistency token is a current timestamp value.

SQL0722N   Error performing *bind-command* for user *auth-id*. Package *package-name* does not exist.

Explanation: Bind or rebind was issued against a package that does not exist. The error would occur if ACTION REPLACE was specified and the REPLVER option specified a version that doesn't exist.

bind-command
   Type of bind command (BIND | REBIND) issued. Note that the value 'BIND' will be used for a precompile as well.

auth-id
   Authorization ID that issued the bind or rebind.

package-name
   Name of the package in the form 'pkgschema.pkgname.pkgversion'. If the package version is the empty string, then '.pkgversion' is omitted from the name.

The package was not bound or rebound.

Explanation: Check the SYSCAT.PACKAGES catalog view for the correct 'pkgschema.pkgname.pkgversion' to specify in the REPLVER option.

sqlcode: -722

sqlstate: 42704

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*

Explanation: 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.

User Response: 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 *object-name* **of type** *object-type* **would exceed the maximum level of indirect SQL cascading.**

Explanation: Cascading of indirect SQL occurs when a trigger activates another trigger (possibly through referential constraint delete rules) or a routine, containing SQL, invokes another routine. 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, or where a routine directly or indirectly invokes itself, 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 *object-type* is one of TRIGGER, FUNCTION, METHOD, or PROCEDURE.

The *object-name* specified is one of the objects that would have been activated at the seventeenth level of cascading.

User Response: Start with the objects that are activated or invoked by the statement that received this error. If any of these objects are recursive, ensure that there is some condition that prevents the object from being activated or invoked more than the limit allows. If this is not the cause of the problem, follow the chain of objects that are activated or invoked to determine the chain that exceeds the cascading limit.

sqlcode: -724

sqlstate: 54038

SQL0727N   An error occurred during implicit system action type *action-type***. Information returned for the error includes SQLCODE** *sqlcode***, SQLSTATE** *sqlstate* **and message tokens** *token-list***.**

Explanation: The processing of a statement or command has caused the database manager to implicitly perform additional processing. During this processing an error was encountered. The action attempted is shown by the *action-type*:

1   implicit rebind of a package

2   implicit prepare of a cached dynamic SQL statement

3   implicit regeneration of a view

4   This return code is reserved for use by DB2.

5   incremental bind of a static SQL statement, not bound during package bind time

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 original SQL statement or command that caused the *action-type* cannot be processed and the implicit system action was not successful.

Federated system users: You may have received this message because you dynamically prepared an SQL statement in a pass-through session and then tried to execute the statement after the session was closed.

User Response: Check the message associated with the SQLCODE of the SQL statement that failed. Follow the action suggested by that message.

For an invalid package, the REBIND command

can be used to recreate the error or to explicitly validate the package once the cause of the error has been resolved.

For a failure while regenerating a view, the name of the view that failed is recorded in the administration notification log. The failing view can be dropped or a change made to the statement or command that caused the view regeneration.

Federated system users: If the statement that failed was dynamically prepared in a pass-through session, open another pass-through session, write and prepare the statement again, and execute it while the session is still open.

sqlcode: -727

sqlstate: 56098

SQL0740N   **Routine** *routine-name* **(specific name** *specific-name***) is defined with the MODIFIES SQL DATA option, which is not valid in the context where the routine is invoked.**

Explanation: The routine *routine-name* (specific name *specific-name*) is defined with MODIFIES SQL DATA and is therefore not allowed in:

- a BEFORE trigger
- a correlated subquery
- a search-condition (such as a WHERE clause or check constraint)

User Response: Ensure that routine resolution resolved to the desired routine. Remove the routine from the failing statement or redefine the routine with an SQL access indicator other than MODIFIES SQL DATA.

sqlcode: -740

sqlstate: 51034

SQL0746N   **Routine** *routine-name* **(specific name** *specific-name***) violated nested SQL statement rules when attempting to perform operation** *operation* **on table** *table-name***.**

Explanation: The routine *routine-name* (specific name *specific-name*) attempted to perform the operation *operation* on table *table-name*. This operation conflicts with other uses of the table by either the application, or a routine invoked directly or indirectly from that application.

If the operation is "READ", then the table *table-name* is currently being written to by either the application or another routine.

If the operation is "MODIFY", then the table *table-name* is already being read from or written to by the application or another routine.

If *table-name* references an explain table and the statement that receives this error message is either a PREPARE statement or an EXECUTE IMMEDIATE statement, then a conflict occurred when inserting explain information into the explain table.

User Response: The operation may be successful if retried. Redesign either the application or the routine to avoid the conflict.

If the conflict occurred when inserting explain information for a dynamic statement, then disable explain for dynamic statements and try the PREPARE statement or EXECUTE IMMEDIATE statement again.

sqlcode: -746

sqlstate: 57053

SQL0750N   **The source table cannot be renamed because it is referenced in a view, materialized query table, trigger, SQL function, SQL method, check constraint, or referential constraint.**

Explanation: 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 materialized query tables.
- The table is referenced in one or more existing triggers. This includes triggers on the table or references in triggered SQL statements.
- The table is referenced in one or more existing SQL functions or SQL methods.
- 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.

User Response: Drop the views, materialized query table, triggers, SQL functions, SQL methods, check constraints, or referential constraints on the table before issuing the RENAME statement. For views or materialized query tables 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 SQL functions or SQL methods, query SYSCAT.FUNCDEP 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   Routine *routine-name* **(specific name** *specific-name*) **attempted to execute a statement that is not allowed.**

Explanation: The program used to implement the body of a routine is not allowed to issue any connection statements. If the routine is a function or method, COMMIT and ROLLBACK (without

the SAVEPOINT option) are also not allowed.

User Response: Remove the statement that is not allowed, then recompile the program.

sqlcode: -751

sqlstate: 38003

sqlstate: 42985

SQL0752N   Connecting to a database is not permitted within a logical unit of work when the CONNECT type 1 connection setting is in use.

Explanation: 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.

User Response:
- 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

SQL0773N   Case not found for CASE statement.

Explanation: A CASE statement without an ELSE clause was found in the routine body of an SQL routine. None of the conditions specified in the CASE statement were met.

User Response: Change the CASE statement to handle all conditions that can occur.

sqlcode: -773

sqlstate: 20000

SQL0774N **The statement cannot be executed within an ATOMIC compound SQL statement.**

Explanation: A COMMIT or ROLLBACK statement was encountered within the context of an ATOMIC compound SQL statement. These statements are not allowed in this context.

User Response: Remove the COMMIT or ROLLBACK statement or change the context so that it is not an ATOMIC compound statement.

sqlcode: -774

sqlstate: 2D522

SQL0776N **Use of cursor** *cursor-name* **is not valid.**

Explanation: Cursor *cursor-name* is specified as the cursor name on a FOR statement in an SQL procedure. The cursor cannot be specified on a CLOSE, FETCH, or OPEN statement within the FOR statement.

User Response: Remove the CLOSE, FETCH, or OPEN statement.

sqlcode: -776

sqlstate: 428D4

SQL0777N **Nested compound statements are not allowed.**

Explanation: Atomic compound statements in the routine body of an SQL procedure cannot be nested.

User Response: Ensure that the SQL procedure does not contain nested atomic compound statements.

sqlcode: -777

sqlstate: 42919

SQL0778N **End label** *label* **is not the same as the begin label.**

Explanation: Label *label* specified at the end of a FOR, IF, LOOP, REPEAT, WHILE or compound statement is not the same as the label at the

beginning of the statement. You cannot specify an end label if a begin label is not specified.

User Response: Ensure that the end label is the same as the begin label for FOR, IF, LOOP, REPEAT, WHILE, and compound statements.

sqlcode: -778

sqlstate: 428D5

SQL0779N **Label** *label* **specified on a GOTO, ITERATE, or LEAVE statement is not valid.**

Explanation: Label *label* is specified on a GOTO, ITERATE or LEAVE statement. The label is not defined or is not a valid label for the statement.

The label on an ITERATE statement must be the label for a FOR, LOOP, REPEAT, or WHILE statement.

The label on a LEAVE statement must be the label for a FOR, LOOP, REPEAT, WHILE, or compound statement.

The label on a GOTO must be defined within a scope that the GOTO statement can reach.

• If the GOTO statement is defined in a FOR statement, *label* must be defined inside the same FOR statement, excluding a nested FOR statement or nested compound statement.

• If the GOTO statement is defined in a compound statement, *label* must be defined inside the same compound statement, excluding a nested FOR statement or nested compound statement.

• If the GOTO statement is defined in a handler, *label* must be defined in the same handler, following the other scope rules.

• If the GOTO statement is defined outside of a handler, *label* must not be defined within a handler.

User Response: Specify a valid label on the GOTO, ITERATE, or LEAVE statement.

sqlcode: -779

sqlstate: 42736

SQL0780N **UNDO is specified for a handler when ATOMIC is not specified for the compound statement.**

Explanation: UNDO is specified for a handler in a compound statement in an SQL procedure. UNDO cannot be specified unless the compound statement is ATOMIC.

User Response: Either specify that the compound statement is ATOMIC or specify EXIT or CONTINUE on the handler.

sqlcode: -780

sqlstate: 428D6

SQL0781N **Condition** *condition* **specified in a handler is not defined.**

Explanation: Condition *condition* specified in a handler in an SQL procedure is not defined.

User Response: Define the condition using the DECLARE CONDITION statement or remove the condition from the handler.

sqlcode: -781

sqlstate: 42737

SQL0782N **A condition or SQLSTATE value specified in a handler is not valid.**

Explanation: A condition or SQLSTATE value specified in a handler in an SQL is not valid for one of the following reasons:

- The condition or SQLSTATE value has already been specified by another handler in the same scope.
- The condition or SQLSTATE value was specified in the same handler as SQLEXCEPTION, SQLWARNING, or NOT FOUND.

User Response: Remove the condition or SQLSTATE value from the handler.

sqlcode: -782

sqlstate: 428D7

SQL0783N **A duplicate column name or unnamed column was specified in a DECLARE CURSOR statement of a FOR statement.**

Explanation: The select list in the FOR statement must contain unique column names. The select list specified contains either duplicate column names or unnamed expressions.

User Response: Specify unique column names in the select list specified in the FOR statement.

sqlcode: -783

sqlstate: 42738

SQL0785N **The declaration or use of the SQLSTATE or SQLCODE variable is not valid.**

Explanation: SQLSTATE or SQLCODE was used as a variable in the routine body of an SQL routine, but is not valid for one of the following reasons:

- SQLSTATE is not declared as CHAR(5)
- SQLCODE is not declared as INTEGER
- The variable is assigned the NULL value.

The statement cannot be processed.

User Response: Declare the SQLSTATE variable as CHAR(5) and the SQLCODE variable as INTEGER. Set the variable to a valid value.

sqlcode: -785

sqlstate: 428D8

SQL0787N **RESIGNAL statement not within a handler.**

Explanation: The RESIGNAL statement can only be used inside condition handlers.

User Response: Remove the RESIGNAL statement or use a SIGNAL statement instead.

sqlcode: -787

sqlstate: 0K000

SQL0789N **The data type for parameter or variable** *name* **is not supported in the SQL routine.**

Explanation: The SQL routine (function, method, or procedure) does not support variables or parameters of DATALINK, REFERENCE, DISTINCT, STRUCTURED, or LOB data types.

User Response: Do not use SQL variables or parameters of DATALINK, REFERENCE, DISTINCT, STRUCTURED, or LOB data types in the SQL routine definition. Specify a different data type for the parameter or variable *name*.

sqlcode: -789

sqlstate: 429BB

SQL0797N **The trigger** *trigger-name* **is defined with an unsupported triggered SQL statement.**

Explanation: The trigger must be defined with a triggered SQL statement that can only include statements from the following list.

- A trigger can include the following control statements:
  - dynamic compound statement
  - FOR statement
  - GET DIAGNOSTICS statement
  - IF statement
  - ITERATE statement
  - LEAVE statement
  - SIGNAL statement
  - WHILE statement
- A BEFORE trigger can also include the following triggered SQL statements:
  - a fullselect
  - an assignment statement
- An AFTER trigger can also include the following triggered SQL statements:
  - an INSERT statement

- a searched UPDATE statement
  - a searched DELETE statement
  - a fullselect
  - an assignment statement (excluding assignment of transition variables)
- An INSTEAD OF trigger can also include the following triggered SQL statements:
  - an INSERT statement
  - a searched UPDATE statement
  - a searched DELETE statement
  - a fullselect
  - an assignment statement (excluding assignment of transition variables)

In some cases the *trigger-name* is not available for use in the message.

User Response: Check the triggered SQL statements in the trigger for any statement that does not match the above list and remove it.

sqlcode: -797

sqlstate: 42987

SQL0798N **A value cannot be specified for column** *column-name* **which is defined as GENERATED ALWAYS.**

Explanation: When inserting or updating a row in a table, a value was specified for the GENERATED ALWAYS column *column-name*. GENERATED ALWAYS columns should not be specified in the column-list for an INSERT or in the SET clause for an update unless the keyword DEFAULT is specified.

The INSERT or UPDATE is not performed.

User Response: Remove the GENERATED ALWAYS column from the column-list or SET clause, or specify DEFAULT as the column value.

sqlcode: -798

sqlstate: 428C9

SQL0800 - SQL0899

SQL0801N   **Division by zero was attempted.**

Explanation: 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.

User Response: 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.

Federated system 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.**

Explanation: 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.

User Response: 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.

This error may also be returned when column functions cannot handle the value that is returned by the SQL statement. For example, issuing a SELECT COUNT statement against a table that has more rows than defined by the MAX_LONGINT_INT SQL limit will cause an arithmetic overflow error. Consider using the COUNT_BIG column function for tables with more than 2 147 483 647 rows.

Federated system 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 statement, UPDATE statement, or foreign key update caused by a DELETE statement are not valid because the primary key, unique constraint or unique index identified by** *index-id* **constrains table** *table-name* **from having duplicate rows for those columns.**

Explanation: The INSERT or UPDATE object table *table-name* is constrained by one or more UNIQUE indexes to have unique values in certain columns or groups of columns. Alternatively, a DELETE statement on a parent table caused the update of a foreign key in a dependent table *table-name* that is constrained by one or more UNIQUE indexes. Unique indexes might support primary keys or unique constraints defined on a table. The statement cannot be processed because completing the requested INSERT, UPDATE or DELETE statement would result in duplicate column values.

Alternatively, if a view is the object of the INSERT or UPDATE statement, it is the table *table-name* on which the view is defined that is constrained.

If *index-id* is an integer value, the index name can be obtained from SYSCAT.INDEXES by issuing the following query:

```
SELECT INDNAME, INDSCHEMA
  FROM SYSCAT.INDEXES
  WHERE IID = <index-id>
  AND TABSCHEMA = 'schema'
  AND TABNAME = 'table'
```

where *schema* represents the schema portion of *table-name* and *table* represents the table name portion of *table-name*.

The statement cannot be processed. The table remains unchanged.

User Response: Examine the definition for the index identified by *index-id*.

For an UPDATE statement, ensure that the specified operation is not itself inconsistent with the uniqueness constraint. 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.

For a DELETE statement, examine the identified dependent table for unique constraints on foreign keys that are defined with the rule ON DELETE SET NULL. This table has a foreign key column included in the identified unique index that cannot be set to null since there is already a null in the column for that table.

Federated system users: isolate the problem to the data source failing the request (refer to the Troubleshooting 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 parameters for the current request are not valid. Reason code** *reason-code*. **If a host variable or SQLVAR in the SQLDA is invalid then: host variable/SQLVAR number =** *var-number*, **SQLTYPE =** *sqltype*, **SQLLEN =** *sqllen*, **host variable/SQLVAR type =** *input-or-output*.

Explanation: An error occurred while processing the current request.

- The call parameter list, which is created by the precompiler, may not be correct if the application programmer has modified the output of the precompiler or over-written the call parameter list in another way.

- The SQLDA or host variable(s) in the SQL statement are invalid.

- The request being made is not supported or is out of context.

Reason codes are interpreted as follows:

100 The request being made is not supported or is out of context.

101 SQLDA.SQLN is less than SQLDA.SQLD

102 SQLVAR.SQLTYPE is invalid.

Federated system users: A data type that you specified is not supported by either the federated server or the data source that you want to access.

103 The length specified in SQLVAR.SQLLEN or SQLVAR2.SQLLONGLEN is incorrect for the SQL type given in SQLVAR.SQLTYPE.

104 Doubling of the SQLVARs is expected, but the SQLDOUBLED field of SQLDA.SQLDAID is not set to '2'. This may be required because of a large object type or a structured type.

105 A double-byte character Large Object has an odd value indicated by the

SQLVAR2.SQLDATALEN pointer, which is always in terms of bytes, even for DBCLOBs.

106	The SQLDATA pointer is invalid or points to insufficient storage.
107	The SQLIND pointer is invalid or points to insufficient storage.
108	The SQLDATALEN pointer is invalid or points to insufficient storage.
109	A specific number of host variables/SQLVARS is expected for the current SQL statement.
110	The LOB locator is not associated with a LOB of a compatible type.
111	A LOB is indicated by the SQLTYPE of the SQLVAR, but the second SQLVAR is null.
112	The SQLDATATYPE NAME field is not valid. It does not conform to the format for identifying an existing user-defined type in the database. The format for identifying an existing user-defined type is as follows: 8 bytes, followed by a period, followed by 18 bytes.
113	The SQLFLAG4 field is not valid. If a structured type is specified the value must be X'12'. If a reference type is specified the value must be X'01'. Otherwise the value must be X'00'.

In SQL statements with host variables, use the host variable number to count in from the beginning of the statement (or substatement, in the case of compound SQL) to locate the invalid host variable. For statements using an SQLDA, the SQLVAR number is used to locate the invalid SQLVAR. For an input SQLDA, count only input host variables or SQLVARs; similarly for output. Note that this number is based at 1.

The statement cannot be processed.

User Response: Examine the application program for any errors noted. Note that the programmer should not attempt to modify the precompiler output.

Federated system users: If you received reason code 102, specify a supported data type and resubmit the program.

sqlcode: -804

sqlstate: 07002

SQL0805N   **Package** *package-name* **was not found.**

Explanation: The statement cannot complete because the necessary package was not found in the catalog.

The *package-name* is either in the form:
- 'pkgschema.pkgname 0Xcontoken', where the consistency token is given in hexadecimal.
- pkgschema.pkgname.pkgversion', if the package version is the empty string then '.pkgversion' is omitted from the name.

Possible causes for this message (SQLCODE) are:
- The package was not bound or it was dropped.
- If attempting to run a DB2 utility or CLI application, the DB2 utilities may need to be rebound to the database.

Note that when version-ids are in use for the given package-schema.package-name, there may be packages defined with the same package schema and package name, but the correct package is not found since the existing packages do not match the requested version or consistency token. A package must match all three parts of the package-name. When multiple versions are being used, additional causes for this messages are:
- the version of the application being executed was precompiled, compiled, and linked, but was not bound, or it was bound but that version of package was subsequently dropped.
- the application was precompiled and bound, but not compiled and/or linked, so the application being executed is not up to date.
- the package was bound from a bind file generated by a different precompile of the source file than the precompile that produced

the modified source file that was compiled and linked into the application executable.

- a new application was bound with the same name (and version) of an existing package, thus replacing the existing package. If the application associated with the replaced package is run, this error would result.

In all these cases, the consistency token of the request does not match the consistency token of the existing version and thus the package is considered to be not found.

The statement cannot be processed.

User Response: 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. Ensure that the application or object module being executed is the compiled and linked modified source code associated with the precompile and bind that generated the package.

The following SQL statements can be used to query the catalog to determine if there are different versions of the package.

```
SELECT PKGSCHEMA, PKGNAME,
       PKGVERSION, UNIQUE_ID
FROM SYSCAT.PACKAGES
WHERE PKGSCHEMA = 'pkgschema'
      and PKGNAME='pkgname'.
```

Note that the UNIQUE_ID column corresponds to the consistency token.

If the DB2 utility programs need to be rebound to the database, the database administrator can accomplish this by issuing one of the following CLP command from the bnd subdirectory of the instance, while connected to the database:

- ″DB2 bind @db2ubind.lst blocking all grant public″ for the DB2 utilities.
- ″DB2 bind @db2cli.lst blocking all grant public″ for CLI.

Federated system users: ensure that the packages required for the federated server are bound at the applicable data sources. Refer to the

Federated Systems Guide for more information about binding packages to data sources.

sqlcode: -805

sqlstate: 51002

SQL0808N   **The CONNECT statement semantics are not consistent with those of other existing connections.**

Explanation: 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.

User Response: 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

SQL0811N   **The result of a scalar fullselect, SELECT INTO statement, or VALUES INTO statement is more than one row.**

Explanation: 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.

Federated system users: this situation can be detected by federated server or by the data source.

The statement cannot be processed.

User Response: 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.

Federated system users: isolate the problem to the data source failing the request (refer to the *Troubleshooting 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

SQL0817N   The SQL statement cannot be executed because the statement will result in a prohibited update operation.

Explanation: The application attempted to execute an SQL statement that would result in updates to user data or to the subsystem catalog. This is prohibited for one of the following reasons:

- The application is running as an IMS inquiry-only transaction.
- The application is an IMS or CICS application that is attempting to update data at a remote DBMS that does not support two-phase commit.
- The application is attempting to update data at multiple locations and one of the locations does not support two-phase commit.

These SQL statements include INSERT, UPDATE, DELETE, CREATE, ALTER, DROP, GRANT, and REVOKE.

The statement cannot be executed.

User Response: If the application is running as an IMS inquiry-only transaction, see your IMS system programmer about changing the inquiry-only status of the transaction under which your application is running.

If the IMS or CICS application is attempting a remote update, either the application must be changed to run as a local application on the server DBMS, or the server DBMS must be

upgraded to support two-phase commit.

If the application is attempting to update data at multiple locations, either the application must be changed, or all DBMSs involved must be upgraded to support two-phase commit.

sqlcode: -817

sqlstate: 25000

SQL0818N   A timestamp conflict occurred.

Explanation: 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:

- The application was precompiled, compiled, and linked, but was not bound.
- The application was precompiled and bound, but not compiled and/or linked, so the application being executed is not up to date.
- The package was bound from a bind file generated by a different precompile of the source file than the precompile that produced the modified source file that was compiled and linked into the application executable.
- A new application was bound of the same name and existing package, thus replacing the existing package. If the application associated with the replaced package is run, this error would result.

Federated system users: in addition to the causes listed previously, the problem can also be due to not binding the required packages at all applicable data sources.

The statement cannot be processed.

User Response: 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.

Federated system users: in addition to the previously listed actions, ensure that the packages required for the federated server are bound at the applicable data sources. Refer to the Federated Systems Guide for more information on binding packages to data sources.

sqlcode: -818

sqlstate: 51003

SQL0822N **The SQLDA contains a data address or indicator variable address which is not valid.**

Explanation: The application program placed an address that is not valid in the SQLDA.

The statement cannot be processed.

User Response: 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.**

Explanation: The number of items returned in the SELECT list exceeds the allowable maximum. The maximum for SELECT lists (other than on common table expressions) is 1012. The maximum for SELECT lists in common table expressions is 5000.

The statement cannot be processed.

User Response: 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 **A connection to server** *server-name* **already exists.**

Explanation: SQLRULES(STD) is in effect and a CONNECT statement identifies an existing SQL connection.

User Response: 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 server name does not specify an existing connection.**

Explanation: A statement, command, or API specified a server name that does not identify an existing SQL connection of the application process.

This may have occurred using:

- a SET CONNECTION statement
- a RELEASE statement
- a DISCONNECT statement
- SET or QUERY CLIENT INFORMATION

User Response: 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 the request for that connection.

Correct the error in the application and try again.

sqlcode: -843

sqlstate: 08003

SQL0845N **A PREVVAL expression cannot be used before the NEXTVAL expression generates a value in the current session for sequence** *sequence-name***.**

Explanation: A PREVVAL expression specified sequence *sequence-name*, but a value has not yet been generated for this sequence. A NEXTVAL expression must be issued in this session to generate a value for this sequence before a PREVVAL expression for the sequence can be issued.

User Response: Issue at least one NEXTVAL expression for a sequence before issuing any PREVVAL expression for the same sequence in a session.

sqlcode: -845

sqlstate: 51035

SQL0846N **Invalid specification of identity column or sequence object** *object-type object-name***. Reason code = ** *reason-code***.**

Explanation: For an identity column or sequence object, the specification of an attribute in a CREATE or ALTER statement may be invalid for one of the following reasons:

1. The underlying data type of the identity column or sequence object is not supported. Identity columns and sequence objects support the following data types: SMALLINT, INTEGER, BIGINT and DECIMAL (or NUMERIC) with a scale of zero.

2. The value for START WITH, INCREMENT BY, MINVALUE or MAXVALUE is outside of the range for the data type of the identity column or sequence object.

3. MINVALUE must be less than or equal to MAXVALUE.

4. An invalid value was specified for CACHE. The value must be an INTEGER with a minimum value of 2.

User Response: Correct the syntax and resubmit the statement.

sqlcode: -846

sqlstate: 42815

SQL0859N **Access to the Transaction Manager Database failed with SQLCODE** *SQLCODE***.**

Explanation: 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.

User Response: 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 DB2 Connect.

sqlcode: -859

sqlstate: 08502

SQL0863W   A successful connection was made, but only single byte characters should be used.

Explanation: 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).

Federated system users: possible causes are:

- The federated 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 federated system supports only single-byte characters.

User Response: Do not submit SQL statements or commands that use characters which are not common between the application and database codepages.

Federated system users: do not submit SQL statements or commands that use characters which are not common between the client system, the federated system, and the data source.

sqlcode: +863

sqlstate: 01539

SQL0865N   Invalid tm_database value.

Explanation: 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 DB2 Connect).

The statement cannot be executed.

User Response:

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*

Explanation: 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.

User Response: 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.**

Explanation: A current or dormant connection to a server exists, and an attempt to CONNECT to this server using the USER/USING clause is invalid.

User Response: 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

SQL0880N **SAVEPOINT** *savepoint-name* **does not exist or is invalid in this context.**

Explanation: An error occurred issuing a RELEASE or ROLLBACK TO SAVEPOINT *savepoint-name* statement. Either a savepoint with that name does not exist, or it was established outside the current atomic execution context.

The statement cannot be processed.

User Response: Correct the name of the savepoint in the statement and reissue the statement.

sqlcode: -880

sqlstate: 3B001

SQL0881N **A SAVEPOINT with name** *savepoint-name* **already exists, but this savepoint name cannot be reused.**

Explanation: The name *savepoint-name* has already been used in a SAVEPOINT statement. The savepoint name cannot be reused because at least one of the SAVEPOINT statements that used this name also specified the UNIQUE keyword, which asserts that the name must be unique.

The statement cannot be processed. The new savepoint is not set. The old savepoint with the same name still exists.

User Response: Choose another name for this savepoint and reissue the SAVEPOINT statement. If this existing savepoint name must be reused, issue the RELEASE SAVEPOINT statement to free the existing savepoint. However, be aware that the RELEASE SAVEPOINT statement also releases any savepoints that were established in the transaction since the specified savepoint was established. For more information, refer to the SQL Reference.

sqlcode: -881

sqlstate: 3B501

SQL0882N **Savepoint does not exist.**

Explanation: An error occurred issuing a ROLLBACK TO SAVEPOINT statement. Issuing a ROLLBACK TO SAVEPOINT without specifying a specific savepoint name is not allowed when there are no existing savepoints.

The statement cannot be processed.

User Response: Issue a different statement, or try rolling back the entire transaction with the ROLLBACK statement.

sqlcode: -882

sqlstate: 3B502

SQL0900 - SQL0999

SQL0900N **The application state is in error. A database connection does not exist.**

Explanation: 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.

User Response: Reestablish a current connection be either switching to an existing dormant connection (using 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*.**)**

Explanation: 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).

User Response: Record the message number (SQLCODE) and reason *reason*.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Then contact your technical service representative with the following information:

- Problem description
- SQLCODE
- reason *reason*
- SQLCA contents if possible
- Trace file if possible.

Federated system users: isolate the problem to the data source failing the request (refer to the

Troubleshooting 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.**

Explanation: A system error occurred.

User Response: 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. 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.

Federated system users: isolate the problem to the data source failing the request (refer to the Troubleshooting 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*

Explanation: 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.

Federated system users: if the database you connected to is a federated server 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 or nodes participating in the unit of work was accessed, but unable to prepare to commit.

Federated system users: if the database you connected to is a federated server database where nicknames are used, one of the data sources required for a nickname is unable to prepare to commit.

03   A DB2 Data Links Manager participating in the unit of work was unable to prepare to commit.

04   One or more declared temporary tables are in an inconsistent state.

05   An unexpected error occurred. Check the administration notification log for details.

User Response: 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

SQL0904N   Unsuccessful execution caused by an unavailable resource. Reason code: *reason-code***, type of resource:** *resource-type***, and resource name:** *resource-name***.**

Explanation: The SQL statement could not be executed because resource *resource-name* of type *resource-type* was not available at the time for the reason indicated by *reason-code*. Refer to the Problem Determination documentation of DB2 for MVS for an explanation of resource type codes.

User Response: Verify the identity of the resource that was not available. To determine why the resource was unavailable, refer to the specified *reason-code*.

sqlcode: -904

sqlstate: 57011

SQL0905N   Unsuccessful execution due to resource limit being exceeded. Resource name = *resource-name***, limit =** *limit-amount1* **CPU seconds (***limit-amount2* **service units) derived from** *limit-source***.**

Explanation: The execution of the SQL statement was terminated because a resource limit was exceeded.

The name of the resource whose limit was exceeded is *resource-name*. It is also the name of the column in the resource limit specification table from which the limit was derived. The limit that was exceeded in CPU seconds is *limit-amount1* and in service units it is *limit-amount2*. The *resource-name* may be ASUTIME, which is the number of CPU seconds permitted for each SQL statement. The maximum number of CPU seconds permitted is *limit-amount1*. The maximum number in service units is *limit-amount2*.

The source used to derive the limit-amount is *limit-source* and is either the name of a resource limit specification table or a 'system parameter'. If the source is a system parameter, the resource limit specification table did not contain an

applicable entry or an error occurred while accessing the table. In either case, the limit is obtained from an install (system) parameter.

User Response: Determine why this SQL statement took so long and take appropriate action. Consider simplifying the SQL statement, restructuring tables and indexes, or contacting the installation group responsible for maintaining the resource limit specification tables.

An application program that receives this return code can execute additional SQL statements.

sqlcode: -905

sqlstate: 57014

SQL0906N   **The SQL statement cannot be executed because this function is disabled due to a prior error.**

Explanation: Execution of the SQL statement failed because the requested function had been disabled by a prior error. This situation can arise if the application program has intercepted an abend (for instance, by an ON ERROR condition in a PL/I program) and continued to execute SQL statements. This situation may also arise if a DB2 CICS transaction encountered a create thread error yet continued to issue SQL requests without issuing a SYNCPOINT ROLLBACK first.

User Response: In general, an application program should terminate upon receipt of this return code. All subsequent attempts by the application to execute other SQL statements will also fail with the same return code. In the case of a DB2 CICS transaction, if the SQLERRP field in the SQLCA contains the module name DSNCEXT1, the transaction may issue a SYNCPOINT ROLLBACK and continue processing. If the transactions chooses to ROLLBACK and continue processing, it must be capable of correcting the situation that caused the create thread error to occur originally.

sqlcode: -906

sqlstate: 24514, 51005, 58023

SQL0908N   *bind-type* **error using** *auth-id* **authority BIND, REBIND, or AUTO_REBIND operation is not allowed.**

Explanation: For BIND and REBIND, the indicated authorization ID is not allowed to perform the indicated *bind-type* against a plan or package. An entry in the resource limit specification table (RLST) prohibits binding and rebinding by this authorization ID, or all authorization IDs. For AUTO-REBIND, the system parameter controlling AUTO-REBIND operations is set to disallow AUTO-REBIND.

bind-type
            Type of bind operation (BIND, REBIND or AUTO-REBIND).

auth-id Authorization ID of the invoker of the BIND subcommand or primary authorization ID of the invoker of the plan for AUTO-REBIND operations.

User Response: If the indicated authorization id should be allowed to bind, change the entry in the active RLST table. If AUTO-REBIND operations are disabled, rebind the package before reexecuting the package.

sqlcode: -908

sqlstate: 23510

SQL0909N   **The object has been deleted.**

Explanation: The application program has either (1) dropped a table and then attempted to access it, or (2) dropped an index and then tried to access its object table using that index.

User Response: The logic of the application program must be corrected such that it does not attempt to access or use an object after it has been dropped.

Dropping indexes within an application program is especially hazardous, because there is no way of determining whether or not the plan that has been generated for the application (by BIND or REBIND) actually uses a particular index for access to its object table.

sqlcode: -909

sqlstate: 57007

SQL0910N **The SQL statement cannot access an object on which a modification is pending.**

Explanation: The application program attempted to access an object within the same unit of work in which either:

- The application program issued a DROP against the object or a related object (e.g. an index for a table)

- The application program issued a statement against the object that added or dropped a constraint.

- The application program issued a DROP TRIGGER or CREATE TRIGGER statement that affected the object, either directly or indirectly.

- The application program issued a ROLLBACK TO SAVEPOINT statement, which placed the object in the modification pending state.

- The application program issued a statement that caused all rows of a NOT LOGGED declared temporary table to be deleted.

Federated system users: in addition to the previously listed causes, other data source specific restrictions may exist that prevent access to the object.

The SQL statement cannot be processed.

User Response: Modify the application program so there is no attempt to access an object within the same unit of work in which the modification was made. In general, it is better to isolate Data Definition Language (DDL) statements in a different unit of work than Data Manipulation Language (DML) statements that access the same objects.

The unit of work must be committed or rolled back before the statement that failed can be processed successfully. If the committed modification caused an object to be dropped, then the object may have to be recreated in order for the failing SQL statement to be processed successfully.

If the object is modified within a SAVEPOINT, modify the application program so that it does not attempt to access the object after the application issues a ROLLBACK TO SAVEPOINT statement. Any cursor that accesses the modified object and is open at the time of the ROLLBACK TO SAVEPOINT will not be accessible. Modify the application so that it closes the cursor.

If an insert, delete or update statement involving a NOT LOGGED declared temporary table fails, then all rows in that table are deleted. Any cursor that is open against that declared temporary table at the time of the failure will not be accessible and should be closed by the application.

Federated system users: if the previous action does not solve the problem, isolate the request (refer to the Troubleshooting Guide to determine which data source is failing to process the SQL statement) and determine what additional constraints exist on the data source that are preventing access to the object. Ensure that the application is not violating any of those constraints.

sqlcode: -910

sqlstate: 57007

SQL0911N **The current transaction has been rolled back because of a deadlock or timeout. Reason code** *reason-code*.

Explanation: The current unit of work was involved in an unresolved contention for use of an object and had to be rolled back.

The reason codes are as follows:

2 transaction rolled back due to deadlock.

68 transaction rolled back due to lock timeout.

72 transaction rolled back due to an error concerning a DB2 Data Links Manager involved in the transaction.

Note: The changes associated with the unit of work must be entered again.

The application is rolled back to the previous COMMIT.

User Response: To help avoid deadlock or lock timeout, issue frequent COMMIT operations, if possible, for a long-running application, or for an application likely to encounter a deadlock.

Federated system users: the deadlock can occur at the federated server or at the data source. There is no mechanism to detect deadlocks that span data sources and potentially the federated system. It is possible to identify the data source failing the request (refer to the problem determination guide to determine which data source is failing to process the SQL statement).

Deadlocks are often normal or expected while processing certain combinations of SQL statements. It is recommended that you design applications to avoid deadlocks to the extent possible.

sqlcode: -911

sqlstate: 40001

SQL0912N **The maximum number of lock requests has been reached for the database.**

Explanation: The maximum number of locks for the database was reached because insufficient memory was allocated to the lock list.

The statement cannot be processed.

User Response: The application should submit a COMMIT or ROLLBACK statement before submitting other SQL statements. Consider increasing the database configuration parameter (*locklist*) to allow more lock list space.

sqlcode: -912

sqlstate: 57011

SQL0913N **Unsuccessful execution caused by deadlock or timeout. Reason code** *reason-code***.**

Explanation: 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 branch failed due to deadlock.

68 transaction branch failed due to lock timeout.

72 transaction rolled back due to an error concerning a DB2 Data Links Manager involved in the transaction.

80 statement failed due to timeout.

User Response:

- 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 committed 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

SQL0917N **Remote bind from a DRDA application requester failed.**

Explanation: An error has occurred during a remote bind from a DRDA application requester. This error can be issued during bind or commit processing.

User Response: This problem is usually caused by an SQL statement which did not get bound because of an error. The user should consult the diagnostics facility at the DRDA application requester to determine which statement is causing the error and correct it.

sqlcode: -917

sqlstate: 42969

SQL0918N   Application must execute a rollback.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Request data only from server workstations.

sqlcode: -920

sqlstate: 57019

SQL0925N   SQL COMMIT invalid for application execution environment.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response:

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.**

Explanation: 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.

User Response: 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.

- If the error occurs on a statement which referenced a routine (UDF, stored procedure or method), it may be that the storage required to contain the arguments and return values was more than the memory available in the system. This might occur if the routine is defined with BLOB, CLOB, DBCLOB parameters or return values, having a large size (2GB for example).

If the possible solutions listed above do not solve the problem, it may be necessary to consider changing the definition of the routine.

sqlcode: -930

sqlstate: 57011

SQL0931C   **Operating system file table overflow occurred. Subsequent SQL statements cannot be processed.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Close any cursors required and resubmit the statement.

sqlcode: -950

sqlstate: 55006

SQL0951N **The object** *object-name* **of type** *object-type* **cannot be altered because it is currently in use by the same application process.**

Explanation: An ALTER statement or SET INTEGRITY statement for an object cannot be issued when it is either locked or in use.

The statement cannot be processed. The object is not altered.

User Response: Close any cursors that depend either directly or indirectly on the object *object-name* and resubmit the statement.

sqlcode: -951

sqlstate: 55007

SQL0952N **Processing was cancelled due to an interrupt.**

Explanation: 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.

Federated system users: this situation can also be detected by the data source.

User Response: Continue the application.

If installing the sample database, drop it and install the sample database again.

sqlcode: -952

sqlstate: 57014

SQL0954C **Not enough storage is available in the application heap to process the statement.**

Explanation: All available memory for the application has been used.

The statement cannot be processed.

User Response: 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. Reason code =** *reason-code***.**

Explanation: Insufficient virtual memory is available to the database agent for sort processing, as indicated by the reason code:

1 Insufficient private process memory.

2 Insufficient shared memory in the database-wide shared memory area designated for sort processing.

The statement cannot be processed but other SQL statements may be processed.

User Response: One or more of the following:

• Decrease the value of the sort heap parameter (*sortheap*) in the corresponding database configuration file.

• For reason code 1, increase the private virtual memory available, if possible. For example, on UNIX systems you can use the ulimit command to increase the maximum size of the data area for a process.

• For reason code 2, increase the size of the database-wide shared memory area designated for sort processing. To increase the size of this area without affecting the sortheap threshold for private sorts, increase the value of the SHEAPTHRES_SHR database configuration parameter.

To increase both the size of the database-wide shared memory area designated for sort processing as well as the sortheap threshold for private sorting, increase the value of the SHEAPTHRES database manager configuration parameter and set SHEAPTHRES_SHR to 0

sqlcode: -955

sqlstate: 57011

SQL0956C   **Not enough storage is available in the database heap to process the statement.**

Explanation: All available memory for the database has been used.

The statement cannot be processed.

User Response: 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.

To change dbheap, enter a command similar to the following, which sets *dbheap* to size 2400 for database *sample*:

```
db2 UPDATE DB CFG FOR sample
    USING DBHEAP 2400
```

Disconnect from the database to have the change take effect.

sqlcode: -956

sqlstate: 57011

SQL0958C   **The maximum number of open files has been reached.**

Explanation: The maximum number of file handles available to the database has been reached.

The statement cannot be processed.

User Response: 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.**

Explanation: All available memory in the server communication heap has been used.

The command or statement cannot be processed.

User Response: 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.**

Explanation: The maximum number of database files has been reached.

The statement cannot be processed.

User Response: 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

SQL0964C   **The transaction log for the database is full.**

Explanation: 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.

User Response: 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** *SQLCODE* **in the message file on this workstation. The warning was returned from module** *name* **with original tokens** *token-list*.

Explanation: Database server returned code *SQLCODE* to your application. The warning code does not correspond to a message in the DB2 message file on this workstation.

User Response: Refer to the documentation for

your database server to determine the cause for specified *SQLCODE*.

SQL0966N **The error mapping file** *name* **specified in the Database Connection Services directory cannot be found or cannot be opened.**

Explanation: 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.

User Response: 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.**

Explanation: An error occurred while the program was reading the error mapping file.

The error mapping data could not be retrieved.

User Response: Correct all syntactic errors in the error mapping file.

sqlcode: -967

sqlstate: 55031

SQL0968C **The file system is full.**

Explanation: One of the file systems containing the database is full. This file system may contain the database directory, the database log files, or a table space container.

The statement cannot be processed.

User Response: 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*.

Explanation: The Database server returned SQLCODE *error* to your application. The error code does not correspond to a message in the DB2 message file on this workstation.

User Response: Refer to the documentation for your database server to determine the cause for the specified SQLCODE. Take action specified in the documentation for the database server to fix this problem.

Federated system users: isolate the problem to the data source failing the request (refer to the Troubleshooting Guide to determine which data source is failing to process the SQL statement). Find the *error* in the applicable manual at the data source. If the problem 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.**

Explanation: A file used by the database is marked read-only or does not exist. The database requires write access to this file.

The statement cannot be processed.

User Response: Terminate the application on receipt of this message (SQLCODE). Ensure that all database files are allowed both read and write access. Check for unnecessary blank spaces within the specified filename.

sqlcode: -970

sqlstate: 55009

SQL0972N   **The database drive does not contain the correct diskette.**

Explanation: The diskette in the drive is not the database diskette.

The statement cannot be processed.

User Response: 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   **Not enough storage is available in the** *heap-name* **heap to process the statement.**

Explanation: All available memory for this heap has been used. The statement cannot be processed.

User Response: Terminate the application on receipt of this message (SQLCODE). Increase the configuration parameter for *heap-name* to increase the heap size by issuing the following command:

```
db2 update db cfg
for db-name
using heap-name heap-size
```

For APPGROUP_SHARE_HEAP, the following three database configuration parameters control its size and usage: APPGROUP_MEM_SZ, GROUPHEAP_RATIO, and APP_CTL_HEAP_SZ. The number of applications in one application group is calculated by: APPGROUP_MEM_SZ / APP_CTL_HEAP_SZ. The application group

shared heap size is calculated by: APPGROUP_MEM_SZ * GROUPHEAP_RATIO / 100.

sqlcode: -973

sqlstate: 57011

SQL0974N   The drive containing the database is locked.

Explanation: The system reported that the drive containing the database is locked.

The statement cannot be processed.

User Response: 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*.

Explanation: 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.

User Response: 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.

Explanation: The drive door for the database is open.

The statement cannot be processed.

User Response: Close the drive door and retry the operation.

sqlcode: -976

sqlstate: 57021

SQL0977N   COMMIT state unknown.

Explanation: 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.

User Response: 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.

Explanation: A write operation was attempted to the database but the database diskette is write protected.

The statement cannot be processed.

User Response: 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*.

Explanation: An application was connected to multiple databases, and a COMMIT was issued which failed for one or more of these connections.

Federated system users: if one of the failed connections is a federated server database where nicknames are used, then a commit against one of the data sources required for a nickname has failed.

User Response: 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.**

Explanation: 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.

User Response: Record all error information from the SQLCA, if possible. Terminate all applications using the database. Determine if the error is a hardware error. 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.**

Explanation: 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.

User Response: Terminate the application on receipt of this message (SQLCODE).

sqlcode: -982

sqlstate: 58004

SQL0983N   **The transaction log does not belong to the current database.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

The statement cannot be processed.

User Response: 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. Contact your technical service representative with the following information:

Information required:

- Problem description
- SQLCODE
- SQLCA contents if possible
- Trace file if possible.

Federated system users: isolate the problem to the data source failing the request (refer to the *Troubleshooting 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.**

Explanation: The system is unable to recover from an I/O error on a catalog file.

The system cannot process any statements using the database.

User Response: 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.**

Explanation: The data in the table is no longer valid.

The system cannot process any statements using the table.

User Response: 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

SQL0987C **The application control shared memory set cannot be allocated.**

Explanation: The application control shared memory set cannot be allocated. This error is caused by insufficient memory resources for either 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.

User Response: One or more of the following:
- Ensure that sufficient memory resources are available to satisfy the database manager requirements and those of the other programs running on the system.
- Reduce the database manager memory requirement for this memory set by reducing the database configuration parameter, app_ctl_heap_sz, which affects it.
- Reduce one or more of the database configuration parameters dbheap, util_heap_sz, and buffpage. Refer to the *Administration Guide* for the parameters affecting the amount of database global memory allocated.
- Reduce the database manager configuration parameter, sheapthres, if intra_parallel is set to yes, or set intra_parallel to no.
- Where appropriate, stop other programs using the system.

sqlcode: -987

sqlstate: 57011

SQL0990C   An index error occurred. Reorganize the table.

Explanation: An index has had considerable activity that used all the free space for indexes.

Federated system users: this situation can also be detected by the data source.

The statement cannot be processed.

User Response: Commit your work and retry the command. If the error continues, roll back your work. If errors still persist, reorganize the table, if possible.

Federated system users: isolate the problem to the data source failing the request (refer to the Troubleshooting 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.

Explanation: 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.

User Response: Use only programs that are precompiled with a compatible release level of the database manager.

sqlcode: -992

sqlstate: 51008

SQL0993W   The new path to the log (newlogpath) in the database configuration file is not valid.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: There are corrupted internal database pages or internal logic errors in a table space.

User Response: 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***.**

Explanation: 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

User Response: No action required.

SQL0998N   Error occurred during transaction or heuristic processing. Reason Code = *reason-code* **Subcode =** *subcode***.**

Explanation: 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 federated 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 federated 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.
  - 10 - invalide exe_type.
- 05 - (XAER_PROTO) routine invoked in improper context.
- 06 - (XAER_RMFAIL) resource manager unavailable.
- 07 - (XAER_DUPID) the XID already exits.
- 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.
- 14 - The enlistment for DB2 with the Microsoft Distributed Transaction Coordinator has failed. MSDTC service may be down. You must terminate the current transaction.
- 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.
- 41 - Transaction is not heuristically committed on some MPP subordinator nodes because of node failure.
- 42 - DB2 Connect XA support is only available to outbound connections that were established using TCPIP.
- 69 - Database log ID mismatch during DUOW resynchronization.
- 85 - As a result of heuristic processing, the transaction has partially committed and rolled back.
- 210 - Only heuristic commits are allowed for this transaction. Some nodes are already in commit state.
- 221 - The version of DBMS on the host requires that all applications that participate in the same XA transaction use the same user ID to connect to the database.
- 222 - The version of DBMS on the host requires that all applications that participate the same XA transaction have the same CCSID.
- 223 - DB2 Connect XA support is only available to local clients or remote clients that use TCPIP to establish the inbound connection.
- 224 - DB2 Connect XA support is only available to clients that are at least version 7.1.
- 225 - Operation Invalid in an XA Loosely Coupled Transaction. Possible subcodes are:
  - 01 - DDL statements are not allowed.
  - 02 - updates and deletes specifying WHERE CURRENT OF clause are not allowed.
  - 03 - buffered insert operations are not allowed.
- 226 - The transaction has already been rolled back.

- 227 - Heuristic commit should be run on the coordinator node.

User Response: 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 nickname, the location of the failed data source will always appear in federated server'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 14, ensure that the MSDTC service is active.

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 41, you can find more information about the problem in the administration notification log. You need to restart DB2 on the failed nodes. It may be necessary to contact the system administrator for assistance.

For reason code 42, an outbound connection can be established by modifying the gateway catalogs to use TCPIP communication protocol.

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, you must manually check every data source updated by the transaction to correct the data.

For reason code 210, some of the nodes are already in commit state. You have to perform heuristic commit to resolve the indoubt transaction.

For reason code 221, ensure all applications that participate in the same XA transaction use the same user ID to connect to the database.

For reason code 222, ensure all applications that participate in the same XA transaction have the same CCSID.

For reason code 223, modify the application and client set-up to use local clients, or for remote clients use TCPIP as the communication protocol to connect to the gateway.

For reason code 224, update the client to version 7.1 or later.

For reason code 225, ensure the application did not contain any of the invalid operations in an XA Loosely Coupled Transaction. Common application server environments where XA Loosely Coupled Transaction can be used are IBM Encina Transaction Server, IBM WebSphere Application Server, Microsoft Transaction Server, and BEA Tuxedo. If you encounter this reason code, then you must review your application and ensure none of the above actions are performed.

For reason code 226, the transaction has reached a rollback state, and is now rolled back on this node.

For reason code 227, heuristic commit should be run on the coordinator node, see db2diag.log for the coordinator node number.

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 federated database, you should also record the location of the failed data source found in the federated server's system log.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. 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 a federated server.

There might be additional information in the console or message logs of the transaction manager and the database manager.

sqlcode: -998

sqlstate: 58005

SQL1000 - SQL1099

SQL1000N *alias* **is not a valid database alias name.**

Explanation: 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.

User Response: Resubmit the command with the correct alias.

SQL1001N *name* **is not a valid database name.**

Explanation: 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.

User Response: Resubmit the command with the correct database name.

sqlcode: -1001

sqlstate: 2E000

SQL1002N *drive* **is not a valid drive.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Ensure 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.**

Explanation: There is not enough storage on the specified file system to process the command.

In a partitioned database environment on Windows environments, each node in the partitioned database group must have the exact same physical hard drive specification (letter) available and useable (must contain useable space) for the CREATE DATABASE command to succeed. The physical hard drive letter is specified in the database manager configuration. If DFTDBPATH is left blank the default will be the hard drive where DB2 is installed on the instance owning machine (db2 installation path).

The command cannot be processed.

User Response: Choose a different file system or erase some non-database files from the

specified file system to make space for the database manager functions.

In a partitioned database environment on OS/2 and Windows environments, follow the following steps:

- Determine which hard drive specification (letter) is required. The drive letter is specified in the error message.
- Determine which node/s of the database partition are experiencing the problem. You can usually find this information in the db2diag.log file of the instance owning node.
- Correct the drive problem on the individual node that is experiencing the problem or change the drive specification in the database manager configuration so that the same drive is available (and has sufficient space) on each and every node of the partitioned database group.
- re-issue the command.

SQL1005N **The database alias** *name* **already exists in either the local database directory or the system database directory.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: There are corrupted internal database pages or internal logic error for a table space.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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 Unix, OS/2, Windows NT, and Windows 95 platforms, the type can be '3' for a database that has a DCE global name.

The command cannot be processed.

User Response: Resubmit the command with a valid type.

SQL1011N   No path was specified in the CATALOG DATABASE command for an indirect entry.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the command with the *nodename* parameter or a different type.

SQL1013N **The database alias name or database name** *name* **could not be found.**

Explanation: 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.

User Response: 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.

For CREATE DATABASE with the AT NODE clause, ensure that the database name is in the system database directory with an entry type of INDIRECT and with a catalog node number that does not equal -1.

Federated system 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

sqlstate: 42705

SQL1014W **There are no more entries in the directory, file, or list being scanned.**

Explanation: The scan of the directory, file, or list is finished.

User Response: No action is required.

SQL1015N **The database must be restarted because the previous session did not conclude normally.**

Explanation: The database must be restarted because of an abnormal termination of the previous session (for example, a power failure).

The command cannot be processed.

User Response: 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. In a partitioned database server environment, the command must be issued on all nodes.

sqlcode: -1015

sqlstate: 55025

SQL1016N **The local_lu alias** *name* **specified in the CATALOG NODE command is not valid.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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 specified node name cannot be the same as the local instance name.

The command cannot be processed.

User Response: Ensure the node name is not the same as the local instance name by displaying the value of the DB2INSTANCE environment variable. On UNIX operating systems, display the DB2INSTANCE environment variable by entering the following command:

```
echo $DB2INSTANCE
```

On Windows and OS/2 operating systems, display the DB2INSTANCE environment variable by entering the following command:

```
echo %DB2INSTANCE%
```

Resubmit the command with the correct node name.

SQL1020C **The node directory is full.**

Explanation: The node directory cannot hold any more entries.

The command cannot be processed.

User Response: Uncatalog any unnecessary entries in the node directory.

SQL1021N **The node name** *name* **specified in the UNCATALOG NODE command was not found.**

Explanation: The *nodename* specified in the command could not be found in the node directory.

The command cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: An error occurred in the communications conversation.

The command cannot be processed.

User Response: Retry the original command. If the error persists, contact your communications administrator.

Federated system users: this situation can also be detected by the data source.

sqlcode: -1023

sqlstate: 08001

SQL1024N   A database connection does not exist.

Explanation: 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.

User Response: 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   The database manager was not stopped because databases are still active.

Explanation: 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.

User Response: 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.

Explanation: A start database manager command is already processed.

The command cannot be processed.

User Response: Because the command is already processed, the application may continue processing.

SQL1027N   The node directory cannot be found.

Explanation: The list node directory command cannot be processed because the node directory could not be found.

The command cannot be processed.

User Response: Issue a CATALOG NODE command with the appropriate parameters and resubmit the current command.

SQL1029N   The partner_lu alias *name* specified in the CATALOG NODE command is not valid.

Explanation: 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.

User Response: 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.

Explanation: Either the system database directory or the local database directory cannot hold any more entries.

The command cannot be processed.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Issue a start database manager command and resubmit the current command.

If using multiple logical nodes, ensure the DB2NODE environment variable is set correctly. The DB2NODE environment variable indicates the node the application will attempt to connect to. DB2NODE must be set to the node number of one of the nodes defined on the same host as the application.

sqlcode: -1032

sqlstate: 57019

SQL1033N   **The database directory cannot be accessed because it is being used.**

Explanation: The database directory cannot be accessed 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.

User Response: Wait until the access has completed, 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.**

Explanation: 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.

User Response: 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. In a partitioned database server environment, check the syslog to find out if the RESTART command fails because of node or communication failures before restoring the database from a backup. If so, ensure the database manager is up and running and communication is available among all the nodes, then resubmit the restart command.

If you encountered this error during roll-forward processing, you must restore the database from a backup and perform roll-forward again.

Note that in a partitioned database environment, the RESTART database command is run on a per-node basis. To ensure that the database is restarted on all nodes, use the command:

```
db2_all db2 restart database
   <database_name>
```

This command may have to be run several times to ensure that all in-doubt transactions have been resolved.

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.**

Explanation: 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.

- Activate/Deactivate database is in progress.

The command cannot be processed.

User Response: 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.

Explanation: 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.
- The system cannot create the database because an interrupt was received while the system was creating or deleting a database file or a directory for the database.
- The system cannot locate the database subdirectory or database configuration file during connect.

The database cannot be used.

User Response: 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.

If the error occurred while trying to connect to a database, get a trace and call IBM support for the possibility of recovering the database.

sqlcode: -1036

sqlstate: 58030

SQL1037W   The Node directory is empty.

Explanation: An attempt was made to read the contents of the node directory, but no entries existed.

User Response: No action is required.

sqlcode: +1037

sqlstate: 01606

SQL1038C   An I/O error occurred while accessing the node directory.

Explanation: The node directory could not be accessed because of an I/O error.

The command cannot be processed.

User Response: 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.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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. 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.

Federated system 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.**

Explanation: The CREATE DATABASE command failed while initializing the system catalogs.

User Response: 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. 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.**

Explanation: The user may have pressed the interrupt key sequence.

Processing is stopped.

Federated system users: this situation can also be detected by the data source.

User Response: Continue processing to handle the interrupt.

If installing the sample database, drop it and install the sample database again.

If starting the database manager, issue a db2stop before issuing any db2 commands.

sqlcode: -1044

sqlstate: 57014

SQL1045N   **The database was not found because it was not cataloged correctly.**

Explanation: An indirect entry in the database directory points to another non-HOME entry.

The command cannot be processed.

User Response: Verify the entries in all the associated database directories with the directory scan commands.

sqlcode: -1045

sqlstate: 58031

SQL1046N   **The authorization ID is not valid.**

Explanation: 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 30 characters for Windows platforms or 8 characters for other platforms.
- 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.

User Response: Log on with a valid authorization ID.

Federated system users: if necessary isolate the problem to the data source rejecting the request (see the Troubleshooting 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.**

Explanation: 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.

User Response: 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, an X for exclusive use or an N for exclusive use at single node. For DB2 Connect connections, only S is supported. N is only supported in MPP configuration.**

Explanation: 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 DB2 Connect, only shared access is allowed. Mnemonic equates for these values are provided in the SQLENV.H file.

The command cannot be processed.

User Response: 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.**

Explanation: The connection to the database has been lost.

The command cannot be processed.

User Response: Issue a CONNECT RESET statement.

SQL1050N   **The database cannot be uncataloged because it is a home database.**

Explanation: 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.

User Response: Continue processing if the database name was specified correctly.

SQL1051N   **The path** *path* **for the database directory does not exist.**

Explanation: 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.

User Response: Resubmit the command with the correct path for the database directory.

sqlcode: -1051

sqlstate: 57019

SQL1052N   **The database path** *path* **does not exist.**

Explanation: The path specified in the *path* parameter of the command is not valid. There is no path with that name.

In a partitioned database environment on Windows environments, each node in the partitioned database group must have the exact same physical hard drive specification (letter) available and useable (must contain useable space) for the CREATE DATABASE command to succeed. The physical hard drive letter is specified in the database manager configuration. If DFTDBPATH is left blank the default will be the hard drive where DB2 is installed on the instance owning machine (db2 installation path).

The command cannot be processed.

User Response: Resubmit the command with

the correct database path.

In a partitioned database environment on OS/2 and Windows environments, follow the following steps:

- Determine which hard drive specification (letter) is required. The drive letter is specified in the error message.
- Determine which node/s of the database partition are experiencing the problem. You can usually find this information in the db2diag.log file of the instance owning node.
- Correct the drive problem on the individual node that is experiencing the problem or change the drive specification in the database manager configuration so that the same drive is available (and has sufficient space) on each and every node of the partitioned database group.
- re-issue the command.

SQL1053N   An interrupt is already being processed.

Explanation: The system is currently processing an interrupt so another interrupt is not honored.

The interrupt request is ignored.

User Response: Wait until the current interrupt processing finishes and resubmit the command.

SQL1054N   A COMMIT is in progress and cannot be interrupted.

Explanation: The system is currently processing a COMMIT. The user entered an interrupt key sequence.

The interrupt request is ignored.

Federated system users: this situation can also be detected by the data source.

User Response: Wait until COMMIT is finished and resubmit the request.

SQL1055N   A ROLLBACK is in progress and cannot be interrupted.

Explanation: The system is currently processing a ROLLBACK. The user entered an interrupt key sequence.

The interrupt request is ignored.

Federated system users: this situation can also be detected by the data source.

User Response: Wait until ROLLBACK is finished and resubmit the request.

SQL1056N   There are already eight database directory scans open.

Explanation: Eight database directory scans in this process are already open. No more than eight open scans are allowed.

The command cannot be processed.

User Response: Issue one or more CLOSE DIRECTORY SCAN commands and resubmit the command.

sqlcode: -1056

sqlstate: 54029

SQL1057W   The system database directory is empty.

Explanation: An attempt was made to read the contents of the system database directory, but no entries existed.

User Response: No action is required.

sqlcode: +1057

sqlstate: 01606

SQL1058N   The handle parameter in the Directory Scan command is not valid.

Explanation: 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.

User Response: Resubmit the command with a valid *handle* parameter.

SQL1059N **A Get Next command cannot be processed because no Open Scan command was issued.**

Explanation: A directory scan command was issued before the scan was opened.

The command cannot be processed.

User Response: 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.**

Explanation: 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.

Federated system users: this situation can also be detected by the data source

The command cannot be processed.

User Response: Contact the system administrator or database administrator for the database and request a GRANT CONNECT for the authorization ID. Resubmit the command.

Federated system users: if necessary isolate the problem to the data source rejecting the request (see the Troubleshooting Guide for procedures to follow to identify the failing data source) and ensure that correct privileges have been granted for that data source.

sqlcode: -1060

sqlstate: 08004

SQL1061W **RESTART completed successfully, but indoubt transactions exist against the database.**

Explanation: 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.

User Response: 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.

Note that in a partitioned database server environment, the RESTART database command is run on a per-node basis. In order to ensure that the database is restarted on all nodes, use the following command:

```
db2_all db2 restart database
  <database_name>
```

Issuing the above command will resolve any in-doubt transactions if all nodes are operational.

This command may have to be run several times to ensure that all in-doubt transactions have been resolved.

SQL1062N **The database path *path* was not found.**

Explanation: 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.

In a partitioned database environment on Windows environments, each node in the partitioned database group must have the exact same physical hard drive specification (letter) available and useable (must contain useable space) for the CREATE DATABASE command to succeed. The physical hard drive letter is specified in the database manager configuration. If DFTDB is left blank the default will be the hard drive where DB2 is installed on the instance

owning machine (db2 installation path).

The command cannot be processed.

User Response: Verify the path or the default path and make sure they exist in the system. Resubmit the command.

In a partitioned database environment on OS/2 and Windows environments, follow the following steps:

- Determine which hard drive specification (letter) is required. The drive letter is specified in the error message.
- Determine which node/s of the database partition are experiencing the problem. You can usually find this information in the db2diag.log file of the instance owning node.
- Correct the drive problem on the individual node that is experiencing the problem or change the drive specification in the database manager configuration so that the same drive is available (and has sufficient space) on each and every node of the partitioned database group.
- re-issue the command.

SQL1063N   The start database manager processing was successful.

Explanation: The command to start the database manager completed successfully.

SQL1064N   The stop database manager processing was successful.

Explanation: 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*.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

SQL1068N **The domain that owns the user ID *user-ID* for the CONNECT or ATTACH statement is not defined in the DB2DOMAINLIST environment variable.**

Explanation: A user ID in a CONNECT TO or ATTACH TO statement does not belong to a domain defined in the DB2DOMAINLIST environment variable.

User Response: Specify the name of the domain which owns the user ID in the DB2DOMAINLIST environment variable by using the DB2SET command.

sqlcode: -1068

sqlstate: 08004

SQL1069N **The database *name* is not a home database.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Ensure that a valid address is used in the application program and the input string is null terminated.

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.

Explanation: 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).

User Response: 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).

- 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, there may be an inode conflict. This is probably the case if you find that two instances cannot be active at the same time, but can be started individually. It is necessary to change the inode being used to generate IPC keys for one of the instances.

  For single-node instances, perform the following as the instance owner from the sqllib directory:

  – delete the .ftok file

    rm .ftok

  – create a new .ftok file

    touch .ftok

  For multi-node instances, perform the following as the instance owner:

  – 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.

Explanation: The node directory release number does not match the product expected release number. The node directory could be from a previous release.

User Response: Re-catalog all the node entries and try the command again.

SQL1074N   **The address of the password parameter is not valid.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The application program used an address that is not valid for the *count* parameter.

The command cannot be processed.

User Response: Ensure that a valid address is used in the application program.

SQL1077N   **The address of the handle parameter is not valid.**

Explanation: The application program used an address that is not valid for the *handle* parameter.

The command cannot be processed.

User Response: Ensure that a valid address is used in the application program.

SQL1078N   **The address of the buffer parameter is not valid.**

Explanation: The application program used an address that is not valid for the *buffer* parameter.

The command cannot be processed.

User Response: Ensure that a valid address is used in the application program.

SQL1079N   **The address of the nodename parameter is not valid.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Ensure that a valid address is used in the application program and the input string is null terminated.

SQL1083N   The database description block cannot be processed, reason code=*reason-code*.

Explanation: 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.

User Response: Correct the error and resubmit the command.

SQL1084C   Shared memory segments cannot be allocated.

Explanation: 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.

User Response: 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. 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.

Explanation: 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.

User Response: 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.**

Explanation: The command received an error from the operating system that prevents further processing.

The command cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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 error will also occur if the application program is using down-level database manager libraries or DLLs while accessing the installed version of the database manager configuration file.

The command cannot be processed.

User Response: Verify that there is no older version of the database manager libraries or DLLs that will be picked up for your application processing.

If the problem persists, repeat the precompile process 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.**

Explanation: The release number of the database is not compatible with the release number of the installed version of the database manager. This can be the release number when the database was created, the release number when the database was last migrated, or the highest version, release, modification, and fixpak level defined in the catalog.

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. If the error occurred on the

first connection after removing a fixpak, you are attempting to connect to a database that is defined to use a higher level of database manager code.

User Response: 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. If the error occurred on the first connection after removing a fixpak, the database should be restored from a backup prior to using the utility to update the database to the fixpak level.

sqlcode: -1091

sqlstate: 08004

SQL1092N *authorization-ID* **does not have the authority to perform the requested command.**

Explanation: The user attempted to execute a command or operation without having the proper authority for that command or operation.

The command cannot be processed.

Federated system users: this situation can also be detected by the data source.

User Response: 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.

If you are using Kerberos authentication in a Windows environment, ensure that you log on to the machine using a domain account. Only domain users can use Kerberos authentication in a Windows 2000 environment.

If you are using LDAP support, ensure that you or the DB2 Connect gateway has the authority to perform the CATALOG DATABASE, NODE and DCS DATABASE commands. Invoke the command "UPDATE DBM CFG USING CATALOG_NOAUTH YES" at the client or the gateway to correct the problem.

Federated system users: if necessary isolate the problem to the data source rejecting the request (see the Troubleshooting Guide for procedures to follow to identify the failing data source) and ensure that the authorization id specified has the proper authority on that data source.

Contact the System Administrator for authority request assistance. Do not attempt to execute the command without appropriate authorization.

SQL1093N **The user is not logged on.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the command after the update is finished.

sqlcode: -1094

sqlstate: 57009

SQL1095N   There are already eight node directory scans open.

Explanation: Eight node directory scans are already open in this process and no more than eight are allowed.

The command cannot be processed.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

Federated system users: this situation can also be detected by the data source.

User Response: 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.

Federated system 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, issue a CATALOG NODE command for that node.

sqlcode: -1097

sqlstate: 42720

SQL1098N   The application is already connected to this database.

Explanation: A connection to a database was requested but the application is already connected to the specified database.

The command cannot be processed.

User Response: No action is necessary.

sqlcode: -1098

sqlstate: 53056

SQL1099N   The diskette is write-protected.

Explanation: A write operation was attempted to a database residing on a write-protected diskette.

The statement cannot be processed.

User Response: Verify you are using the correct diskette. Remove protection from the diskette, if necessary.

SQL1100 - SQL1199

SQL1100W   **The node name** *name* **specified in the Catalog Database command has not been cataloged in the Node Directory.**

Explanation: 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.

User Response: Issue a CATALOG NODE command.

SQL1101N   **Remote database** *name* **on node** *node-name* **could not be accessed with the specified authorization id and password.**

Explanation: 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.

User Response: Resubmit the request with a valid authorization ID and password combination for the remote system.

SQL1102N   **A database name was not specified.**

Explanation: No database name was provided to proceed with migration.

The statement cannot be processed.

User Response: Specify a database name for migration.

SQL1103W   **The Migrate Database command processing was successful.**

Explanation: 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.

User Response: No action is required.

SQL1104N   **The address of the program name parameter is not valid.**

Explanation: The application program has used an address for the program name that is not valid.

The command cannot be processed.

User Response: Use a correct address in the application program.

SQL1105N   **The SQL CONNECT RESET statement is not allowed in a Remote Application Interface Procedure.**

Explanation: The remote application procedure contains an SQL CONNECT RESET statement.

The remote procedure is not allowed to continue.

User Response: Remove the SQL CONNECT RESET statement and retry the remote procedure.

sqlcode: -1105

sqlstate: 38003

SQL1106N   **The specified DLL** *name* **module was loaded, but the function** *function* **could not be executed.**

Explanation: The function within the DLL (dynamic link library) cannot be found.

The command cannot be processed.

User Response: 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*.

Explanation: When loading the DLL (dynamic link library) module, the command was interrupted, perhaps by Ctrl+Break.

Processing is stopped.

User Response: 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*.

Explanation: 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.

User Response: 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. Then contact your technical service representative.

sqlcode: -1108

sqlstate: 42724

SQL1109N **The specified DLL** *name* **could not be loaded.**

Explanation: The DLL (dynamic link library) module specified could not be found.

The command cannot be processed.

User Response: 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.**

Explanation: The data area is not initialized correctly.

The command cannot be processed.

User Response: Ensure that the user-supplied input SQLDA or output SQLDA fields are initialized correctly.

SQL1111N **The program name** *name* **specified was not valid.**

Explanation: The DLL (dynamic link library) module or program name syntax is not correct.

The command cannot be processed.

User Response: 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*.

Explanation: There is not enough random access memory (RAM) to load the specified DLL (dynamic link library) module.

The command cannot be processed.

User Response: 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

SQL1113N **The data type for sqlvar** *n* **in the output SQLDA was changed from** *type-1* **to** *type-2***.**

Explanation: 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.

User Response: 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***.**

Explanation: 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.

User Response: 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***.**

Explanation: 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.

User Response: Correct the remote stored procedure so the sqld field in the output SQLDA is not changed.

sqlcode: -1115

SQL1116N **A connection to or activation for database** *name* **cannot be made because of BACKUP PENDING.**

Explanation: The specified database requires a backup to be taken to provide a starting point for roll-forward recovery.

No connection was made.

User Response: 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.**

Explanation: The specified database is enabled for roll-forward recovery and it has been restored but not rolled forward.

No connection was made.

Federated system users: this situation can also be detected by the data source.

User Response: 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.

Federated system users: if necessary isolate the problem to the data source rejecting the request (see the Troubleshooting 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.**

Explanation: The database is in an inconsistent state because a system error occurred during a backup function.

No connection was made.

Federated system users: this situation can also be detected by the data source.

User Response: Issue the BACKUP command and then try the command again.

Federated system users: if necessary isolate the problem to the data source rejecting the request (see the Troubleshooting 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.**

Explanation: The database is in an inconsistent state because a system error occurred during a restore function.

No connection was made.

Federated system users: this situation can be detected by the data source.

User Response: Issue the RESTORE command and then try the command again.

Federated system users: if necessary isolate the problem to the data source rejecting the request (see the Troubleshooting 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.**

Explanation: 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.

No connection was made.

Federated system users: this situation can also be detected by the data source.

User Response: Issue the BACKUP or RESTORE command, and then try the command again.

Federated system users: if necessary isolate the problem to the data source rejecting the request (see the Troubleshooting 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.**

Explanation: 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.

User Response: Verify that the program allocates the required buffer area and resubmit the command.

SQL1122N   **The address of the protocol structure parameter is not valid.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Verify the protocol type in the node structure and resubmit the command.

SQL1124N   The remote workstation name *name* **is not valid.**

Explanation: 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.

User Response: 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.**

Explanation: The adapter number specified in the NETBIOS protocol structure for the Catalog command is not valid.

The command cannot be processed.

User Response: Verify that the adapter number is valid and resubmit the command.

SQL1126N   The network ID *ID* **is not valid.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Verify the characters specified in the remote LU name. Resubmit the command with a remote LU name.

SQL1128W   SAVECOUNT is ignored.

Explanation: Consistency points are not supported when loading into MDC tables.

User Response: No action is required.

SQL1129N   **New DARI (Stored Procedure) process could not be started because there are insufficient resources for a new process to be created.**

Explanation: New DARI (Stored Procedure) process could not be started because there are insufficient resources for a new process to be created.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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).**

Explanation: An illegal command has been made within the scope of DARI (Stored Procedure).

The DARI procedure is not allowed to continue.

User Response: Remove the illegal command in the DARI procedure and retry.

sqlcode: -1132

sqlstate: 38003

SQL1133N   **A pointer address inside an sqlvar (index = n) of the output SQLDA was modified within the DARI (Stored Procedure) function.**

Explanation: 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.

User Response: 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).**

Explanation: 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.

User Response: Remove the illegal command in the DARI procedure and retry.

sqlcode: -1134

sqlstate: 38003

SQL1135N **An invalid value was specified for the number of segments when creating a database.**

Explanation: The value that was specified for the number of segments was out of range. The valid range is from 1 to 256.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: Either the list of containers could not be accessed or some failure occurred while

attempting to remove them or the database directory.

User Response: 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.

SQL1138W **The unique index** *name* **was migrated to support deferred uniqueness checking. A new index was not created.**

Explanation: A CREATE INDEX operation was attempted on an existing index. Since the index had not yet been migrated to support deferred uniqueness checking, this migration was performed.

The migrated format of the unique index will allow multiple row updates to check the uniqueness of the columns of the index at the end of the update statement instead of when each row is updated.

User Response: No action is required.

sqlcode: +1138

sqlstate: 01550

SQL1139N **The total size of the table space is too big.**

Explanation: The size of the current table space is too big. The size of a REGULAR or USER TEMPORARY table space is limited to 0xFFFFFF (16777215) pages while the size of a SYSTEM TEMPORARY or LONG table space is limited to 2 terabytes (2 TB).

User Response: Check the administration notification log for details. Reduce the size of the table space and correct the SQL statement.

sqlcode: -1139

sqlstate: 54047

SQL1140W   **Estimated processor cost of**
estimate-amount1 **processor seconds**
(*estimate-amount2* **service units) in**
cost category *cost-category* **exceeds**
a resource limit warning threshold
of *limit-amount* **service units.**

Explanation: The prepare of a dynamic INSERT,
UPDATE, DELETE, or SELECT SQL statement
resulted in a cost estimate that exceeded the
warning threshold value specified in the resource
limit specification table (RLST).

This warning is also issued if DB2's cost category
value was "B", and the default action specified in
the RLF_CATEGORY_B column in the RLST is to
issue a warning.

estimate_amount1
> The cost estimate (in processor seconds)
> if the prepared INSERT, UPDATE,
> DELETE or SELECT statement were to
> be executed.

estimate_amount2
> The cost estimate (in service units) if the
> prepared INSERT, UPDATE, DELETE or
> SELECT statement were to be executed.

cost-category
> DB2's cost-category for this SQL
> statement. The possible values are A or
> B.

limit-amount
> The warning threshold (in service units)
> specified in the RLFASUWARN column
> of the RLST.

The prepare of the dynamic INSERT, UPDATE,
DELETE, or SELECT statement was successful.
An SQLCODE -905 might be issued if the
execution of the prepared statement exceeds the
ASUTIME value specified in the RLST.

User Response: Ensure that there is application
logic to handle the warning to either allow the
statement to execute or to stop the statement
from being executed. If this SQLCODE was
returned because the cost category value is "B", it
might be that the statement is using parameter

markers or that some statistics are not available
for the referenced tables and columns. Make sure
the administrator has run the utility RUNSTATS
on the referenced tables. It might also be that
UDFs will be invoked when the statement is
executed, or for INSERT, UPDATE, or DELETE
statements that triggers are defined on the
changed table. Check the
DSN_STATEMNT_TABLE or the IFCID 22 record
for this statement to find the reasons this SQL
statement has been put in cost category "B".

If the warning is caused by an SQL statement
that is consuming too much processor resource,
attempt to rewrite the statement to perform more
efficiently. Another option is to ask the
administrator to increase the warning threshold
value in the RLST.

sqlcode: +1140

sqlstate: 01616

SQL1141N   **The operation completed with**
errors or warnings. The details are
located in the result file *file-name*
that needs to be formatted with
the db2inspf utility.

Explanation: In a partitioned database
environment, the file extension corresponds to
the database partition node number. The files are
located in the directory specified by the
DIAGPATH database manager configuration
parameter.

User Response: Use the db2inspf utility to
format the inspection data result file *file-name*.

SQL1142N   **The operation cannot complete**
because the file *file-name* **is**
already in use.

Explanation: There is an existing file using the
file name specified in the INSPECT command.

The command cannot be processed.

User Response: Resubmit the command using a
different file name that is not already in use. Or
remove the existing file *file-name* and resubmit
the command.

In a partitioned database environment, the file extension corresponds to the database partition node number. The files are located in the directory specified by the DIAGPATH database manager configuration parameter.

SQL1143N   **The operation cannot complete because a file error occurred for the file** *file-name***.**

Explanation: The file could not be accessed because of an I/O error. The system cannot open, read from, or write to the file. The file may be incomplete or the disk may be full.

The utility stops processing.

User Response: If possible, record all error information from the SQLCA. Look in the diagnostics log file for additional information.

In a database partitioned environment, the file extension corresponds to the database partition node number. The files are located in the directory specified by the DIAGPATH database manager configuration parameter.

SQL1144N   **The current transaction was rolled back as a result of a failure creating an index, sqlcode =** *sqlcode***.**

Explanation: In the same transaction, the last remaining index on a table was dropped and then a new index was created on the same table. The index creation either failed with an error, *sqlcode*, or was rolled back by a ROLLBACK TO SAVEPOINT statement (sqlcode=0). A rollback of the index creation cannot be successfully completed when a drop index of the last remaining index on a table has not yet been committed. The entire transaction is rolled back for either of these cases. Note that indexes are also created or dropped by adding or dropping unique or primary key constraints in an ALTER TABLE statement.

User Response: Where possible, commit the drop of an index before creating a new index on the same table in order to avoid rolling back the entire transaction. If the *sqlcode* is not 0, then refer to the message for the sqlcode for corrective

action. All statements in the rolled back transaction need to be issued again. If the *sqlcode* is zero and the index is dropped within the savepoint scope, the statements may need to be re-ordered to create the new index before dropping the old index on the same table. If the drop and add unique or primary key constraints are in the same ALTER TABLE statement, this may need to be done in two statements, performing the ADD in the first statement and the drop in the second statement.

sqlcode: -1144

sqlstate: 40507

SQL1145N   **PREPARE statement is not supported when using a gateway concentrator. Reason code:** *reason-code***.**

Explanation: The statement failed for one of the following reasons, based on *reason-code*.

1   When gateway concentrator feature is ON, dynamically prepared statements from embedded SQL are not supported. In this configuration, dynamically prepared statements are only supported if the client is a CLI application.

2   When the gateway concentrator feature is ON, dynamically prepared SET statements are not supported.

User Response: Based on the reason code, perform the following actions:

1   Change the application to use CLI for dynamic SQL statements, or change the application to use static SQL.

2   Use EXECUTE IMMEDIATE for SET statements.

sqlcode: -1145

sqlstate: 560AF

SQL1146N There are no indexes on table *table-name*.

Explanation: The *table-name* specified for index reorganization has no indexes.

User Response: Resubmit the command with a valid table name.

SQL1147W TOTALFREESPACE is ignored for MDC tables.

Explanation: Because the table free space is well managed for MDC tables, the TOTALFREESPACE file type modifier is unnecessary and will be ignored.

User Response: No action required.

SQL1150N The address of the user id parameter is not valid.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Ensure that the TSR for the specified communications protocol has been successfully loaded, then run your application again.

SQL1161W Reconcile processing failed. DataLink column(s) not defined on DB2 DataLinks Manager(s). Check the administration notification log for details.

Explanation: Metadata information about one of more DataLink columns of the table are missing on DB2 DataLinks Managers. Reconcile processing has failed. Table is placed in DataLink Reconcile Not Possible (DRNP) state.

User Response: To take the table out of DataLink Reconcile Not Possible state, follow the procedure mentioned under "Removing a table from the DataLink Reconcile Not Possible state" in the Administration Guide.

SQL1162W Reconcile processing failed. DB2 DataLinks Manager(s) became unavailable during exception processing.

Explanation: One of more DB2 DataLinks Managers referred to by the table data became unavailable during reconcile exception processing. Reconcile processing has failed. Table is placed in DataLink Reconcile Pending (DRP) state.

User Response: Run reconcile again.

SQL1163N The table cannot be enabled for data capture as the identifier name *ident-name* of type *ident-type* is too long.

Explanation: Data capture is not supported with certain identifier types exceeding specific lengths. In processing the attempted change, it was found that the identifier *ident-name* of type *ident-type* is too long. The identifier types and the maximum lengths allowed for data capture enablement are as follows:

1. column. Column names cannot be longer than 18 bytes for data capture to be enabled.

2. table. Table names cannot be longer than 18 bytes for data capture to be enabled.

3. schema. Schema names cannot be longer than 18 bytes for data capture to be enabled.

User Response: If data capture is to be enabled for this table than ensure that the identifiers in question do not exceed the maximum sizes stated above. Otherwise disable data capture for the table in order to use longer identifier names.

sqlcode: -1163

sqlstate: 42997

SQL1164N **The SQLDA or host variables, of type** *type***, used in the SQL statement are not valid. Reason code** *reason-code***, host variable/SQLVAR number** *var-number***.**

Explanation: An error occurred while processing the SQLDA or host variable(s) in the SQL statement.

The call parameter list, which is created by the precompiler, may not be 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-written the call parameter list in another way.

Also, if an SQLDA is passed directly by the application, it may not be initialized correctly.

Host variable/SQLDA types:

1 input host variable(s) or SQLDA

2 output host variable(s) or SQLDA

In SQL statements with host variables, use the host variable number to count in from the beginning of the statement (or substatement, in the case of compound SQL) to locate the invalid host variable. For statements using an SQLDA, the SQLVAR number is used to locate the invalid SQLVAR. For an input SQLDA, count only input host variables or SQLVARs; similarly for output. Note that this number is based at 1, and may not be applicable to all reason codes. Reason codes are interpreted as follows:

1 SQLDA.SQLN is less than SQLDA.SQLD

2 SQLVAR.SQLTYPE is invalid.

3 The length specified in SQLVAR.SQLLEN or SQLVAR2.SQLLONGLEN is incorrect for the SQL type given in SQLVAR.SQLTYPE.

4 A Large Object SQLVAR is present, but the SQLDOUBLED field of SQLDA.SQLDAID is not set to '2'.

5 An input varchar is supplied whose current length (from the length field of the varchar itself) is greater than the maximum length. The maximum length is determined by the declaration (in the case of a host variable) or by the setting of SQLVAR.SQLLEN (in the case of a user-defined SQLDA).

6 An input Large Object is passed in whose current length (either in the length field of the LOB itself, or as indicated by the SQLVAR2.SQLDATALEN pointer) is greater than the maximum length. The maximum length is determined by the declaration (in the case of a host variable) or by the setting of SQLVAR2.SQLLONGLEN (in the case of a user-defined SQLDA).

7 A double-byte character Large Object has an odd value indicated by the SQLVAR2.SQLDATALEN pointer, which is always in terms of bytes, even for DBCLOBs.

8 The SQLDATA pointer is invalid or points to insufficient storage.

9 The SQLIND pointer is invalid or points to insufficient storage.

10 The SQLDATALEN pointer is invalid or points to insufficient storage.

11 A specific number of input host variables/SQLVARS is expected for the current SQL statement.

The statement cannot be processed.

User Response: Examine the application program for any errors noted. Note that the

programmer should not attempt to modify the precompiler output.

sqlcode: -1164

sqlstate: 07002

SQL1165W A value cannot be assigned to a host variable because the value is not within the range of the host variable's data type.

Explanation: 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 processing continued returning a null indicator of -2.

User Response: 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*.

sqlcode: +1165

sqlstate: 01515

SQL1166W Division by zero was attempted.

Explanation: The processing of an arithmetic expression resulted in division by zero. This warning may be returned with a row other than the row which caused the warning. For example, this occurs with arithmetic expressions in predicates or when the query is processed using system temporary tables. The warning may be issued again since it is also returned whenever a null indicator variable is set to -2.

The statement processing continued, using null as the result of the division expression and possibly returning a null indicator of -2.

User Response: 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.

sqlcode: +1166

sqlstate: 01564

SQL1167W Arithmetic overflow or other arithmetic exception occurred.

Explanation: The processing of an arithmetic expression has resulted in an arithmetic overflow, underflow or other arithmetic exception. This warning may be returned with a row other than the row which caused the warning. For example, this occurs with arithmetic expressions in predicates or when the query is processed using system temporary tables. The warning may be issued again since it is also returned whenever a null indicator variable is set to -2.

The statement processing continued, using null as the result of the arithmetic expression and possibly returning a null indicator of -2.

User Response: 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.

sqlcode: +1167

sqlstate: 01519

SQL1178N The federated *object-type* called *object-name* does not reference a federated database object.

Explanation: The object of type *object-type* identified by *object-name* is defined using the keyword FEDERATED but the fullselect in the statement does not reference a federated database object.

The statement cannot be processed.

User Response: Remove the keyword FEDERATED from the statement.

sqlcode: -1178

sqlstate: 429BA

SQL1179W **The** *object-type* **called** *object-name*
may require the invoker to have
necessary privileges on data
source objects.

Explanation: The object identified by *object-name*
references a federated object (such as an OLE DB
table function, federated routine, federated view,
or a nickname), where the actual data exists at a
data source. When the data source data is
accessed, the user mapping and authorization
checking is based on the user that initiated the
operation.

If the *object-type* is SUMMARY TABLE, then the
operation is refreshing the data for the
materialized query table. The user that invoked
the REFRESH TABLE or SET INTEGRITY
statement that causes the refresh may be required
to have the necessary privileges to access the
underlying data source object at the data source.

If the *object-type* is VIEW, then any user of the
view may be required to have the necessary
privileges to access the underlying data source
object at the data source.

If the *object-type* is PROCEDURE, FUNCTION, or
METHOD then the invoker of the routine may be
required to have the necessary privileges to
access the underlying data source object at the
data source for any SQL statements in the
routine.

In any case, an authorization error may occur
when the attempt is made to access the data
source object.

User Response: Granting privileges to the
object may not be sufficient to support
operations that access the data from the data
source. User access may need to be granted at
the data source for the underlying data source
objects.

sqlcode: +1179

sqlstate: 01639

SQL1180N **Routine** *routine-name* **(specific**
name *specific-name***) has caused an**
OLE error. HRESULT=*hresult***.**
Diagnostic text: *message-text***.**

Explanation: DB2 received an OLE error code
while trying to communicate with the OLE
automation server of user defined function
(UDF) or stored procedure *routine-name* (specific
name *specific-name*). HRESULT *hresult* is the
returned OLE error code, and *message text* is the
retrieved error message.

The following shows a partial list of error
messages, HRESULTS, and possible causes. The
error message text may be changed by OLE, and
new error codes may be added by OLE.

Unknown interface (0x80020001):
The specified OLE object does not
support the IDispatch interface.

Type mismatch (0x80020005):
One or more of the SQL data types do
not match the data types of the method
arguments.

Unknown name (0x80020006):
The specified method name was not
found for the specified OLE object.

Invalid number of parameters (0x8002000E):
The number of arguments passed to the
method is different from the number of
arguments accepted by the method.

Invalid class string (0x800401F3):
The specified ProgID or CLSID is
invalid.

Class not registered (0x80040154):
CLSID is not properly registered.

Application not found (0x800401F5):
Local server EXE not found.

DLL for class not found (0x800401F8):
In-process DLL not found.

Server execution failed (0x80080005):
Creation of the OLE object failed.

User Response: Consult the *OLE Programmer's
Reference Guide* for a complete documentation,
including the meaning of the special terminology.

sqlcode: -1180

sqlstate: 42724

SQL1181N   **Routine** *routine-name* **(specific name** *specific-name***) has raised an exception with description** *message-text***.**

Explanation: The user defined function (UDF) or stored procedure *routine-name* (specific name *specific-name*) has raised an exception. The message text shows the textual description of the exception returned by the routine.

User Response: The user will need to understand the meaning of the exception. See the author of the routine.

sqlcode: -1181

sqlstate: 38501

SQL1182N   **User defined function** *function-name* **could not initialize data source object of specified OLE DB provider. HRESULT=***hresult***. Diagnostic text:** *message-text***.**

Explanation: The OLE DB data source object of the specified OLE DB provider could not be instantiated or initialized. *hresult* is the returned OLE DB error code, and *message-text* is the retrieved error message.

The following shows a partial list of HRESULTS and possible causes.

0x80040154
　　　　Class (OLE DB provider) not registered.

0x80040E73
　　　　The initialization string specified does not conform to specification.

0x80004005
　　　　Unspecified error (during initialization).

User Response: Verify proper registration of the OLE DB provider and correct initialization of the parameters in the connection string. Consult Microsoft OLE DB Programmer's Reference and Data Access SDK for a complete documentation

of HRESULT codes and Data Link API connection string syntax in OLE DB Core Components.

sqlcode: -1182

sqlstate: 38506

SQL1183N   **User defined function** *function-name* **received an OLE DB error from specified OLE DB provider. HRESULT=***hresult***. Diagnostic text:** *message-text***.**

Explanation: The specified OLE DB provider returned an OLE DB error code. *hresult* is the returned OLE DB error code, and *message-text* is the retrieved error message.

The following shows a partial list of HRESULTS and possible causes.

0x80040E14
　　　　The command contained one or more errors, e.g. syntax error in pass-through command text.

0x80040E21
　　　　Errors occurred, e.g. the supplied columnID was invalid (DB_INVALIDCOLUMN).

0x80040E37
　　　　The specified table does not exist.

User Response: Consult Microsoft OLE DB Programmer's Reference and Data Access SDK for a complete documentation of HRESULT codes.

sqlcode: -1183

sqlstate: 38506

SQL1184N   **One or more EXPLAIN tables were not created using the current version of DB2.**

Explanation: EXPLAIN will not be able to insert into these tables until they are migrated using DB2EXMIG, or dropped and created with the EXPLAIN.DDL CLP script for the current version of DB2.

User Response: Either migrate the tables using

DB2EXMIG, or drop and re-create them with the EXPLAIN.DDL CLP script for the current version of DB2. Re-issue the command.

sqlcode: -1184

sqlstate: 55002

SQL1185N   **FEDERATED** *value* **is incorrectly used in binding the package.**

Explanation: If *value* is NO, then at least one static SQL statement in the package includes a reference to either a nickname or OLE DB table function or OLE DB procedure. In this case, you must specify FEDERATED YES to bind the package.

If *value* is YES, then no static SQL statement in the package includes a reference to either a nickname or OLE DB table function or OLE DB procedure. In this case, you must specify FEDERATED NO to bind the package.

The package is not created.

User Response: Specify the correct FEDERATED option.

SQL1186N   **The object** *object-type* **named** *object-name* **is either being altered to become a federated object, or being created as a federated object, without specifying the FEDERATED clause.**

Explanation: The object identified by *object-name* references a federated object (such as an OLE DB table function, federated routine, federated view, or a nickname) where the actual data exists at a data source.

If a non-federated view is being altered and the fullselect now refers to a federated database object, either directly or indirectly, then FEDERATED must be specified.

If a federated view is being altered and the fullselect still refers to a federated database object, either directly or indirectly, then NOT FEDERATED must not be specified.

If a materialized query table is being created and the fullselect refers to a federated database

object, either directly or indirectly, then NOT FEDERATED must not be specified.

If an SQL routine is being created and the routine refers to a federated database object, either directly or indirectly, then NOT FEDERATED must not be specified.

User Response: To alter a non-federated view to become a federated view, specify the FEDERATED clause.

To alter a federated view that continues to be a federated view, do not specify the NOT FEDERATED clause.

To create a materialized query table in which the fullselect refers to a federated database object, either directly or indirectly, do not specify the NOT FEDERATED clause.

To create an SQL routine which refers to a federated database object, either directly or indirectly, do not specify the NOT FEDERATED clause.

sqlcode: -1186

sqlstate: 429BA

SQL1187W   **The database was created successfully, but an error occurred while creating the detailed deadlocks event monitor** *event-monitor-name*. **The detailed deadlocks event monitor is not created.**

Explanation: CREATE DATABASE could not create a detailed deadlocks event monitor for the newly created database.

User Response: Create a detailed deadlocks event monitor if desired.

SQL1188N   **Column** *src-col-num* **of the SELECT or VALUES statement is not compatible with table column** *tgt-col-num*. **The source column has sqltype** *src-sqltype*, **and the target column has sqltype** *tgt-sqltype*.

Explanation: The source and target columns are not compatible. This may be a result of an incorrect ordering of the columns in the SELECT or VALUES statement, or of an incorrect ordering of the METHOD P or METHOD N specification.

User Response: Correct the command so that the source and target columns are compatible, and then resubmit the command.

SQL1189N   **The LOAD METHOD specified is invalid for the specified file type.**

Explanation: An invalid METHOD was specified for the specified file type.

User Response: Resubmit the command with a different METHOD.

SQL1190N   **The LOAD utility encountered SQLCODE** *sqlcode*, **SQLSTATE** *sqlstate*, **and message tokens** *token-list*.

Explanation: The LOAD utility encountered an error with the SQLCODE *sqlcode* and has stopped processing.

User Response: Look up the SQLCODE *sqlcode* in the Message Reference. Take the required corrective actions, and then resubmit the command.

SQL1191N   **Column** *col-spec* **specified in the METHOD parameter does not exist.**

Explanation: Column *col-spec* does not exist in the result of the query.

User Response: Correct the invalid column specification and resubmit the command.

SQL1192W   **Too many input sources were specified for the current filetype. The maximum number allowed is** *max-input-sources*.

Explanation: The filetype specified does not allow more than *max-input-sources* input sources to be specified for a single load.

User Response: Resubmit the command with a number of input sources that does not exceed *max-input-sources*.

SQL1193I   **The utility is beginning to load data from the SQL statement** *statement*.

Explanation: This is an informational message indicating that a load from an SQL statement has begun.

User Response: No action is required.

SQL1194W   **The value in row** *row-num* **corresponding to target column number** *col-num* **is not valid. The row has been rejected.**

Explanation: Row number *row-num* has been rejected because the value corresponding to target column *col-num* is not a valid value for that column.

Possible reasons are :

- A numeric value is not within the minimum or maximum range of its target column.
- A character value is too long for its target column.

User Response: Examine the input value. If necessary, define a new table replacing target column *col-num* with a more appropriately defined column and repeat the process.

SQL1195N   **The Tools Catalog Database** *database-name* **creation or initialization failed at** *server-name*. **SQLCODE =** *SQLCODE*.

Explanation: The Tools Catalog Database *database-name* creation or initialization failed at

server-name. SQLCODE = *SQLCODE*.

User Response: Refer to the user response for the SQLCODE returned from the operation.

SQL1196W **Backup completed successfully. However, one or more Data Links File Managers could not be contacted. The timestamp for this backup image is** *backup-image-timestamp***.**

Explanation: Backup completed successfully. One or more Data Links File Managers were not available during backup processing. Additional verification are needed from the Data Links File Managers when they are available.

User Response: Bring up the Data Links File Managers that were unavailable during backup processing. Once all the Data Links File Managers can be contacted, DB2 will automatically complete the final verification for this backup image.

SQL1197N **Invalid value specified for keyword** *keyword* **in command** *command***.**

Explanation: The *keyword* value specified for the *command* is invalid. For numeric values, the value may be outside a defined range. For other types, the value specified is not in the defined set of valid values.

User Response: Determine from reference documentation for the *command* what are valid values and make the appropriate change. If this error has been returned by an API, determine from the reference documentation for the *command* API what are valid values for the corresponding API option and make the appropriate change.

SQL1198N **This command is not supported in the current downlevel client-server configuration. Reason code =** *reason-code***.**

Explanation: The command you have entered is not supported in your current configuration involving either a pre-V8 client or server. The possible reason codes are as follows:

1   Scrollable cursors are not supported from a pre-V8 client through a V8 or later gateway to a DB2 UDB server.

2   Utility commands are not supported from a pre-V8 client through a V8 or later gateway to any DB2 server.

3   Access to a pre-V8 server through a V8 or later gateway is not supported.

4   Utility commands are not supported to a pre-V8 server from a V8 or later client.

98   Operation not supported to a pre-V8 server from a V8 or later client.

99   Operation not supported from a pre-V8 client to a V8 or later server.

User Response: The action is based on the reason code as follows:

1   Do not use scrollable cursors from a downlevel client through a gateway. Use a direct client-server connection or upgrade your client to a compatible release.

2   Do not issue utility commands from a downlevel client through a gateway. Use a direct client-server connection or upgrade your client to a compatible release.

3   Use a compatible level of gateway or configure a direct connection (non-gateway) to the server.

4   Use a compatible client level to issue V8 utility commands.

98   Use a compatible level of client to issue this command or consider upgrading your server's level of code.

99   Use a compatible level of server to issue this command or consider upgrading your client's level of code.

SQL1200 - SQL1299

SQL1200N   The object parameter is not valid.

Explanation: 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.

User Response: Correct the object parameter and resubmit the COLLECT DATABASE STATUS function call.

SQL1201N   The status parameter is not valid.

Explanation: 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.

User Response: Correct the status parameter and resubmit the COLLECT DATABASE STATUS function call.

SQL1202N   Status has not been collected.

Explanation: 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.

User Response: Correct the *handle* parameter and resubmit the COLLECT DATABASE STATUS function call.

SQL1203N   The database has no users connected.

Explanation: A request was made for user status on a database, but the database has no users connected.

The command cannot be processed.

User Response: 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 territory code *territory-code* is not supported by the installed version of the database manager.

Explanation: This version of the database manager does not support the active code page or territory code or both.

The command cannot be processed.

Choose an active code page and territory code supported by this version of the database manager.

Federated system users: if necessary isolate the problem to the data source rejecting the request (see the Troubleshooting Guide for procedures to follow to identify the failing data source) and choose an active code page and territory code supported by both the federated server and the data source.

User Response: Exit the current program and return to the operating system.

sqlcode: -1204

sqlstate: 22522

SQL1205N **The code page** *code-page* **and/or territory code** *territory* **that has been specified is not valid.**

Explanation: This version of DB2 does not support the active code page or territory code or both, as specified by the Create Database command.

The command cannot be processed.

User Response: Refer to the Create Database command in the *Command Reference* for details of valid code page and territory code pairs supported by DB2/2.

SQL1206N **PRUNE LOGFILE is not supported in this database configuration.**

Explanation: The PRUNE LOGFILE request is not supported if:
1. both LOGRETAIN and USEREXIT are set to NO, or
2. the active logfile path is set to a raw device.

User Response: Do not issue the PRUNE LOGFILE command for this database.

SQL1207N **Communication Manager configuration file** *name* **cannot be found.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: No action is required.

SQL1210W **Default values were returned for one or more DOS Requester/WINDOWS Requester configuration file parameters.**

Explanation: 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.

User Response: Ensure that the DOS Requester/WINDOWS Requester configuration file exists in the appropriate path and that the parameters are explicitly defined.

SQL1211N **The computer name** *name* **is not valid.**

Explanation: The computer name specified in the NPIPE protocol structure for the Catalog command is not valid. The size of the computer name must be 15 characters or less.

The command cannot be processed.

User Response: Verify that the computer name is valid and resubmit the command.

SQL1212N **The instance name** *name* **is not valid.**

Explanation: The instance name specified for the Catalog command is not valid. The size of

the instance name must be 8 characters or less.

The command cannot be processed.

User Response: Verify that the instance name is valid and resubmit the command.

SQL1213N **The change password LU name** *name* **is not valid.**

Explanation: The change password logical unit (LU) name specified in the APPN protocol structure for the CATALOG command is not valid.

The change password LU name is the remote SNA LU 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.

User Response: Verify the characters specified in the change password LU name.

Resubmit the command with a valid change password LU name.

SQL1214N **The transaction program name** *name* **is not valid.**

Explanation: The transaction program (TP) name specified in the APPN protocol structure for the CATALOG command is not valid.

The TP name is the remote SNA Application TP name and must be 1 to 64 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 $.

The command cannot be processed.

User Response: Verify the characters specified in the TP name.

Resubmit the command with a TP name.

SQL1215N **The LAN adapter address** *address* **is not valid.**

Explanation: The LAN adapter address specified in the APPN protocol structure for the CATALOG command is not valid.

The LAN adapter address is the remote SNA LAN adapter address and must be a 12-hexadecimal number.

The command cannot be processed.

User Response: Verify the LAN adapter address.

Resubmit the command with a valid LAN adapter address.

SQL1216N **Graphic data and graphic functions are not supported for this database.**

Explanation: 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.

Federated system users: this situation can also be detected by the data source.

User Response: Resubmit the command with valid data types.

sqlcode: -1216

sqlstate: 56031

SQL1217N **The REAL data type is not supported by the target database.**

Explanation: An SQL operation is using a data type of REAL (single-precision floating point number) as an input or output variable. The REAL data type is not supported on the target database for this request.

The statement is not processed.

User Response: Replace the declaration of any host variable that corresponds to the SQL data

type of REAL with a declaration that corresponds to an SQL data type of DOUBLE in your application.

sqlcode: -1217

sqlstate: 56099

SQL1218N   There are no pages currently available in bufferpool *buffpool-num***.**

Explanation: All of the pages in the bufferpool are currently being used. A request to use another page failed.

The statement cannot be processed.

User Response: The bufferpool is not large enough to provide pages to all database processes or threads at this time. The bufferpool is too small or there are too many active processes or threads.

The statement may be successful if executed again. If this error occurs frequently, some or all of the following actions may prevent further failures:

1. increase the bufferpool size
2. decrease the maximum number of database agents and/or connections
3. decrease the maximum degree of parallelism
4. decrease the prefetch size for table spaces that are in this bufferpool
5. move some table spaces into other bufferpools.

sqlcode: -1218

sqlstate: 57011

SQL1219N   The request failed because private virtual memory could not be allocated.

Explanation: 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.

User Response: 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.

Explanation: 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

User Response: 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 **The Application Support Layer heap cannot be allocated.**

Explanation: 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

User Response: 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.**

Explanation: All available memory in the Application Support Layer heap has been used.

User Response: Increase the *aslheapsz* configuration parameter.

sqlcode: -1222

sqlstate: 57011

SQL1223N **An agent could not be started to service this request.**

Explanation: The request failed because the *maxagents* configuration parameter limit would be exceeded.

User Response: 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.**

Explanation: 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.
- A user with SYSADM authority issued a FORCE QUIESCE command while you were connected. Because your user ID does not have CONNECT QUIESCE authority for the database/instance or belong to a group that has CONNECT QUIESCE authority, you were disconnected from the database/instance.

Additional federated server 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 federated server or by the data source.

User Response:

- 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.

- Have the SYSADM unquiesce the database/instance or add the user to the quiesced group.

Federated system users, should also:

- Isolate the problem to the data source rejecting the request (see the *Troubleshooting 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 federated 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 federated server, use the command:
```
ps -ef | grep instdj1 | wc -l
```

where "instdj1" is the instance name of the federated 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). For users on AIX version 4.2.1 or newer, the environment variable EXTSHM can be set to ON to increase the number of shared memory segments to which a single process can be attached.

sqlcode: -1224

sqlstate: 55032

SQL1225N   The request failed because an operating system process, thread, or swap space limit was reached.

Explanation: An operating system process, thread, or swap space limit was reached. You can find more information about the problem in the administration notification log. On AIX systems, the maxuproc value might be too low. On OS/2-based systems, the THREADS CONFIG.SYS value might be too low.

User Response: Check the administration notification log to determine which limit was reached and increase that limit.

sqlcode: -1225

sqlstate: 57049

SQL1226N   The maximum number of client connections are already started.

Explanation: The number of client connections is already equal to the maximum value defined in the system configuration file.

The operation cannot be processed.

User Response: Wait for other applications to disconnect from the database. If one or more applications are required to run concurrently, increase the value for max_connections. The new value takes effect after the next successful database manager start. If the value for max_connections is larger than the value for max_coordagents, the concentrator feature is turned on.

sqlcode: -1226

sqlstate: 57030

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*

Explanation: The value or format of a statistic provided for an updatable 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 not 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) or column group (COLGROUPCARD in SYSCOLGROUPS) cannot be greater than the cardinality of its corresponding table (CARD statistic in SYSTABLES).

7   No statistics are supported for columns with the following data types: LONG

VARCHAR, LONG VARGRAPHIC, BLOB, CLOB, DBCLOB, or user-defined structured types.

8   A statistic is inconsistent with other related statistics for this entity or is invalid in this context.

User Response: 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

SQL1228W   DROP DATABASE has completed but the database alias name or database name *name* **could not be found on** *num* **nodes.**

Explanation: The drop database command has completed successfully, however, there are some nodes where the database alias or database name was not found. It is possible that DROP DATABASE AT NODE was already performed on these nodes.

User Response: This is a warning message only. No response is necessary.

SQL1229N   The current transaction has been rolled back because of a system error.

Explanation: One of the following has occurred:

1. A system error, such as node failure or connection failure, has occurred. The application is rolled back to the previous COMMIT.

    Note that in the case of DB2 utility functions, the behavior is described below:

    Import The application is rolled back. If the COMMITCOUNT parameter was used, the operation is rolled back to a previous committed point.

Reorg   The operation is aborted and must be resubmitted.

Redistribute
       The operation is aborted, however, some of the operation may have been successful. Issuing the request again with the "continue" option will restart the operation from the point of failure.

Rollforward
       The operation is aborted and the database is still in rollforward pending state. The command must be resubmitted.

Backup/Restore
       The operation is aborted and must be resubmitted.

2. The service port numbers assigned for FCM (Fast Communication Manager) communications are not the same on all nodes in the DB2 instance. Check the services file being used on all nodes to ensure that the ports are the same. The ports are defined using the format:

```
DB2_<instance>        xxxx/tcp
DB2_<instance>_END    xxxx/tcp
```

where <instance> is the DB2 instance name and xxxx is the port number Ensure that these port numbers are not being used for DB2 remote client support.

User Response:

1. Try the request again. If the error persists, you can find more information about the problem in the administration notification log. It may be necessary to contact the system administrator for assistance since the most common reason for this error is that a node has failed.

    Note that in an SP environment where the high speed switch is used, this error can be a symptom of a failure in the high speed switch.

2. Update the service ports to be the same on all nodes and try the request again.

The sixth sqlerrd field of the SQLCA will indicate the node number that detected the node failure. On the node that detected the failure a message will be placed in the db2diag.log that identifies the failed node.

sqlcode: -1229

sqlstate: 40504

SQL1230W   At least one agent id specified could not be forced.

Explanation: 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.
- The agent id specified was for a parallel agent which cannot be forced.

User Response: If a nonexistent or invalid agent id was specified, retry the command using the valid agent id.

SQL1231N   Invalid Force count specified.

Explanation: 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.

User Response: Correct the value of count and resubmit the command.

SQL1232N   Invalid Force mode specified.

Explanation: 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.

User Response: Set the mode to SQL_ASYNCH and resubmit the command.

SQL1233N   The use of this clause or scalar function is supported only for graphic data that is UCS-2.

Explanation: UCS-2 is not supported for this database. The support of UCS-2 is required for the following:

- Specifying a graphic string expression as the first argument of the VARCHAR scalar function
- Specifying a character string expression as the first argument of the VARGRAPHIC scalar function when a second argument is specified
- Specifying a UCS-2 hexadecimal constant using the format UX'hex-digits'

The statement cannot be processed.

User Response: Resubmit the command with valid data types.

sqlcode: -1233

sqlstate: 560AA

SQL1240N   The maximum number of quiescers has been reached.

Explanation: An attempt was made to acquire a quiesce state on a table space which has already been quiesced by 5 processes.

User Response: 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*.

Explanation: The value for a table space attribute was out of range. See the Data Structures section of Application Development Guide for the format of the sqletsdesc structure used for the create database api. The identified attribute is the field name of this structure.

User Response: Correct the create database request.

SQL1244W **Disconnect for transaction manager database** *server-name* **will occur at next COMMIT.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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 **No data returned for heuristic query.**

Explanation: There are no in-doubt transactions in the database, nor any transactions that have ended and are waiting to enter the syncpoint process.

User Response: No action is required.

SQL1259N **The ARCHIVE LOG command failed for database** *db-name* **on database partition** *partition-list* **due to SQLCODE** *SQLCODE*.

Explanation: An SQL error occurred while attempting to issue the ARCHIVE LOG command for database *db-name* on database partition *partition-list*.

User Response: Refer to the SQLCODE in the Message Reference for the user response and resubmit the ARCHIVE LOG command.

SQL1260N **Database** *name* **is not configured for roll-forward recovery on node(s)** *node-list*.

Explanation: The specified database is not configured for roll-forward recovery on the specified node(s). If ",..." is displayed at the end of the node list, see the administration notification log for the complete list of nodes.

The database is not rolled forward on the specified node(s).

(Note : if you are using a partitioned database server, the node number(s) indicate which

node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: Ensure that recovery is required on the specified node(s), then restore the most recent backup version of the database on these nodes.

SQL1261N **Database** *name* **is not in rollforward pending state on node(s)** *node-list*, **so it does not need to be rolled forward on these nodes.**

Explanation: The specified database is not in roll-forward pending state on the specified node(s). This may be because the database has not been restored, or was restored with the WITHOUT ROLLING FORWARD option, or roll-forward recovery is complete on these nodes.

If ",..." is displayed at the end of the node list, see the administration notification log for the complete list of nodes.

The database is not rolled forward.

(Note : if you are using a partitioned database server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: Do the following:
1. Ensure that recovery is required on the specified node(s).
2. Restore a backup version of the database on these nodes.
3. Issue the ROLLFORWARD DATABASE command.

SQL1262N **The point-in-time specified for rolling forward database** *name* **is not valid.**

Explanation: 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

sssss represents optional microseconds).

The database is not rolled forward.

User Response: 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 for database** *name* **on node** *node-number***.**

Explanation: The specified archive log file exists in the database log directory or the overflow log directory on the specified node, but the file is not valid.

Roll-forward recovery processing stops.

(Note : if you are using a partitioned database server, the node number indicates which node the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: To determine the correct archive log file, issue the ROLLFORWARD DATABASE 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 issue the ROLLFORWARD DATABASE command again. 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* **on node** *node-number***.**

Explanation: The archive log file specified was found in the log directory or the overflow log directory but does not belong to the specified database.

Roll-forward recovery processing stops.

(Note : if you are using a partitioned database server, the node number indicates which node the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: To determine the correct archive log file, issue the ROLLFORWARD DATABASE 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 issue the ROLLFORWARD DATABASE command again. 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 database** *name* **on node** *node-number***.**

Explanation: 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. The specified archive log file was found in the log directory or the overflow log path for the database on the specified node but the log file is not in the correct log sequence.

Roll-forward recovery processing stops.

(Note : if you are using a partitioned database server, the node number indicates which node the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: To determine the correct archive log file, issue the ROLLFORWARD DATABASE 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 issue the ROLLFORWARD DATABASE command again. 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.**

Explanation: Roll-forward is requested to stop at a timestamp that is before the current point in time in the specified database log file. This can happen when rolling forward either the database or a subset of table spaces to a point in time.

Database roll-forward processing stops.

User Response: Specify the correct point in time or restore the database or table space subset from a backup version and resubmit the ROLLFORWARD command.

SQL1267N **The system could not find db2uexit in the current PATH environment variable.**

Explanation: 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.

User Response: 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* **for database** *name* **on node** *node-number*.

Explanation: Roll-forward processing invokes *db2uext2* to retrieve the log file. The error may have occurred in *db2uext2*.

Roll-forward processing stopped. The database or table space is still in a roll forward pending state.

(Note : if you are using a partitioned database server, the node number indicates which node the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: 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.**

Explanation: 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.

User Response: 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 **The LANG environmental variable is set to** *string*. **This language is not supported.**

Explanation: The LANG environmental variable is set to a language that is not supported by the database manager. Processing cannot continue.

User Response: 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*.

Federated system users: see the *Federated Systems Guide* for NLS information.

SQL1271W **Database** *name* **is recovered but one or more table spaces are off-line on node(s)** *node-list*.

Explanation: This message can occur after crash recovery, database level roll-forward recovery, or table space level roll-forward recovery. For database level roll-forward recovery, the database is available for use if the STOP option was specified. One or more table spaces on the specified node(s) are not available. This could happen if:

- the STOP option is not specified for a table space level roll-forward recovery to a point in time.

- one or more table spaces receive errors during recovery.

- additional table spaces are restored while a table space level roll-forward recovery is already in progress.
- the database is rolled forward past a previous point-in-time table space roll-forward, causing all table spaces involved to be placed in restore pending state.

The table spaces that are off-line can be identified on the specified node(s) by the LIST TABLESPACES command or the db2dart utility. The administration notification log will provide more information on specific table spaces.

If ",..." is displayed at the end of the node list, see the administration notification log for the complete list of nodes.

(Note : if you are using a partitioned database server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: Repair or restore the table spaces if necessary and run roll-forward recovery. If the same error occurs, try running table space level roll-forward recovery offline.

SQL1272N **Table space level roll-forward recovery for database** *name* **has stopped before completion on node(s)** *node-list*.

Explanation: Table space level roll-forward recovery has stopped on the specified node(s) before all qualifying table spaces could be rolled forward. This can be cause by one of the following:

- The transaction table is full.
- All table spaces being rolled forward received I/O errors.
- Point-in-time table space level roll-forward encountered an I/O error on one of the table spaces being rolled forward.
- Point-in-time table space level roll-forward encountered an active transaction that made changes to one of the table spaces being rolled forward. This transaction could be an indoubt transaction.

- The table space level roll-forward was interrupted and, before it was resumed, all table spaces that were being rolled forward are restored again.

If ",..." is displayed at the end of the node list, see the administration notification log for the complete list of nodes.

(Note : if you are using a partitioned database server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: Check the administration notification log for the cause. Do one of the following depending on the cause:

- Use the LIST TABLESPACES command to determine if the table spaces received I/O errors. If so, repair them.
- If the transaction table became full, include the MAXAPPLS database configuration parameter or try running table space level roll-forward recovery offline.
- If the cause is an active or indoubt transaction, complete the transaction.
- If the table spaces were restored after a previous table space level roll forward was interrupted, the previous table space level roll-forward is now canceled. The next table space level roll-forward command will check for table spaces in rollforward pending state.

Run table space level roll-forward recovery again.

SQL1273N **Roll-forward recovery on database** *name* **cannot reach the specified stop point (end-of-log or point-in-time) because of missing log file** *name* **on node** *node-number*.

Explanation: The Rollforward Database utility cannot find the specified archive log file in the database log directory or the overflow log directory on the specified node.

Roll-forward recovery has stopped.

(Note : if you are using a partitioned database

server, the node number indicates which node the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: Do one of the following:

- 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 issue the ROLLFORWARD DATABASE command again. Alternatively, resubmit the command with the overflow log path pointing to the correct archive file.

- If you cannot find the missing log files, restore the database/table space on all nodes, then do point-in-time recovery using a timestamp that is earlier than that of the missing log file.

SQL1274N   **The database *name* requires roll-forward recovery and the point-in-time must be to the end of logs.**

Explanation: The database needs to be rolled forward. For database level roll-forward recovery, the point-in-time must be to the end of logs because a database level roll-forward to end of logs is already in progress. To continue the roll forward, the same stoptime must be specified.

For table space level roll-forward recovery, the point-in-time must be to the end of logs for one of the following reasons:

- The system catalogs require roll-forward recovery. System catalogs must always be rolled forward to the end of logs to remain consistent with all other table spaces.

- A table space level roll-forward to the end of logs is already in progress. To continue the roll forward, the same stoptime must be specified.

The database is not rolled forward.

User Response: 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*, because database *name* on node(s) *node-list* contains information later than the specified time.**

Explanation: The database or at least one of the table spaces to be rolled forward were backed up online. An advanced virtual timestamp is in the database or table space backup. The stoptime passed to roll-forward must be greater than or equal to the time the online backup ended on the specified node.

If ",..." is displayed at the end of the node list, see the administration notification log for the complete list of nodes.

(Note : if you are using a partitioned database server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: Do one of the following:

- Resubmit the command with a stoptime greater than or equal to *timestamp*. Time must be specified in CUT (Coordinated Universal Time).

- Restore an earlier backup on node(s), then issue the ROLLFORWARD DATABASE command again.

SQL1276N   **Database *name* cannot be brought out of rollforward pending state until roll-forward has passed a point in time greater than or equal to *timestamp*, because node *node-number* contains information later than the specified time.**

Explanation: A request was made to bring the database or subset of table spaces out of rollforward pending state by specifying caller action SQLUM_ROLLFWD_STOP, SQLUM_STOP, SQLUM_ROLLFWD_COMPLETE, or SQLUM_COMPLETE. However, the database or at least one of the table spaces to be rolled forward was backed up online. The request cannot be granted until the database or all table spaces have been rolled forward to the end of

the online backup timestamp on the specified node.

This error can also occur if not all the log files are provided to perform the requested recovery.

(Note : if you are using a partitioned database server, the node number indicates which node the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: If the stoptime specified on the ROLLFORWARD command is smaller then *timestamp*, resubmit the command with a stoptime greater than or equal to *timestamp*.

Verify that all the log files were provided. the ROLLFORWARD QUERY STATUS command shows which log file is to be processed next. Some of the reasons for missing log files include:

- the log path has changed. The file can be found in the old log path.
- the userexit program is not able to retrieve the log file, because it may have been archived to a different location.

If the missing log file is found, copy it to the log path and resubmit the command.

SQL1277N   Restore has detected that one or more table space containers are inaccessible, or has set their state to ″storage must be defined″.

Explanation: Restore validates that the containers needed by each table space 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 table space, 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 table spaces 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.

User Response: To determine the list of

containers for each table space being restored, use the TABLESPACE CONTAINER QUERY api or the LIST TABLESPACE CONTAINERS command. To specify an updated list for each table space, 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.

SQL1278W   Roll-forward operation has completed successfully. Active or indoubt transactions required rollback on node(s) *node-list*.

Explanation: The roll-forward of a table space subset to a point in time was completed successfully, however one or both of the following situations occurred:

1. There was one or more active transactions at the point in time specified. Each transaction was rolled back from the table spaces in the table space subset.

2. There was one or more indoubt transactions at the point in time specified. Each indoubt transaction was rolled back from the table spaces in the table space subset.

Transactions that were rolled back from the table spaces in the table space subset may still be committed in other table spaces not involved in the roll-forward.

If ″,...″ is displayed at the end of the node list, see the administration notification log for the complete list of nodes.

(Note : if you are using a partitioned database server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: The administration notification log will contain details on the transactions that were rolled back by roll-forward recovery.

SQL1279W **Some indexes may not have been recreated.**

Explanation: An error occurred during index recreation while performing a database restart or following a table reorganization which prevented some of the indexes from being recreated successfully. Details can be found in administration notification log.

Database restart or Reorg table was successful.

User Response: Examine the administration notification log to determine why the index(es) could not be recreated and correct the problem. The invalid indexes for a table will be recreated when the table is first accessed.

SQL1280N **The stoptime passed to roll-forward must be less than or equal to** *timestamp* **for database** *name* **because at least one of the table spaces was already rolled forward to this point in time.**

Explanation: At least one of the table spaces specified by the roll-forward to point in time has already been rolled forward to an earlier point in time. It cannot be rolled forward any further.

User Response: Do one of the following:

- Resubmit the command with stoptime *timestamp*.
- Restore all table spaces again, and resubmit the command with a stoptime less than *timestamp*.
- Restore the table space backup(s) taken following the earlier table space roll-forward to point in time and resubmit the command with the same stoptime.

Time must be specified in CUT (Coordinated Universal Time).

SQL1281N **The connection to database** *database-alias* **has been severed because the pipe** *pipe-name* **has been broken.**

Explanation: The connection was lost because the DB2 server broke the pipe. The current transaction has been rolled back.

User Response: 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. 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.**

Explanation: 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.

User Response: 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   **The attempt to connect to database** *database-alias* **failed because the pipe** *pipe-name* **is in use by another process.**

Explanation: The name of the named pipe is already in used by another process. Named pipe support did not start.

User Response: 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.**

Explanation: The server has not started its named pipe support, or the server is using a different name for the named pipe.

User Response: 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.**

Explanation: The alternate pipe name set by the environment variable DB2PIPENAME is invalid.

User Response: 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***.**

Explanation: Named pipe failed because the operating system ran out of resources (swapping space, disk space, file handles). The current transaction has been rolled back.

User Response: 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.**

Explanation: The server has not started its named pipe support, or the instance name is incorrect.

User Response: 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.

SQL1288N   **The remote server does not support this operation.**

Explanation: The remote server is running on a 64-bit platform that does not support non-SQL requests from Version 7 (or earlier) 32-bit clients.

User Response: Perform this operation from a supported client.

SQL1290N   **The value of either the DFT_CLIENT_COMM database manager configuration parameter or the DB2CLIENTCOMM environment variable is not valid.**

Explanation: Either an incorrect value was specified or the protocol(s) specified are not supported by the target database. The acceptable values are:

• Unix platforms: TCPIP and APPC

- OS/2: TCPIP, APPC, IPXSPX, and NETBIOS
- Windows: TCPIP, APPC (Windows 32-bit only), NETBIOS, and NPIPE

If more than one value is 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 DB2 Connect gateway and your client workstation does not use global directory services, this message may be returned from the DB2 Connect gateway.

User Response: 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***.**

Explanation: 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 DCE include file xds.h.

For XOM functions such as om_get, the return code values can be found in the DCE include file xom.h.

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 DB2 Connect gateway and your client workstation does not use global directory services, this message may be returned from the DB2 Connect gateway.

User Response: 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.**

Explanation: 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 example, if you are trying to connect to a DRDA server via a DB2 Connect gateway and your client workstation does not use global directory services, this message may be returned from the DB2 Connect gateway.

User Response: Correct the global name and try again.

SQL1293N   **An error was found in a global directory entry. Error code:** *error-code***.**

Explanation: An error has been detected in one of the global database directory entries used. Refer to the following error codes for details:

1   The database object does not contain authentication information

2   Both the database object and the database locator object do not contain communication protocol information

10   Entry not a database object

11   Native database name in the database object is either not found or too long

12   Database protocol in the database object is either not found or too long

13   Invalid authentication value found in the database object

14   Insufficient or invalid communication protocol information in the database object

15   Database locator object name not found in the database object

16   Database locator object name in the database object invalid

20   Entry not a database locator object

22   Insufficient or invalid communication protocol information in the database locator object

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 DB2 Connect gateway and your client workstation does not use global directory services, this message may be returned from the DB2 Connect gateway.

User Response: 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.**

Explanation: 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 DB2 Connect gateway and your client workstation does not use global directory services, this message may be returned from the DB2 Connect gateway.

User Response: 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.

Explanation: 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 DB2 Connect gateway and your client workstation does not use global directory services, this message may be returned from the DB2 Connect gateway.

User Response: 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.

Explanation: 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 "/.:/".

User Response: 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.

Explanation: The function requested using this command is not supported on this platform.

User Response: Do not use this command.

SQL1300 - SQL1399

SQL1300N   Invalid DCE principal name in catalog statement.

Explanation: The DCE principal name in the catalog database operation was invalid. DCE

principal names are subject to the following conditions:

- If AUTHENTICATION is specified as DCE, a principal name must be included in the catalog statement.

- If AUTHENTICATION is not specified as DCE, a principal name can not be included in the catalog statement.
- The maximum length of the principal name is 1024 bytes.

User Response: Ensure that the principal name conforms to the conditions above, and resubmit the catalog command.

SQL1301N   An error occurred when accessing the server's DCE keytab file.

Explanation: An error occurred when accessing the server's DCE keytab file. For a keytab file to be valid, the following conditions must be satisfied:

- The server's keytab file exists, has the name keytab.db2, and is in the sqllib/security directory.
- There is only a single entry in the keytab file.

User Response: Make sure that DCE is started. Next, check that the keytab file exists, and contains a single entry (with rgy_edit). Retry the operation.

SQL1302N   DCE principal to DB2 authorization id mapping error. Reason code: *reason-code*.

Explanation: An error occurred when mapping a DCE principal to a DB2 authorization id. Refer to the reason code below.

- 1. Missing or invalid DCE user to DB2 authorization id mapping
- 2. Missing or invalid DCE group to DB2 authorization id mapping

User Response: The DCE principal must have an ERA mapping to a DB2 authorization id. Add the missing entry to the DCE registry, and retry the operation.

sqlcode: -1302

sqlstate: 08001

SQL1303N   Security daemon could not be restarted.

Explanation: An attempt to restart the security daemon has failed after communication between the agents and the security daemon was severed or the security daemon terminated abnormally. All new connections to the database manager will be rejected, as authentication is not possible.

User Response: Stop the database manager and start it again. If the db2start command fails, follow the user response for that sqlcode.

sqlcode: -1303

sqlstate: 58004

SQL1304N   The TCP/IP security type SOCKS is not valid.

Explanation: The TCP/IP security type SOCKS in the TCP/IP protocol structure of the Catalog Node command is invalid with authentication type DCE.

User Response: Make sure you do not use the combination of TCP/IP protocol with security type SOCKS and authentication type DCE.

sqlcode: -1304

sqlstate: 08001

SQL1305N   An internal DCE error occurred.

Explanation: DB2 processing failed due to an internal DCE error.

User Response: Make sure DCE is started. If the problem persists, contact a service representative for assistance.

sqlcode: -1305

sqlstate: 58004

SQL1306N   An invalid parameter was specified during invocation of the security audit facility. Reason Code: *reason-code*.

Explanation: One of the parameters of the security audit API is incorrect. Reasons:

1   An invalid audit option was specified.

2   Invalid pointer to Configure/Describe sqleaucfg structure.

3   Invalid Configure/Describe parameter token.

4   Invalid Configure/Describe parameter value. The value is incorrect or outside the valid range for the parameter.

5   Invalid count specification for Configure/Describe parameter.

6   Insufficient length allocated for Configure/Describe parameter.

7   Invalid pointer to Extract sqleauextract structure.

8   Invalid Extract parameter token.

9   Invalid Extract parameter value. The value is incorrect or outside the valid range.

10   Invalid count specification for Extract parameter.

11   Invalid length for Extract parameter.

User Response: The system administrator should take specific actions for each of the reasons.

1   Provide the right option value for the audit API call by looking at the sqlutil.h include file.

2   Check to see if a valid pointer has been provided to a Configure/Describe structure.

3   The correct parameter should be specified by looking up the audit facility reference section for the sqleaucfg parameter tokens.

4   Correct the parameter value by looking up the valid values in the audit facility reference.

5   Specify the correct count and allocate/initialize the appropriate length for variable length parameters.

6   Correct the length allocated for

Configure/Describe parameter based on the error token that is returned in the SQLCA.

7   Check to see if a valid pointer has been provided to a Extract structure.

8   The correct parameter should be specified by looking up the audit facility reference section for the sqleauextract parameter tokens.

9   Correct the parameter value by looking up the valid values in the audit facility reference.

10   Specify the correct count and allocate/initialize the appropriate length for variable length parameters.

11   Correct the length allocated for Extract parameter based on the error token that is returned in the SQLCA.

SQL1307N   **An error occurred during invocation of the security audit facility. Reason Code:** *reason-code***.**

Explanation: The invocation of the security audit API caused an error to occur. Reasons:

1. Audit is already started.
2. Audit is already stopped.
3. Invalid checksum for audit configuration file.
4. The default or user-provided audit pathname is too long.
5. Could not update the audit configuration file. Either the filesystem is full or the permissions do not allow write.
6. Configuration File not found. Either the file or the directory which contains the file does not exist.
7. The extract file is not found.
8. Invalid format of audit records during extract. The file is corrupted.

User Response: The system administrator should take specific actions for each of the reasons.

1. No action required.
2. No action required.

3. Restore the configuration file from backup or issue 'audit reset' command.

4. Choose a different audit pathname which is within the filename length limit.

5. If the file permissions are incorrect,set the permissions to allow write by owner. If the filesystem is full, create free space before proceeding.

6. If the audit configuration file is missing, then restore it from a backup or issue the 'reset' command to initialize the file to defaults. If the directory is missing, restore from a backup or recreate the database manager instance.

7. Verify that file exists in the specified path. If file missing, restore from backup if available.

8. The audit log file is most likely corrupted. If problem persists for other audit log files, notify DB2 service.

SQL1308W   **The audit Extract Facility completed Processing.** *num-records* **records were extracted.**

Explanation: The security audit extract facility completed its processing succesfully and extracted the specified number of records.

User Response: If zero records were extracted, the user should verify that the extract pathname contains extract files and the extract parameters are correct.

SQL1309N   **Invalid server principal name.**

Explanation: The server principal name specified in the database catalog statement does not exist in the DCE registry. Because of this a DCE ticket could not be obtained for the DB2 server.

User Response: Make sure that the principal name in the database catalog entry corresponds to the DCE principal being used by the DB2 server. It may be necessary to fully qualify the principal name.

sqlcode: -1309

sqlstate: 08001

SQL1310N   **The Database Connection Services directory service failed while attempting to access the Database Connection Services directory.**

Explanation: 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.

User Response: 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.**

Explanation: The directory cannot be found. The directory may have been deleted.

The function cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Ensure that the application program has allocated the required buffer area and resubmit the command.

SQL1315N **The local database name is not valid.**

Explanation: 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.

User Response: Ensure that the characters used in the local database name are from the database manager base character set and resubmit the command.

SQL1316N **No entry was found in the Database Connection Services directory for the specified local database name.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Wait for the current activity to complete and resubmit the command.

SQL1321N   The structure ID specified in the directory entry structure is not valid.

Explanation: The structure ID received in the directory entry structure does not represent a recognized value.

The function cannot be processed.

User Response: Verify the structure ID passed in the directory entry structure is a valid value and resubmit the command.

SQL1322N   An error occurred when writing to the audit log file.

Explanation: The DB2 audit facility encountered an error when invoked to record an audit event to the trail. There is no space on the file system the audit log resides on. Free up space on this file system or prune the audit log to reduce its size.

When more space is available, use db2audit to flush out any data in memory, and to reset the auditor to a ready state. Ensure that appropriate extracts have occurred, or a copy of the log has been made prior to pruning the log, as deleted records are not recoverable.

User Response: The system administrator should take the suggested corrective action so that the audit facility can resume logging.

sqlcode: -1322

sqlstate: 58030

SQL1323N   An error occurred when accessing the audit configuration file.

Explanation: db2audit.cfg could not be opened, or was invalid. Possible reasons:

- The db2audit.cfg file either does not exist, or has been damaged. Take one of the following actions:
  - Restore from a saved version of the file.
  - Reset the audit facility configuration file by issuing the reset command from the db2audit executable.

User Response: The system administrator should take the suggested action to resolve the problem.

sqlcode: -1323

sqlstate: 57019

SQL1325N   The remote database environment does not support the command or one of the command options.

Explanation: An attempt has been made to issue a DB2 workstation database specific command or command option against a host database through DB2 Connect or federated server. 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 Action String (e.g. "REPLACE into ...") must be INSERT.
- EXPORT (Export table) The filetype must be IXF.

The command cannot be processed.

User Response: Do not attempt to issue this command against a host database through DB2 Connect or federated server.

SQL1326N   **The file or directory** *name* **cannot be accessed.**

Explanation: The file or directory *name* cannot be accessed because either the file permissions are incorrect, the file path is incorrect, or there is insufficient space in the directory or path.

User Response: Verify that any path or filename specified in the command is valid, you have the proper permissions to access the path or filename, and there is sufficient space to contain the file. 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response:

- 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.
- 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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 federated server. When DB2 Connect is used, it is only supported when authentication type is DCE, KERBEROS, or SERVER_ENCRYPT.

• 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 SERVER is used with DB2 Connect or federated server, or when authentication type is DCE, KERBEROS, or SERVER_ENCRYPT.

• SQL_CPIC_SECURITY_PROGRAM
  – Both a userid and a password will be included. This is not supported when authentication type CLIENT is used with DB2 Connect or when authentication type is DCE, KERBEROS,or SERVER_ENCRYPT.

User Response: Set the security type to one of the above and resubmit the command.

sqlcode: -1331

sqlstate: 08001

SQL1332N **The Host Name** *name* **is not valid.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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**

Explanation: An application requestor name was specified that contains characters that are not valid. All characters must be from the database manager base character set.

User Response: 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.**

Explanation: 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.

User Response: 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.

Federated system users: verify that the remote host is cataloged correctly in the SYSCAT.SERVERS view.

SQL1337N   **The service** *service-name* **was not found.**

Explanation: 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.

Federated system users: this situation can also be detected by the data source.

User Response: 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.

Federated system 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.**

Explanation: 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.
- The SNA communication subsystem has not been started.

User Response: 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.

Start the SNA communication subsystem if it has not been started.

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*.

Explanation: One or more SQL substatements within a Compound SQL statement resulted in SQL errors (negative return codes).

The error tokens will not be returned for CLI/ODBC applications. CLI/ODBC applications can use SQLGetDiagRec, SQLGetDiagField, or SQLError APIs to get more information about each error.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Specify a workstation name in the client database manager configuration file.

SQL1342N   The File Server name *name* **is missing or invalid.**

Explanation: The File Server name specified in the command/API is missing or invalid.

User Response: 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.**

Explanation: The Object Name specified in the command/API is missing or invalid.

User Response: 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***.**

Explanation: 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.

07   Federated system users: the application cannot process this command after executing SQL statements.

User Response: Corresponding to *rc*:

01   Complete the unit of work (using either COMMIT or ROLLBACK) before reissuing this command.

02-05   Issue the required call(s) to complete the utility in progress, then reissue this command.

07   Federated system 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.

Explanation: The user attempted to interrupt a process which is not interruptible.

User Response: Do not interrupt the current process.

SQL1361W   Execution time has exceeded timeout value. Do you wish to interrupt it?

Explanation: 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.

User Response: There are three choices: YES - interrupt now; NO - continue, don't prompt me; CANCEL - continue, same timeout.

SQL1362W One or more of the parameters submitted for immediate modification were not changed dynamically. Client changes will not be effective until the next time the application is started or the TERMINATE command has been issued. Server changes will not be effective until the next DB2START command.

Explanation: Some changes to the database manager configuration could not be applied immediately. For these parameters, the changes become effective only after DB2 is started. This typically occurs after a DB2START on a server and after the application restarts on a client.

User Response: To see which parameter changes took effect dynamically and which ones did not, retrieve the database manager configuration parameters and display the parameter details using the following command:

```
DB2 GET DBM CFG SHOW DETAIL
```

Changes to database manager configuration parameters can only take effect dynamically if you are attached to the instance. Not all configuration parameters support dynamic update. To see which parameters can be changed dynamically, refer to the Administration Guide.

If several parameters were submitted in a group, submit the parameters individually. In those cases where a configuration parameter could not change dynamically, do one or more of the following:

- for user applications: stop and start the application
- for CLP: TERMINATE and reconnect
- for a server: issue DB2STOP and DB2START

SQL1363W One or more of the parameters submitted for immediate modification were not changed dynamically. For these configuration parameters, all applications must disconnect from this database before the changes become effective.

Explanation: The database configuration command has been successfully processed. However, not all changes were processed right away. After the applications disconnect from the database, the first connection to the database will cause the changes to become effective.

User Response: To see which parameter changes took effect dynamically and which ones did not, retrieve the database configuration parameters and display the parameter details using the following command:

```
DB2 GET DB CFG FOR
   <database-alias> SHOW DETAIL
```

Changes to database configuration parameters can only take effect dynamically if you are connected to the database. Not all configuration parameters support dynamic update. To see which parameters can be changed dynamically, refer to the Administration Guide.

If several parameters were submitted in a group, submit the parameters individually. In those cases where a configuration parameter could not change dynamically, do one or more of the following:

- Ensure that all applications disconnect from the database and issue a db2 connect to command.
- Rebind your packages after the new configuration parameters take effect so that the new values will be used during the bind.
- Use the FLUSH PACKAGE CACHE command to invalidate dynamic statements in the SQL cache.

SQL1364W **One or more of the configuration parameters were set to AUTOMATIC in a case where the parameter does not support AUTOMATIC.**

Explanation: One or more of the configuration parameters were set to AUTOMATIC in a case where the parameter does not support AUTOMATIC.

User Response: If the parameter changes were submitted as a group, resubmit the changes individually to see which parameter changes were successful.

If only one parameter was submitted then this message indicates that the value AUTOMATIC is not supported for this parameter.

To find out which configuration parameters support the AUTOMATIC value, refer to the Administration Guide.

SQL1370N **An attempt to quiesce instance or database** *name1* **failed because instance or database** *name2* **is already quiesced by user** *username*. **Quiesce type:** *type*.

Explanation: 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.

User Response: 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.

SQL1371W **Instance or database** *name* **is already quiesced. Quiesce type:** *quiesce-type*.

Explanation: An attempt was made to quiesce an instance or database which is already quiesced.

User Response: No action required.

SQL1372N **Cannot perform a quiesce while in a transaction.**

Explanation: 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.

User Response: Complete the unit of work (COMMIT or ROLLBACK) and retry the request.

SQL1373W **Cannot unquiesce instance or database** *name*, **because it is not quiesced.**

Explanation: Unquiesce failed because the instance or database is not quiesced.

User Response: Ensure that the unquiesce is being issued for the correct instance or database.

SQL1374N **Cannot QUIESCE RESET instance or database** *name*, **because it has been quiesced by another user** *username*.

Explanation: The instance or database is quiesced, but you are not the quiescer.

User Response: Ensure that the quiesce reset is being issued for the correct instance or database.

SQL1375N **An invalid parameter was passed to the api. Parameter** *parm-code* **is in error.**

Explanation: *parm-code* indicates the parameter in error:

1 scope

2 options

The value may be out of range or invalid.

User Response: Check the syntax of the api, correct the parameter and try again.

**SQL1380N   An unexpected Kerberos security
error has occurred.**

Explanation: An unexpected Kerberos security
error occurred during authentication.

User Response: No action required.

**SQL1381N   The Security Support Provider
Interface is not available.**

Explanation: Authentication failed because the
Security Support Provider Interface (SSPI) was
not available.

User Response: When running under Windows
operating system, ensure that the file security.dll
exists under the system directory. Also ensure
that SSPI is supported for the operating system
that is being used.

SQL1382N   Kerberos support is not available.

Explanation: Authentication failed because
Kerberos support is not installed.

User Response: Ensure that Kerberos support is
installed and operational, then retry the
connection.

**SQL1383N   The target principal name is not
valid.**

Explanation: The target principal name that was
specified in the CATALOG DATABASE
command is not valid.

User Response: Remove the database entry that
has the invalid target principal name by using
the UNCATALOG DATABASE command.
Re-catalog the database entry with a valid target
principal name using the CATALOG DATABASE
command and retry the connection.

When running in a Windows operating system
environment, the target principal name is the
logon account name of the DB2 service in the
format of <domain name>\<user ID>.

**SQL1384N   Unable to complete mutual
authentication.**

Explanation: The connection failed because
either the client or server was unable to complete
mutual authentication.

User Response: If the target principal name was
specified in the CATALOG DATABASE
command, ensure that the target principal name
is valid for the server to which the client is
attempting to connect or attach.

When running in a Windows operating system
environment, the target principal name is the
logon account name of the DB2 service in the
format of <domain name>\<user ID>.

If the target principal name appears to be
invalid, then contact IBM service for assistance.

**SQL1390C   The environment variable
DB2INSTANCE is not defined or
is invalid.**

Explanation: The environment variable
DB2INSTANCE is not defined or is not set to a
valid instance owner.

User Response: 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*.

Federated system users: refer to the *Federated
Systems 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).

**SQL1391N   The database is already in use by
another instance.**

Explanation: 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.

User Response:
- 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

SQL1392N   Multiple instances of applications using *prep,-bind,-import,-export* **is not supported.**

Explanation: Only 1 instance of prep, bind, import, or export can be running at any one time within WINDOWS.

User Response: Within WINDOWS, do not attempt to start more than 1 application that uses prep, bind, import, or export.

SQL1393C   The environment variable DB2PATH is not defined or is invalid.

Explanation: The environment variable DB2PATH is not defined or is not set to a valid directory path.

User Response: Set the DB2PATH environment variable to the directory where the database manager is installed.

SQL1394N   The instance is not defined.

Explanation: The new instance can not be set for the application because the instance is not defined.

User Response: Ensure that the specified instance exists. Use the db2ilist command to display the list of instances:

    db2ilist

SQL1395N   Unable to switch to another instance because the application is using multiple contexts.

Explanation: The request to switch to another instance failed because the application is using multiple contexts.

User Response: Ensure that the application is not using multiple contexts before switching to another instance.

SQL1396N   Unable to switch to another instance because the application is connected to a database or attached to an instance.

Explanation: Request to switch to another instance failed because the application is connected to a database or attached to an instance.

User Response: Ensure that the application is not connected to any database or attached to any instance before switching to another instance.

SQL1397N   The DB2 service failed to logon.

Explanation: DB2 service did not start due to a logon failure.

User Response: If you are starting the DB2 Administration Server, then use the DB2ADMIN SETID command to set a new logon account. If you are starting a DB2 server on Windows NT, you can set the logon account for a DB2 service by using the Services dialog box from the Control Panel.

SQL1400 - SQL1499

SQL1400N **Authentication is not supported.**

Explanation: The authentication type supplied is not one of the supported types.

The command can not be processed.

User Response: The user should reissue the command with a valid parameter value.

See *Application Development Guide* for a list of supported authentication types.

SQL1401N **Authentication types do not match.**

Explanation: 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.

Federated system users: this message can also appear when:

- A data source is identified in SYSCAT.SERVEROPTIONS 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.SERVEROPTIONS 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.SERVEROPTIONS and the system default value for PASSWORD violates the data source password requirement.

User Response: 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.

Federated system users:

- If the problem is that the data source requires a password but SYSCAT.SERVEROPTIONS contains SETTING='N' for OPTION='PASSWORD' for that server, alter SYSCAT.SERVEROPTIONS to reflect the true

data source password requirement using the ALTER SERVER SQL statement.

- If the problem is that the data source does not require a password but SYSCAT.SERVEROPTIONS contains SETTING='Y' for OPTION='PASSWORD' for that server, alter SYSCAT.SERVEROPTIONS to reflect the true data source password requirement using the ALTER SERVER SQL statement.

- If the server option for OPTION='PASSWORD' is not set in SYSCAT.SERVEROPTIONS, create an entry using the CREATE SERVER SQL statement to reflect the true data source password requirement.

sqlcode: -1401

sqlstate: 08001

SQL1402N **Unable to authenticate user due to unexpected system error.**

Explanation: Call your system administrator. On unix-based systems, the file *db2ckpw* may not have the correct permission bits set, or the system may have run out of swap/paging space. On Windows NT, the DB2 security service may not have been started or the account may be locked.

The command can not be processed.

Federated system users: this situation can also be detected by the data source.

User Response: On unix-based systems, have the system administrator ensure the correct access permissions are set for *db2ckpw*, and that there is enough swap/paging space allocated. On Windows NT, have the system administrator verify that the DB2 security service is installed and started. If the service is running then ensure that the account is not locked.

SQL1403N **The username and/or password supplied is incorrect.**

Explanation: 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 DB2 Connect 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.

Federated system users: this situation can also be detected by the data source.

User Response: Supply the correct username and password combination.

Federated system users: ensure that the entry in SYSCAT.USEROPTIONS contains the correct username and password for the data source being accessed.

sqlcode: -1403

sqlstate: 08004

SQL1404N   Password expired.

Explanation: Your password has expired.

User Response: Change your password, then retry the request using the new password. You can use the DB2 Client Configuration Assistant, or Command Line Processor CONNECT and ATTACH commands to change your password.

sqlcode: -1404

sqlstate: 08004

SQL1405N   Unable to communicate with local DB2 authentication server.

Explanation: The application failed to authenticate due to an error while communicating to the local DB2 authentication server.

User Response: 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.

Explanation: 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.

User Response: This error is returned to prevent further processing by the system of statements prepared using service features and is expected.

SQL1420N   Too many concatenation operators.

Explanation: The database manager has reached an internal limit while evaluating an expression, of long or large object string result type, that contains concatenation operators.

User Response: 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*.

Explanation: 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

User Response: 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.**

Explanation: 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 2 * extentsize pages in length. The maximum size of a container is operating system dependent. The most common system limit is 2 gigabytes (524288 4K pages).

User Response: Check the administration notification log for details. Then correct the SQL statement.

sqlcode: -1422

sqlstate: 54039

SQL1423N **The query includes a column with a large object data type.**

Explanation: 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.

User Response: 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*.**

Explanation: 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.

User Response: 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.**

Explanation: Any command/API that accepts a userid and password will not accept a password without a userid.

User Response: Resubmit the command/API and supply a userid if you are also supplying a password.

SQL1426N   A default instance cannot be determined.

Explanation: 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.

User Response: Set one of these environment variables to a valid instance name and resubmit the command.

SQL1427N   An instance attachment does not exist.

Explanation: The application is not attached to an instance. The attempted command/API cannot be executed unless there is an existing instance attachment.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: The CATALOG NODE command or API does not allow an entry whose nodename matches the value of the DB2INSTANCE environment variable.

User Response: 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*.

Explanation: 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.

User Response: 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.

Explanation: When the application is remote from the server, relative paths are not allowed.

User Response: 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.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Ensure that the application has no connection before issuing the command or that the connection that exists is to the correct database.

SQL1434N   A CONNECT or ATTACH statement failed because of a client/server incompatibility between 32 bit and 64 bit platforms.

Explanation:
1. Version 7 does not support client/server connections between 32 bit and 64 bit platforms.
2. Version 8 Windows 64-bit database server does not support client/server connections from Version 7 64-bit clients.

User Response: For scenario 1 listed above, you can issue a CONNECT or ATTACH statement in the following scenarios:
- from a 32 bit client to a 32 bit server
- from a 64 bit client to a 64 bit server

For scenario 2 listed above, reissue the statement from a supported client.

sqlcode: -1434

sqlstate: 08004

SQL1440W   The WITH GRANT OPTION has been ignored on the GRANT (Database Authorities) statement, GRANT (Index Privileges) statement, or when granting CONTROL privilege on a table or view.

Explanation: The WITH GRANT OPTION is not applicable when granting database authorities or privileges or privileges on indexes. The WITH GRANT OPTION does not apply to CONTROL privilege on tables, views, indexes, or packages.

All valid requested privileges were granted.

User Response: When granting database authorities or index privileges do not include the WITH GRANT OPTION clause. When granting CONTROL, use a separate grant statement for the CONTROL privilege without specifying the WITH GRANT OPTION clause.

sqlcode: +1440

sqlstate: 01516

SQL1441N   Invalid parameter. Reason code *code*.

Explanation: 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

User Response: 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***.**

Explanation: 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.

User Response: 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.**

Explanation: The user attempted to attach a context to a thread, and the thread is already using a context.

User Response: Detach from the previous context before attaching to a new one.

SQL1444N **The application context cannot be destroyed because it is in use.**

Explanation: 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.

User Response: 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.**

Explanation: A context type of SQL_CTX_MULTI_MANUAL is in effect, but the current thread or process has not been attached to a context.

User Response: Make sure the current thread or process is attached to a context before making any database calls.

SQL1450N **Invalid registration information pointer.**

Explanation: An invalid registration information pointer was passed to the register/deregister DB2 server command/API.

User Response: 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.**

Explanation: Register/deregister DB2 server was issued from an invalid node.

User Response: Reissue the register/deregister DB2 server command/API from a server node.

SQL1452N **An invalid registration location was specified.**

Explanation: An invalid registration location was passed to the register/deregister DB2 server command/API.

User Response: 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.**

Explanation: The file server name specified in the configuration command/API or in the database manager configuration file is missing or invalid.

User Response: 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.**

Explanation: The object name specified in the configuration command/API or in the database manager configuration file is missing or invalid.

User Response: 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.**

Explanation: The IPX socket number specified in the configuration command/API or in the database manager configuration file is missing or invalid.

User Response: 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.**

Explanation: A duplicate object name was found when trying to register the DB2 server object name on the NetWare file server.

User Response: 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.**

Explanation: A bindery login using NWLoginToFileServer cannot be performed if a NetWare directory services connection is already established to the specified file server.

User Response: 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.**

Explanation: 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 '*'.

User Response: 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.**

Explanation: 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.

User Response: Define the missing environment variable and resubmit the command.

SQL1461N **The security option** *security* **is not valid.**

Explanation: 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.

User Response: 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.

SQL1462N **Request is only valid for a syncpoint manager connection.**

Explanation: A request was attempted which is only valid for a syncpoint manager connection, but the syncpoint manager instance is not connected to.

User Response: Connect to the syncpoint manager instance and reissue the request.

SQL1468N **The database manager TCP/IP listener must be configured and running on the server instance** *instance* **(nodenum** *node-num1***) before attempting to CONNECT or ATTACH to node** *node-num2***.**

Explanation: The SET CLIENT command or api or the environment variable DB2NODE was used to set the node for CONNECT or ATTACH to *node-num2*. In order to CONNECT or ATTACH to this node, the Database Manager TCP/IP listener must be configured and running on the server instance *instance* (node *node-num1*).

Note: This message may be returned by an implicit CONNECT or ATTACH.

User Response: Either:
- Ensure that *svcename* has been specified in the database manager configuration on the instance *instance*, node *node-num1*, the DB2COMM environment variable has been set to use TCP/IP, and that TCP/IP listener was successfully started at DB2START time.

  or
- Explicitly catalog the node and database.

sqlcode: -1468

sqlstate: 08004

SQL1469N **Instance** *instance-name* **(nodenum** *node-num1***) does not have node** *node-num2* **specified in its db2nodes.cfg file.**

Explanation: The SET CLIENT command or api or the environment variable DB2NODE was used to set the node for CONNECT or ATTACH to *node-num2*. Subsequent CONNECT or ATTACH processing was unable to locate this node in the db2nodes.cfg file on the instance *instance-name* (node *node-num1*).

Note: This message may be returned by an implicit CONNECT or ATTACH.

User Response: Ensure that the node number specified by the SET CLIENT command or api or the DB2NODE environment variable exists in the db2nodes.cfg file of the intermediate instance *instance-name*, node *node-num1*.

sqlcode: -1469

sqlstate: 08004

SQL1470N **Value of DB2NODE environment variable is not valid.**

Explanation: The DB2NODE environment variable indicates the node the application will attempt to connect to. If DB2NODE is unset or is blank, the application will attempt to connect to the default node. Otherwise, DB2NODE must be

set to the node number of one of the nodes defined on the same host as the application.

User Response: Set the DB2NODE environment variable to one of the following values:

unset causes the application to connect to the default node.

blank causes the application to connect to the default node.

A number
> causes the application to connect to the node with that node number. The node must be running on the same host as the application.

sqlcode: -1470

sqlstate: 08001

SQL1471N **Cannot connect to database** *database-name* **on node** *node-number* **because the database on this node is not synchronized with catalog node.**

Explanation: The end-of-log information on this node does not match the corresponding record on the catalog node. This can be caused by restoring the database on various nodes with backups taken from different times.

User Response: If the database is restored without rolling forward on one node, ensure that the database is restored from consistent offline backups on all nodes without rolling forward.

sqlcode: -1471

sqlstate: 08004

SQL1472N **Cannot connect to database** *database-name* **on node** *node-number* **because the difference between the system time on the catalog node and the virtual timestamp on this node is greater than the max_time_diff database manager configuration parameter.**

Explanation: The system time difference for the machines in the configuration (listed in the db2nodes.cfg file) is greater than the *max_time_diff* database manager configuration parameter.

User Response: Synchronize the system times on all machines, and ensure that the *max_time_diff* parameter is configured to allow for normal communication delays among the database machines.

If the above does not fix the problem, see the *Administration Guide* for possible causes and actions.

sqlcode: -1472

sqlstate: 08004

SQL1473N **Cannot commit the transaction because the time difference between the system time on the local node and the virtual timestamps of node(s)** *node-list* **is greater than the max_time_diff database manager configuration parameter. The transaction is rolled back.**

Explanation: The system time difference for the machines in the configuration (listed in the db2nodes.cfg file) is greater than the *max_time_diff* database manager configuration parameter.

If ″,...″ is displayed at the end of the node list, see the syslog file for the complete list of nodes.

User Response: Synchronize the system times on all machines, and ensure that the *max_time_diff* parameter is configured to allow for normal communication delays among the database machines.

sqlcode: -1473

sqlstate: 40504

SQL1474W **The transaction completed successfully, but the time difference between the system time on the local node and the virtual timestamps of node(s)** *node-list* **is greater than the max_time_diff database manager configuration parameter.**

Explanation: The system time difference for the machines in the configuration (listed in the db2nodes.cfg file) is greater than the *max_time_diff* database manager configuration parameter.

This warning message is returned to read-only transactions because they are not affected by this condition. However, other transactions would be rolled back. This message is to inform the user about this condition so that action can be taken at the earliest possible time.

If ″,...″ is displayed at the end of the node list, see the syslog file for the complete list of nodes.

User Response: Synchronize the system times on all machines, and ensure that the *max_time_diff* parameter is configured to allow for normal communication delays among the database machines.

sqlcode: 1474

sqlstate: 01607

SQL1475W **A system error occurred during CONNECT RESET processing.**

Explanation: The CONNECT RESET was successful but a system error such as a node failure or a communications error may have occurred.

User Response: Check the administration notification log for details. The database at this node may need to be restarted.

sqlcode: 1475

sqlstate: 01622

SQL1476N **The current transaction was rolled back because of error** *sqlcode***.**

Explanation: The transaction was rolled back because:

1. An implicit or explicit CLOSE CURSOR failed, or

2. A table was being created with the NOT LOGGED INITIALLY option, or NOT LOGGED INITIALLY was activated for an existing table. During the same unit of work, either an error occurred, or a ROLLBACK TO SAVEPOINT statement was issued. This unit of work is rolled back, with the following effects:

   • any table that was created in this unit of work is dropped.

   • any NOT LOGGED INITIALLY tables that were activated in the transaction are marked inaccessible, and can only be dropped.

   • the *sqlcode* will be 0 if a ROLLBACK TO SAVEPOINT was issued in the transaction.

3. A rollback to savepoint failed.

The *sqlcode* reported is the SQLCODE of the original error.

User Response: Correct the problem as indicated by the error *sqlcode*, then run the transaction again. Remove any ROLLBACK TO SAVEPOINT statement that is used in the same transaction in which a NOT LOGGED INITIALLY table is created or activated.

sqlcode: -1476

sqlstate: 40506

SQL1477N **Table** *table-name* **cannot be accessed.**

Explanation: An attempt was made to access a table that is not accessible. The table may not be accessible because of one of the following reasons:

• The table had NOT LOGGED INITIALLY activated when the unit of work was rolled back.

- The table is a partitioned declared temporary table and one or more partitions failed since the temporary table was declared (all declared temporary tables have the schema name SESSION).
- ROLLFORWARD encountered the activation of NOT LOGGED INITIALLY on this table or a NONRECOVERABLE load on this table.

Access to this table is not allowed because its integrity cannot be guaranteed.

User Response: One of the following actions can be taken.

- If the table had NOT LOGGED INITIALLY activated, drop the table. If this table is required, re-create it.
- If the table is a declared temporary table, drop the table. If this table is required, declare it again.
- Otherwise, restore from a tablespace or database backup. The backup image must have been taken subsequent to the commit point following the completion of the non-recoverable operation (NOT LOGGED INITIALLY operation, or NONRECOVERABLE load).

sqlcode: -1477

sqlstate: 55019

SQL1478W   The defined buffer pools could not be started. Instead, one small buffer pool for each page size supported by DB2 has been started.

Explanation: The defined buffer pools could not be started. Instead, one small buffer pool for each page size supported by DB2 has been started, and extended storage has been disabled. The defined buffer pools could not be started for one or more of the following reasons:

- Not enough memory could be allocated for the total size of the buffer pools and the extended storage specified for this database.
- The buffer pool files in the database directory are missing or corrupt.

User Response: Check the administration notification log for the correct source of the problem. A possible solution includes:

- Drop or alter the size of one or more of the buffer pools to allow the database to start properly. If there is extended storage specified for the database, adjust the configuration values of num_estore_segs and estore_seg_sz for less memory consumption.

After making the changes, disconnect from the database and start the database again.

sqlcode: +1478

sqlstate: 01626

SQL1479W   Attempt to fetch before the result set returned the first rowset.

Explanation: The requested rowset overlapped the start of the result set and, depending on the fetch orientation specified:

SQL_FETCH_PRIOR
> one of the following situations:
> - the current position was beyond the first row, and the number of the current row is less than or equal to the rowset size.
> - the current position was beyond the end of the result set, and the rowset size was greater than the result set size.

SQL_FETCH_RELATIVE
> the absolute value of the fetch offset was less than or equal to the current rowset size.

SQL_FETCH_ABSOLUTE
> the fetch offset was negative, and the absolute value of the fetch offset was greater than the result set size but less than or equal to the current rowset size.

User Response: No action needed.

SQL1480N **The discover type specified in the DISCOVER database manager configuration parameter is invalid.**

Explanation: Valid values for the DISCOVER parameter in the database manager configuration file are: DISABLE, KNOWN, or SEARCH.

User Response: Update the DISCOVER database manager configuration parameter with a type of DISABLE, KNOWN, or SEARCH.

SQL1481N **One or more of the communications protocols specified in the DISCOVER_COMM parameter is invalid.**

Explanation: Valid values for the DISCOVER_COMM parameter in the database manager configuration file are any combination of NETBIOS, and TCPIP, separated by commas.

User Response: Update the DISCOVER_COMM database manager configuration parameter with any combination of NETBIOS, and TCPIP, separated by commas.

SQL1482W **The BUFFPAGE parameter will only be used if one of the buffer pools is defined with a size of -1.**

Explanation: This is a warning that the BUFFPAGE database configuration parameter will be ignored if none of the database's buffer pools is defined with a size of -1. -1 indicates that the buffer pool is to use the BUFFPAGE parameter as the number of buffer pool pages.

User Response: You can select from the SYSCAT.BUFFERPOOLS to review the buffer pool definitions. If none of the buffer pools are defined with size -1 (NPAGES), then setting the BUFFPAGE parameter will not change the size of the buffer pools for the database.

SQL1490W **Activate database is successful, however, the database has already been activated on one or more nodes.**

Explanation: Database has already been explicitly started (activated) on one or more nodes.

User Response: No action required.

SQL1491N **The database** *name* **was not deactivated because the database is still being used.**

Explanation: The database cannot be deactivated if there are any applications connected to the specified database.

User Response: Ensure that all applications have done a CONNECT RESET, then try again.

SQL1492N **The database** *name* **was not deactivated because it was not activated.**

Explanation: The database cannot be deactivated because the specified database was not active.

User Response: No action is required.

SQL1493N **The application is already connected to an active database.**

Explanation: The ACTIVATE DATABASE and DEACTIVATED DATABASE commands cannot proceed because the application is already connected to a database.

User Response: Disconnect from the database then re-submit the command.

SQL1494W **Activate database is successful, however, there is already a connection to the database.**

Explanation: There is already database connection on one or more nodes.

User Response: No action required.

SQL1495W **Deactivate database is successful, however, there is still a connection to the database.**

Explanation: There is still database connection to one or more nodes.

User Response: No action required.

SQL1496W **Deactivate database is successful, but the database was not activated.**

Explanation: Database was not explicitly started on one or more nodes when deactivate database was executed.

User Response: No action required.

SQL1497W **Activate/Deactivate database was successful, however, an error occurred on some nodes.**

Explanation: Activate/Deactivate database is successful at least on the catalog node and coordinator node, however, on some other node(s) an error has occurred.

User Response: Refer to the diagnostic log to find out what error has occurred on which node, fix the problem if possible and then reissue the activate/deactivate database command.

SQL1500 - SQL1599

SQL1512N **ddcstrc could not write to the specified file.**

Explanation: *ddcstrc* was not able to write the trace information to the filename that it has been instructed to write to.

User Response: 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 default file ddcstrc.tmp.

SQL1513W **ddcstrc has not been turned off.**

Explanation: 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.

User Response: 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 or equal to 65536.**

Explanation: The user specified an invalid buffersize for the ddcstrc command.

User Response: Ensure that the buffersize used is numeric and greater than or equal to 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.

SQL1525N **An error occurred when starting the DB2 security daemon.**

Explanation: An unexpected error occurred when starting the DB2 security daemon.

User Response: Retry the DB2START command. If the problem persists, contact IBM Service.

SQL1526N **db2start failed because DB2VIA support was not started. Reason code** *reason-code***.**

Explanation: DB2VIA support was not started successfully at db2start time. The reason code indicates the error:

1. The VIPL library specified in DB2_VI_VIPL registry variable could not be loaded.

2. The Device name specified in DB2_VI_DEVICE registry variable could not be opened.

3. DB2 does not support the VIA implementation installed.

User Response:

1. Make sure the DB2 registry DB2_VI_VIPL is properly set and the name specified in DB2_VI_VIPL is in the %PATH% environment variable.

2. Make sure the DB2 registry DB2_VI_DEVICE is properly set.

3. DB2 only supports a VIA implementation that supports at least Reliable Delivery reliability level. It is also required that the VIA implementation complies with Intel Virtual Interface Architecture Implementation Guide and passes the conformance Suite. Make sure the VIA implementation you chose satisfies these requirements.

SQL1530W **The specified degree of parallelism will be ignored because the system is not enabled for intra-partition parallelism.**

Explanation: Either the DEGREE bind option was specified with a value greater than 1 or the SET CURRENT DEGREE statement was executed with a value greater than 1, but the database manager was not enabled for intra-partition parallelism.

The database manager must be started with the intra_parallel configuration parameter set to ON in order for the instance to be enabled for intra-partition parallelism.

The statement or command completed successfully, but the degree specification was ignored.

User Response: If you want to use intra-partition parallelism restart the database manager with the intra_parallel configuration parameter set to ON.

Otherwise use a value of 1 or ANY for the degree specification.

sqlcode: +1530

sqlstate: 01623

SQL1550N **The SET WRITE SUSPEND command failed. Reason code =** *reason-code***.**

Explanation: You cannot issue the SET WRITE SUSPEND command until the condition indicated by *reason-code* is resolved:

1 Database is not activated.

2 A backup database operation is currently in progress for the target database. You cannot suspend write operations until DB2 completes the backup.

3 A restore database operation is currently in progress for the target database. You cannot suspend write operations for this database until DB2 completes the restore operation.

4 Write operations have already been suspended for this database.

5 The current state of one or more table spaces does not permit write suspend.

User Response:

1 Activate the database by issuing the ACTIVATE DATABASE command, then re-issue the SET WRITE SUSPEND command.

2 Wait until the BACKUP procedure finishes, then re-issue the SET WRITE SUSPEND command.

3 Wait until the RESTORE procedure finishes, then re-issue the SET WRITE SUSPEND command.

4 The database is already in suspended state. To resume write operations for this database, issue the SET WRITE RESUME command.

5 To view the table space states, issue the LIST TABLESPACES command. For those table spaces that are in a pending state, issue the appropriate command to bring them out of the pending state before reissuing the SET WRITE SUSPEND command. If one or more

table spaces involve an operation in progress, wait until the operation is complete before reissuing the SET WRITE SUSPEND command.

sqlcode: -1550

SQL1551N **The SET WRITE RESUME command failed because the database is not currently in WRITE SUSPEND state.**

Explanation: The database is not currently in WRITE SUSPEND state. You can only resume write operations for a database for which write operations have been suspended.

User Response: No action is required, because write operations are enabled for this database. To suspend write operations for the database, issue the SET WRITE SUSPEND command.

sqlcode: -1551

SQL1552N **The command failed because the database is currently in WRITE SUSPEND state.**

Explanation: This command is not allowed when write operations are suspended for the database. The database is in WRITE SUSPEND state.

User Response: If the command that failed was RESTART DATABASE, re-issue the RESTART DATABASE command using the WRITE RESUME option. In a multi-node database environment, re-issue the command concurrently on all nodes.

If the command that failed was a BACKUP or RESTORE command, issue a SET WRITE RESUME FOR DATABASE command to resume write operations for the database. Then re-issue the BACKUP or RESTORE command.

sqlcode: -1552

SQL1553N **DB2 cannot be stopped because one or more databases are in WRITE SUSPEND state.**

Explanation: You cannot shut down a database for which write operations are suspended. The database is in WRITE SUSPEND state.

User Response: Issue the SET WRITE RESUME command to resume write operations for the database, then re-issue the db2stop command.

sqlcode: -1553

SQL1560N **Statistics profile does not exist for table** *table-name***.**

Explanation: A statistics profile needs to be defined before use.

User Response: Register a statistics profile for the table using the SET PROFILE option of the RUNSTATS command and then reissue the operation.

SQL1561N **Statistics option not compatible with the database server level.**

Explanation: One or more of the options specified are not compatible with the database server level. This database server version does not supprort all of the statistics options available using the db2runstats API.

User Response: Use the sqlustat API to collect the statistics or use the db2runstats API and specify only those options that are available using the sqlustat API.

SQL1562N **Statistics node option is not compatible with the existing table and/or index statistics for the table** *table-name***.**

Explanation: The statistics node option is different from the existing table and/or index statistics of the table.

User Response: Use consistent node option to collect the statistics for a table.

SQL1580W **Trailing blanks were truncated while performing conversion from codepage** *source-code-page* **to codepage** *target-code-page***. The maximum size of the target area was** *max-len***. The source string length was** *source-len* **and its hexadecimal representation was** *string***.**

Explanation: 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.

User Response: Ensure that the output is as expected and that the truncation has not caused any unexpected consequences.

sqlcode: +1580

sqlstate: 01004

SQL1581N **The table** *table-name* **cannot be in append mode and have a clustering index.**

Explanation: There are two situations where this error is issued.

• A clustered index exists for a table and ALTER TABLE is used to attempt to place the table in append mode.

• A table is in append mode and CREATE INDEX is used to attempt to create a clustering index.

User Response: If a clustering index is necessary then alter the table to set append mode off. If append mode is desired, drop the existing clustering index on the table.

sqlcode: -1581

sqlstate: 428CA

SQL1582N **The PAGESIZE of the table space** *tbspace-name* **does not match the PAGESIZE of the bufferpool** *bufferpool-name* **associated with the table space.**

Explanation: The PAGESIZE value specified in the CREATE TABLESPACE statement does not match the page size of the bufferpool specified for use with the table space. These values must match.

The statement cannot be processed.

User Response: Change the value specified for PAGESIZE to match the page size of the bufferpool or change the bufferpool to one that has a page size that matches.

sqlcode: -1582

sqlstate: 428CB

SQL1583N **PAGESIZE value** *pagesize* **is not supported.**

Explanation: The PAGESIZE that is specified in the CREATE BUFFERPOOL or CREATE TABLESPACE statement is not a supported page size. The supported page sizes for Version 5 are 4K, 8K, 16K, and 32K.

The statement cannot be processed.

User Response: Specify one of the supported page sizes.

sqlcode: -1583

sqlstate: 428DE

SQL1584N **System temporary tablespace with page size of at least** *pagesize* **could not be found.**

Explanation: A system temporary table space was required to process the statement. There was no system temporary table space available that had a page size of *pagesize* or larger.

The statement cannot be processed.

User Response: Create a system temporary table space with a page size of at least *pagesize*.

sqlcode: -1584

sqlstate: 57055

SQL1585N   A system temporary table space with sufficient page size does not exist.

Explanation: One of the following conditions could have occurred:

1. The row length of the system temporary table exceeded the limit that can be accommodated in the largest system temporary table space in the database.

2. The number of columns required in a system temporary table exceeded the limit that can be accommodated in the largest system temporary table space in the database.

The system temporary table space limits depend on its page size. These values are:

```
Max          Max   Page size of
Record       Cols  temporary
Length             table space
-----------  ----  ------------
1957  bytes  244   2K
4005  bytes  500   4K
8101  bytes  1012  8K
16293 bytes  1012  16K
32677 bytes  1012  32K
```

User Response: Create a system temporary table space of a larger page size supported, if one does not already exist. If such a table space already exists, eliminate one or more columns from the system temporary table. Create separate tables or views, as required, to hold additional information beyond the limit.

sqlcode: -1585

sqlstate: 54048

SQL1590N   LONG VARCHAR and LONG VARGRAPHIC fields are not permitted in TABLESPACEs which are built on DEVICEs.

Explanation: 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.

User Response: Alternatives:

- Choose one of the LOB columns types (BLOB,CLOB,DBCLOB) instead of LONG.
- Use a table space with the correct attributes.

sqlcode: 1590

sqlstate: 56097

SQL1591N   The ON option of the SET INTEGRITY statement is not valid since the table *table-name* is not in the correct state.

Explanation: The ON option can only be specified on a table that is in the check pending state and the first (for foreign key constraints), second (for check constraints) and fifth (for materialized query tables) fields in the CONST_CHECKED column of SYSCAT.TABLES catalog must be either 'Y' or 'U'.

User Response: If the table is not in check pending, do not specify the ON option. If the table is in check pending, perform integrity checking on the table (or refresh the table if it is a materialized query table) before executing the SET INTEGRITY statement with the ON option.

sqlcode: -1591

sqlstate: 55019

SQL1592N   The INCREMENTAL option is not valid with reason code *reason-code* since the table *table-name* cannot be incrementally processed.

Explanation: The cause is based on the *reason-code*:

32   The table is not a REFRESH IMMEDIATE materialized query table, nor a REFRESH DEFERRED materialized query table with a supporting staging table, nor a PROPAGATE IMMEDIATE staging table.

33 A Load Replace or Load Insert has occurred to the table if it is a materialized query table or staging table.

34 A Load Replace has occurred to the table after the last integrity check.

35 One of the following:

- The materialized query or staging table was newly created. Full processing is required for the first time the table is checked for integrity after it has been created.

- New constraint has been added to the table itself or its parents (or its underlying table if it is a materialized query table or staging table) while in check pending.

- If it is a materialized query table or staging table, a Load Replace has occured to any underlying table of the table after the last refresh.

- If it is a materialized query table, at least on underlying table was forced to full access (using the FULL ACCESS option) before the materialized query table was refreshed.

- If it is a staging table, at least one underlying table was forced to full access (USING the FULL ACCESS option) before the staging table was propagated.

- If it is a deferred materialized query table and its corresponding staging table is in incomplete state.

- Some of its parents (or underlying table if it is a materialized query table or staging table) have been non-incrementally checked for integrity.

- The table was in the check pending state before migration. Full processing is required for the first time the table is checked for integrity after migration.

- The table was placed in the check pending state during a point in time roll-forward operation.

User Response: Do not specify the INCREMENTAL option. The system will check the entire table for constraint violations (or if it is a materialized query table, recompute the materialized query table definition query).

sqlcode: -1592

sqlstate: 55019

SQL1593N The REMAIN PENDING option is not valid since the table *table-name* **is not in check pending state.**

Explanation: The table must be in check pending state if the REMAIN PENDING option is to be specified.

User Response: Do not specify the REMAIN PENDING option.

sqlcode: -1593

sqlstate: 55019

SQL1594W Integrity of non-incremental data remains unverified by the database manager.

Explanation: The table was previously unchecked. When the option NOT INCREMENTAL is not specified, incremental processing of the table is done. The previously unchecked portion of the table remains unchecked and the corresponding values in the CONST_CHECKED column remains marked as 'U'.

User Response: No action is required. To verify the integrity of previously unchecked data and have the system maintain the integrity of the data in the table, issue the SET INTEGRITY...OFF statement to place the table in the check pending state and then re-execute the SET INTEGRITY...IMMEDIATE CHECKED statement with the NOT INCREMENTAL option.

sqlcode: +1594

sqlstate: 01636

SQL1595N **The integrity for table** *table-name* **cannot be checked because either a parent table in a referential constraint has not been checked or an underlying table of the materialized query or staging table has not been checked.**

Explanation: To avoid propagating this table with data that might violate integrity checks, all parents must not be in check pending in order to check integrity for this table. If this is a materialized query table, all underlying tables must have been validated in order to refresh this table. If this is a staging table, all underlying tables must not be in check pending in order to propagate this table. A table is validated when the first (for foreign key constraints), the second (for check constraints), the fifth (for materialized tables), and the seventh (for staging tables) fields in the CONST_CHECKED column of SYSCAT.TABLES are either 'Y' or 'U'.

User Response: Check integrity for all its parent tables (or underlying tables if it is a materialized query table or a staging table) and re-execute this statement.

sqlcode: -1595

sqlstate: 55019

SQL1596N **WITH EMPTY TABLE cannot be specified for** *table-name*.

Explanation: The WITH EMPTY TABLE clause cannot be specified for table *table-name* because the table:

- is a materialized query table
- has a dependent refresh immediate materialized query table, or
- is the parent in a referential constraint.

The WITH EMPTY TABLE clause cannot be specified when altering such a table to ACTIVATE NOT LOGGED INITIALLY.

The statement cannot be processed.

User Response: Remove the WITH EMPTY TABLE clause from the ALTER TABLE statement.

sqlcode: -1596

sqlstate: 42928

SQL1600 - SQL1699

SQL1601N **The Database System Monitor input parameter** *parameter* **is a null pointer.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Reissue the command using a valid object type. Refer to the *Administrative API Reference* and *Application Development Guide* for further information about valid object types.

SQL1603N **The parameter** *parameter* **is not specified in the input data structure (sqlma).**

Explanation: 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.

User Response: The user should reissue the command specifying a valid parameter value.

SQL1604N **The parameter** *parameter* **is not null terminated.**

Explanation: A null character is expected at the end of the character string parameter.

The command can not be processed.

User Response: Add a null character at the end of the character string parameter and reissue the command.

SQL1605W **The database** *db-alias* **is not active.**

Explanation: 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.

User Response: Verify that the database alias is correct and the database has been started.

SQL1606W **The Database System Monitor output buffer is full.**

Explanation: 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.

User Response: 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.**

Explanation: The database manager is out of working memory to process the Database System Monitor command.

The command can not be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: The user should reissue the command with a valid parameter value.

SQL1611W No data was returned by Database System Monitor.

Explanation: 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.

User Response: 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.

Explanation: The target path specified in the CREATE EVENT MONITOR statement is not a valid path name. The command could not be processed.

User Response: Resubmit the statement with the correct event monitor target path.

sqlcode: -1612

sqlstate: 428A3

SQL1613N The specified event monitor option is invalid.

Explanation: 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.

User Response: 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*.

Explanation: 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.)

User Response: 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.

Explanation: 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.

User Response: No user response is required.

sqlcode: +1615

sqlstate: 01598

SQL1616N   The limit on the maximum number of active event monitors has already been reached.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: The specified event monitor was created with the same target path as another event monitor. This other event monitor was activated at least once, and has left .evt and/or .ctl files in the target path. These files may be in use by application(s) that are reading the event information they contain.

User Response: If the other event monitor is currently active, deactivate it. After ensuring no application(s) are using the files it created in the target path, remove the files. Then 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.

Explanation: The specified event monitor is currently active, and therefore cannot be dropped.

User Response: Deactivate the event monitor and resubmit the DROP EVENT MONITOR statement.

sqlcode: -1619

sqlstate: 55034

SQL1620N   Unable to flush event monitor. Reason code *rc*.

Explanation: The event monitor could not be flushed. Possible reasons are:

1. The event monitor is not started.

2. The event monitor is running at a pre-version 6 level of output, for which flush is not available.

User Response: Ensure that the event monitor is started. If the event monitor is running at a

pre-version 6 level of output, do not attempt to flush it.

sqlcode: -1620

sqlstate: 55034

SQL1621N   **The transaction in which the specified event monitor was created has not yet been committed. The event monitor cannot be activated.**

Explanation: An event monitor cannot be activated until the transaction in which it was created has been committed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The limit imposed on the number of objects that are permitted in the sqlma input structure has been exceeded.

User Response: 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.**

Explanation: The sqlma parameter contained references to databases that reside on different nodes.

User Response: Modify the sqlma parameter so that all database objects reference the same node and then try the call again.

SQL1625W   **The monitor is not able to convert from the code page** *source* **to the code page** *target*. **This conversion was attempted for data pertaining to type** *type*.

Explanation: Possible types are as follows:

1. statement text
2. dcs application
3. application
4. table
5. locks
6. table space

Conversion of data from the source code page to the target code page is not supported. This condition can occur under the following conditions:

1. source and target code page combination is not supported by the database manager.
2. source and target code page combination is not supported by the operating system character conversion utility on the server node.

This situation can occur when the monitor tries to convert data pertaining to a database whose code page is incompatible to that of the monitor application.

User Response: 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.

If possible ensure that the database being monitored and the monitoring application are in the same codepage.

SQL1626W Overflow occurred while performing conversion from codepage *source* **to codepage** *target*. **The size of the target area was** *max-len* **,the data pertained to type** *type* **and the first eight characters are** *data*.

Explanation: Possible types are as follows:
1. statement text
2. dcs application
3. application
4. table
5. locks
6. table space

The monitor is unable to convert the data due to space constraints. The data is retained in its original form.

User Response: If possible ensure that the database being monitored and the monitoring application are in the same codepage.

SQL1627W A snapshot api request was made at the self-describing datastream level, but the server was only able to return a snapshot of the fixed size structure format.

Explanation: Although the application issuing the snapshot request made it at the SQLM_DBMON_VERSION6 or later level, the server returning the snapshot returned a downlevel view of the data.

User Response: In the self describing data format for snapshot (DB2 version 6 and later), collected information, including the server level, is returned as part of the snapshot datastream. In the pre-version 6 levels of DB2, the snapshot collected information is returned in an sqlm_collected structure. You must use the sqlm_collected structure, and the old data stream processing method to parse this snapshot datastream.

SQL1628W A remote get switches operation returned partial results because the output buffer is full. To retrieve the full results, use a minimum buffer size of *size* **bytes.**

Explanation: The output buffer provided was not large enough to return all the switch data available. The monitor returned as much data as possible using the given output buffer.

User Response: Allocate a larger data buffer and re-issue the switch request.

SQL1629W A remote snapshot operation failed on node or nodes *node-list*, **with reason code or codes** *reason-list*.

Explanation: Some type of failure occurred during an operation on a remote node for reason <reason-code>, which is one of the following:

1   FCM was unable to communicate with the target node, due to a node failure or communications error.

2   The snapshot operation failed to complete on the target node. Please see the administration notification log for the specific sqlca.

User Response: If the cause of the error was a node failure, or communication error, you will need to resolve the communication error, or restart the node that failed to correct the error.

If the error was due to a snapshot operation failure on a remote node, see the administration notification log for the sqlca from the failed operation, and refer to the instructions for that

code to correct the problem.

SQL1650N   **The function invoked is no longer supported.**

Explanation: The user has attempted to call an API that is no longer supported in this version of the database manager.

User Response: The function required may be supported by a different API call. Check in the *Administrative API Reference* and *Application Development Guide* to determine if the function is supported through a different API call.

SQL1651N   **The request cannot be executed because the DB2 server version does not support this functionality.**

Explanation: Some new functionality is not supported against older DB2 server versions. Another possible cause of this error could be the request referenced objects with qualifiers of length that exceeds the support of the server version.

User Response: Execute the request against a DB2 server where the latest DB2 server version has been installed, or upgrade the server to the latest DB2 server version.

SQL1652N   **File I/O error occurred.**

Explanation: Error occurred either opening, reading, writing, or closing a file.

User Response: Check the db2diag.log for details. Also, check for disk full conditions, file permissions, and operating system errors.

SQL1653N   **An invalid profile path was specified.**

Explanation: A full path to a file where server information should be generated must be specified.

User Response: Ensure that the profile path specified is correct and not null.

SQL1654N   **An instance path error was encountered.**

Explanation: The instance path could not be returned.

User Response: Check that the DB2INSTANCE path was specified correctly. Check that the complete path length specified is not approaching the maximum supported by the operating system.

SQL1660N   **The Generator at the server, which is used by Discovery to gather server information, has failed.**

Explanation: A server system failure has occurred.

User Response: Report the failure to your DB2 server administrator. More details of the failure can be found in the server's db2diag.log file.

SQL1670N   **The discover type specified in the DISCOVER database manager configuration parameter indicates discovery is disabled.**

Explanation: DISCOVER = DISABLE is configured in the database manager configuration file.

User Response: If DISCOVER functionality is required, change the discover type to KNOWN or SEARCH.

SQL1671N   **The search discovery request failed. Check the administration notification log for further details.**

Explanation: The search discovery request failed for one of the following reasons:

1. Initialization failed (sqleCommonInitializationForAPIs)
2. Failed to retrieve the client instance path (sqloinstancepath)
3. Failed to open the output file (sqlofopn)
4. Failed to write to output file (sqlofprt)
5. Failed to get memory (sqlogmblk)

6. Failed to retrieve database manager configuration (sqlfcsys)

7. NetBIOS call failed

8. DB2 internal system function failed (sqlogpid, sqlogmt)

Check the db2diag.log for details.

User Response:

1. If initialization fails, try re-booting the machine or re-installing the product.

2. If there was an instance path failure, check the DB2INSTANCE value to ensure it is correctly set.

3. If the open or write to file failed, check that you have access to open and write a file to the <sqllib path>\<instance>\tmp directory on Intel machines, or the <instance path>/sqllib/tmp directory on UNIX machines.

4. If get memory failed, check the available memory on your machine.

5. If retrieving the DBM configuration failed, try re-booting the machine or re-installing the product.

6. If a NetBIOS call failed:

   • If Add Name failed with a return code of 13, check that the nname configured in the DBM configuration was not used to configure another DB2 client/server on the network.

   • Check that NetBIOS was installed and configured properly, and is functioning correctly.

   • Check your network for problems.

7. If a DB2 internal system function failed, check that the operating system functions on your machine are functioning correctly.

DB2 Service can provide details about the error codes returned by the above functions and written to the db2diag.log.

SQL1700 - SQL1799

SQL1673N   The address list specified as input to the discover interface is invalid.

Explanation: The application program used an invalid input address list pointer. The address list points to nothing.

User Response: Ensure that a valid input address list pointer is specified in the application program, and that it is not null.

SQL1674N   The server address specified as input to the discover interface is invalid.

Explanation: The application program used an invalid input server address pointer. The server address points to nothing.

User Response: Ensure that a valid input server address is specified in the application program, and that it is not null.

SQL1675N   Discovery is only allowed against DB2 administration servers. The communications information provided does not access an administration server.

Explanation: A KNOWN discovery request was issued against a DB2 server that is not a DB2 administration server. The communications information specified is not correct.

User Response: Verify that DB2ADMINSERVER is set on the DB2 server instance you are accessing. This indicates that the server instance is a DB2 administration server. Retry the KNOWN discovery request with the correct communications information.

SQL1700N **The reserved schema name** *name* **was found in the database during database migration.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

User Response: No action required.

SQL1703W **The db2event directory could not be created during database migration.**

Explanation: The database was migrated successfully but the db2event directory could not be created.

This is a warning only.

User Response: 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***.**

Explanation: 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.

6 failed to relocate database.

7 failed to access the database subdirectory or one of the database files.

8 failed to update database container tag.

9 table space access is not allowed.

10 an invalid type name was found.

11 Specific name of procedure conflicts with specific name of function.

12 Datajoiner abstract datatype found.

13   Datajoiner extended index found.

17   Failure to allocate new page from the DMS system catalog table space.

18   Duplicate functions encountered for unicode database migration.

For a unicode database, the following are considered equivalent string types:

```
CHAR and GRAPHIC
VARCHAR and VARGRAPHIC
LONG VARCHAR and LONG VARGRAPHIC
CLOB and DBCLOB
```

Function signatures that are only different in equivalent string types are considered "duplicates" and cannot co-exist in the database. For example, func1(CHAR, GRAPHIC, VARCHAR) and func1(GRAPHIC, GRAPHIC, VAR GRAPHIC) are duplicates.

19   Duplicate specificname encountered.

SYSIBM.SYSFUNCTIONS and SYSIBM.SYSPROCEDURES are merged into SYSIBM.SYSROUTINES catalog in the database migration. However, the specificname must be unique within the SYSIBM.SYSROUTINES.

20   ADT data type encountered.

One or more abstract data types was found in the database. Migration failed. All abstract data types must be dropped prior to database migration to the current release.

21   Database migration is complete on the catalog node and other nodes, but not on all nodes. There are some nodes reporting node failure. As a result, migration has not been performed on these nodes.

22   Database migration failed because the catalog node is reporting node failure.

23   Database migration failed. Federated database migration to a 64-bit instance is not supported.

User Response: Possible solutions based on the reason code are:

1   The reserved schema names are: SYSIBM, SYSCAT, SYSSTAT, and SYSFUN. 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 updatable. Resubmit the database migration command. If the problem persists, inform your IBM service representatives.

6   Restore the database from the database backup.

7   Restore the database from the database backup.

8   Resubmit the database migration command. If the problem persists, contact your IBM service representative.

9   Correct the table space by going back to the release of the database manager in use prior to the current release. Refer to

message SQL0290N for the suggested actions to take to correct the table space.

10 The type name is reserved by the system. Ensure that the type and database objects that use the type are dropped and recreated using a non-reserved type 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.

11 Ensure that the procedure is dropped and recreate procedure using a different specific 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.

12 Ensure that the abstract datatype is dropped. 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.

13 Ensure that the extended index is dropped. 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.

17 Restore database backup onto its previous database manager system. Add more containers to the table space. You should allocate 70% free space for database migration. Move back to the current release and migrate the database.

18 Restore the database backup onto its previous database manager system. Drop one of the duplicate functions. If necessary, recreate the function with a different signature. Move back to the current release of the database manager system. Resubmit the database migration command. Please refer to the Quick Beginnings book and follow the

instructions on how to verify that the database is ready for migration before moving to the current release of DB2.

19 Restore the database backup onto its previous database manager system. Drop either the function or procedure. If necessary, recreate it under a different specificname. Move back to the current release of the database manager system. Resubmit the database migration command. Please refer to the Quick Beginnings book and follow the instructions on how to verify that the database is ready for migration before moving to the current release of DB2.

20 Restore the database backup onto its previous database manager system. Drop all the abstract data types. Move back to the current release of the database manager system. Resubmit the database migration command. Please refer to the Quick Beginnings book and follow the instructions on how to verify that the database is ready for migration before moving to the current release of DB2.

21 Check the administration notification log to determine the nodes that are reporting node failure. Correct the node failure situation and resubmit the database migration command. As migration only takes place on nodes that require migration, you can submit the command from any node.

22 Correct the node failure situation on the catalog node. Resubmit the migration command.

23 Migrate the Federated database to a 32-bit instance.

SQL1705W   Unable to update a database directory entry to the current release level.

Explanation: 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.

User Response: Uncatalog the database alias(es) for the migrated database and re-catalog the database alias using the same information.

SQL1706W   At least one non local database was detected in the node directory for this instance during word size instance migration.

Explanation: While performing word size instance migration, at least one database not created under this instance was encountered. Such databases must have the same word size as this instance for migration to complete successfully.

User Response: Ensure that all databases cataloged at the instance have the same word size.

SQL1707N   Unable to migrate the instance word size.

Explanation: There was an error while attempting to migrate the word size of the instance. Please contact your IBM service representative.

User Response: Contact your IBM service representative.

SQL1708W   Database migration completed with the following warning code: *warning-code***.**

Explanation: Database migration completed with a warning. The warning codes are as follows:

1   One or more nodes failed to migrate.

User Response: Possible solution based on the warning codes:

1   Re-issue the database migration command on that node.

SQL1749N   The NOT LOGGED INITIALLY attribute cannot be activated for table *table-name* **since it was not created with NOT LOGGED INITIALLY.**

Explanation: The table *table-name* was not created with the NOT LOGGED INITIALLY attribute specified and therefore this attribute cannot be activated using ALTER TABLE.

The statement cannot be processed.

User Response: Remove the NOT LOGGED INITIALLY clause from the ALTER TABLE statement.

sqlcode: -1749

sqlstate: 429AA

SQL1750N   Foreign keys cannot reference a parent key table *table-name* **that was created with the NOT LOGGED INITIALLY clause.**

Explanation: A table that was created with the NOT LOGGED INITIALLY clause cannot be referenced by a foreign key.

User Response: Remove the foreign key reference to *table-name* in the ALTER or CREATE TABLE statement.

sqlcode: -1750

sqlstate: 429A0

SQL1751N   The resulting nodegroup would not contain any nodes that can be used in a partition map.

Explanation: A nodegroup must contain at least one node that can be used in a partition map. If a node is added to a nodegroup WITHOUT TABLESPACES, it cannot be included in the partition map because there are no containers for the table spaces defined on the nodegroup. If a node is added LIKE another node and the other node does not have containers for all table spaces on the nodegroup then neither node can be included in the partition map.

User Response: Do not drop all the nodes in

the nodegroup without adding at least one node. If tablespaces are already defined on the nodegroup but there are no tables, ensure that at least one node has containers for all the table spaces.

sqlcode: -1751

sqlstate: 428C0

SQL1752N **The table space cannot be created in the nodegroup** *ngname*.

Explanation: The nodegroup IBMTEMPGROUP can be specified if and only if the table space is a system temporary table space.

User Response: For system temporary table spaces, specify nodegroup IBMTEMPGROUP. For other table space types, specify a nodegroup other than IBMTEMPGROUP.

sqlcode: -1752

sqlstate: 429A1

SQL1753N **Node** *node-number* **does not have containers for all the system temporary table spaces defined in the nodegroup IBMTEMPGROUP.**

Explanation: A node must have containers defined for all the system temporary table spaces defined in the nodegroup IBMTEMPGROUP for this database before it can be included in a nodegroup.

User Response: Issue the ALTER TABLESPACE statement to add containers on every node for every system temporary table space in the database.

sqlcode: -1753

sqlstate: 57052

SQL1754N **The index table space or long table space is not in the same nodegroup as the primary table space.**

Explanation: All the table spaces specified in the CREATE TABLE statement must belong to the same nodegroup.

User Response: Make sure all the table spaces specified in the CREATE TABLE statement belong to the same nodegroup.

sqlcode: -1754

sqlstate: 42838

SQL1755N **Node** *node-number* **does not have containers for all the table spaces defined in the nodegroup** *ngname*.

Explanation: A node must have containers defined for all the table spaces defined in a nodegroup before that node can be included in a redistribution operation of a nodegroup.

User Response: Issue the ALTER TABLESPACE statement to add containers on the node for all the table spaces defined in the nodegroup.

SQL1756N **More than one clause specifies containers without the ON NODES clause.**

Explanation: For the CREATE TABLESPACE, the USING clause without the ON NODES clause can only be specified once.

For the ALTER TABLESPACE, the ADD clause without the ON NODES clause can only be specified once.

The statement was not processed.

User Response: Correct the statement and reissue it.

sqlcode: -1756

sqlstate: 428B1

SQL1757N **A USING clause without an ON NODES clause is missing.**

Explanation: In CREATE TABLESPACE statement, every USING clause specifies the ON NODES clause. However not all the nodes in the nodegroup are included, and therefore not all nodes in the nodegroup have containers.

The statement is not processed.

User Response: Ensure that the USING clause

without the ON NODES clause is specified or that each node in the nodegroup is included once in an ON NODES clause.

sqlcode: -1757

sqlstate: 428B1

SQL1758W   The containers not designated for specific nodes are not used by the table space on any node.

Explanation: The ALTER TABLESPACE and CREATE TABLESPACE statement includes container specification for all the nodes in the nodegroup. The specification of the containers that is not followed by an ON NODES clause is redundant and has been ignored.

The statement has been processed.

User Response: If the containers are needed on some of the nodes, issue an ALTER TABLESPACE statement to add the necessary containers.

sqlcode: -1758

sqlstate: 01589

SQL1759W   Redistribute nodegroup is required to change data partitioning for objects in nodegroup *nodegroup-name* **to include some added nodes or exclude some dropped nodes.**

Explanation: This warning indicates that a change being made using the ALTER NODEGROUP or ALTER TABLESPACE statements has not caused the partitioning map for the nodegroup to be changed. The partitioning map for a nodegroup is only changed immediately by these statements if there are no tables defined using any table space defined in the nodegroup or the node being dropped is not in the partitioning map.

This warning is issued in the following cases:

- One or more nodes are added using ALTER NODEGROUP ADD NODE
- One or more nodes are dropped using ALTER NODEGROUP DROP NODE

- Containers are added for table space and no further containers are required for a node to be used.

For all of these cases, a table is already defined using a table space in the nodegroup.

User Response: If you want the node included or excluded for data partitioning then issue the REDISTRIBUTE NODEGROUP command or API. Another alternative is to drop all tables that use table spaces in the nodegroup.

sqlcode: +1759

sqlstate: 01618

SQL1760N   The CREATE statement for stored procedure *procedure-name* **must have a valid LANGUAGE clause, EXTERNAL clause and PARAMETER STYLE clause.**

Explanation: A required clause is missing in the CREATE for procedure *procedure-name*. LANGUAGE, EXTERNAL and PARAMETER STYLE must be specified.

User Response: Add the missing clause, then try again.

sqlcode: -1760

sqlstate: 42601

SQL1761N   The nodegroup *ngname* **is not defined for the buffer pool** *bpname***.**

Explanation: The nodegroup for the table space has not been defined for the buffer pool. The table space cannot be created or altered to use this combination of nodegroup and buffer pool.

The statement cannot be processed.

User Response: The possible actions are:

- specify a buffer pool that has defined the nodegroup for the table space
- alter the buffer pool to add the nodegroup of the table space
- If CREATE TABLESPACE, specify a nodegroup that is defined for the buffer pool.

sqlcode: -1761

sqlstate: 42735

SQL1762N   **Unable to connect to database because there is not enough space to allocate active log files.**

Explanation: There is not enough disk space to allocate active log files. Possible reasons include:

- There is insufficient space available on the device used to store the recovery logs.

- If userexits are enabled, the userexit program may be failing due to an incorrect path, incorrect install directory, sharing violation, or other problem.

User Response: Based on the cause:

- Ensure that there is sufficient space on the device for the primary logs, as DB2 may require extra space to allocate new logs so that the database will start with at least LOGPRIMARY log files. Do NOT delete recovery logs to free space, even if they appear inactive.

- Ensure the userexit program is operating correctly by manually invoking it. Review the instructions provided in the sample userexit source code for compiling and installing the userexit program. Ensure that the archive destination path exists.

As a last resort, try reducing the values for LOGPRIMARY and/or LOGFILSIZ database configuration parameters so that a smaller set of active log files are used. This will reduce the requirement for disk space.

Reissue the connect statement after determining and correcting the problem.

SQL1763N   **There are multiple conflicting container operations in the ALTER TABLESPACE statement.**

Explanation: The specified container operations are conflicting. The BEGIN NEW STRIPE SET container operation cannot be used with any of the other container operations (ADD, DROP, EXTEND, REDUCE, or RESIZE) in the same ALTER TABLESPACE statement for the same node. Pages cannot both be added to (using ADD, EXTEND, or RESIZE) and removed from (using DROP, REDUCE, or RESIZE) a table space in the same ALTER TABLESPACE statement for the same node.

User Response: If using the BEGIN NEW STRIPE SET container operation and other operations in the same statement, move the BEGIN NEW STRIPE SET operation into its own ALTER TABLESPACE statement. If both adding pages to and removing pages from a table space, use two ALTER TABLESPACE statements.

sqlcode: -1763

sqlstate: 429BC

SQL1764N   **Specified size for a RESIZE action in an ALTER TABLESPACE statement is smaller than the current size of the table space container.**

Explanation: The size specified in an ALTER TABLESPACE statement using the RESIZE action is smaller than the current size of the table space container. You can only increase the size of a container.

User Response: Specify a size that is larger than the current size value of the table space container.

sqlcode: -1764

sqlstate: 560B0

SQL1800 - SQL1899

SQL1800N **An invalid pointer to structure sqle_request_info was passed to the catalog admin command/api.**

Explanation: The pointer to structure sqle_request_info passed as a parameter to the catalog admin command/api was invalid. For Client Configuration Assistance request this pointer must not be NULL.

User Response: Specify a valid pointer to sqle_request_info and re-submit the command.

SQL1801N **Invalid request type.**

Explanation: The specified request type is not supported for this command.

User Response: Ensure that the request type is one of the following supported request types:

1. SQLE_CCA_REQUEST - CCA catalog node request for catalog and open scan command
2. SQLE_DAS_REQUEST - DAS catalog node request for catalog and open scan command
3. SQLE_CND_REQUEST - Open scan command for CCA and DAS catalog entries

---

SQL1802N **No entry belongs to the request type.**

Explanation: There is no entry in the node directory that is catalogued by the given request type.

User Response: Catalog an entry with the same request type and re-submit the command.

SQL1803N **The requested operation cannot be executed in "No Package Lock" mode. An affected package is** *pkgschema.pkgname 0Xcontoken***.**

Explanation: The database manager is currently operating in "No Package Lock" mode. This mode has been activated by setting the DB2_APM_PERFORMANCE registry environment variable to "ON".

In this mode, the following classes of operations are prevented from being executed because of their impact on packages:

- An operation that invalidates packages.
- An operation that makes a package inoperative.
- Binding, rebinding (explicitly or implicitly) or dropping a package.

Since the requested operation would affect the package *pkgschema.pkgname 0Xcontoken* in one of these ways, the operation is disallowed.

User Response: Do not attempt disallowed operations in "No Package Lock" mode. In order to execute the requested operation, "No Package Lock" mode must be exited. This is accomplished by unsetting the DB2_APM_PERFORMANCE environment registry variable. In order for the variable change to take effect, that database manager must be stopped and restarted.

sqlcode: -1803

sqlstate: 57056

SQL1816N **Wrapper** *wrapper-name* **cannot be used to access the** *type-or-version* **of data source (***server-type server-version***) that you are trying to define to the federated database.**

Explanation: The wrapper that you specified does not support the type or version of data source that you want to define.

User Response: Consult the documentation to find out which wrapper supports the type and version of data source that you want to define. Make sure that the wrapper has been registered to the federated database by the CREATE WRAPPER statement. Then recode the CREATE SERVER statement so that it specifies this wrapper, and run the CREATE SERVER statement again.

sqlcode: -1816

sqlstate: 560AC

SQL1817N **The CREATE SERVER statement does not identify the** *type-or-version* **of data source that you want defined to the federated database.**

Explanation: When a CREATE SERVER statement references the wrapper that you specified, it must also identify the *type-or-version* of data source that is to be defined to the federated database.

User Response: In the CREATE SERVER statement, code the *type-or-version* option so that it designates the *type-or-version* of data source being defined. Then run the CREATE SERVER statement again.

sqlcode: -1817

sqlstate: 428EU

SQL1818N **The ALTER SERVER statement that you submitted could not be processed.**

Explanation: The ALTER SERVER statement is preceded in a unit of work by a SELECT statement that references a nickname for a table or view within the data source (or category of data sources) that the ALTER SERVER statement references.

User Response: Let the unit of work finish; then resubmit the ALTER SERVER statement.

sqlcode: -1818

sqlstate: 55007

SQL1819N **The DROP SERVER statement that you submitted could not be processed.**

Explanation: The DROP SERVER statement is preceded in a unit of work by a SELECT statement that references a nickname for a table or view within the data source (or category of data sources) that the DROP SERVER statement references.

User Response: Let the unit of work finish; then resubmit the DROP SERVER statement.

sqlcode: -1819

sqlstate: 55006

SQL1820N **Action on the LOB value failed. Reason code =** *reason-code*.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Set the "deferred_lob_retrieval" to "N" in SYSSERVEROPTIONS, resubmit your query, or ignore the warning.

sqlcode: +1821

sqlstate: 01621

SQL1822N **Unexpected error code** *error-code* **received from data source** *data-source-name*. **Associated text and tokens are** *tokens*.

Explanation: While referencing a data source, the federated server received an unexpected error code from the data source that does not map to a DB2 equivalent.

Possible error codes include:

- 4901 Trying to open more than 15 cursors

- 4902 Row size exceeds 32K limit

This error might also be returned if the data source is not available.

User Response: Ensure the data source is available, then 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.

sqlcode: -1822

sqlstate: 560BD

SQL1823N **No data type mapping exists for data type** *data-type-name* **from server** *server-name***.**

Explanation: 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 the federated server. The name of (one of) the unknown type(s) is listed in the message.

User Response: Create a mapping for the specified type name on the specified server using the CREATE TYPE MAPPING statement.

sqlcode: -1823

sqlstate: 428C5

SQL1824W **Some base tables in the operands of this UNION ALL may be the same table.**

Explanation: 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.

User Response: 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 a federated environment.**

Explanation: Due to some limitation, the current SQL statement cannot be handled in a federated environment. Possible limitations include:

- A cursor update/delete statement involves a refetch of a column into the federated server that is not being fetched in the cursor select statement.
- Some internal programming limitation is exceeded.

User Response: 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***.**

Explanation: An invalid value *value* was specified for a column *column-name* in a system catalog object *object-name*.

The statement cannot be processed.

User Response: Refer to the *SQL Reference* for valid values for the specified column in the specified system catalog object. Correct the statement and reissue it.

sqlcode: -1826

sqlstate: 23521

SQL1827N **No user mapping is defined from local authorization ID** *auth-ID* **to server** *server-name***.**

Explanation: An attempt was made to drop or alter a user mapping that is not defined.

User Response: 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***.**

Explanation: An attempt was made to drop or alter a server option that is not defined.

User Response: If it is an ALTER SERVER statement, first create a server option using the CREATE SERVER statement. Then, alter the server option. If it is a DROP SERVER statement, no extra action is needed because the server option for the servers does not exist.

sqlcode: -1828

sqlstate: 42704

SQL1830N **RETURNS clause must be specified prior to a predicate specification using the EXPRESSION AS clause.**

Explanation: The RETURNS clause is not specified before the PREDICATE clause that includes the EXPRESSION AS clause. The RETURNS clause may have been included after the predicate specification or may be missing.

The statement cannot be processed.

User Response: Specify the CREATE FUNCTION statement with the RESULTS clause prior to the PREDICATE clause.

sqlcode: -1830

sqlstate: 42627

SQL1831N **Table statistics for a subtable** *subtable-name* **cannot be updated.**

Explanation: The statement attempts to update the statistics values of NPAGES, FPAGES, or OVERFLOW, for table *subtable-name,* that is defined as a subtable. For typed tables, these statistics can only be updated using the root table of the table hierarchy.

The statement cannot be processed.

User Response: Update the catalog statistics for the root table of the table hierarchy instead of the subtable.

sqlcode: -1831

sqlstate: 428DY

SQL1832N **The routine** *routine-name* **cannot be used to define a filter because it is defined as an SQL function.**

Explanation: The routine (function or method) *routine-name* is specified in a FILTER clause for a user defined predicate specification or an index extension definition. The routine can not be defined with LANGUAGE SQL.

The statement cannot be processed.

User Response: Specify a routine that is not defined with LANGUAGE SQL.

sqlcode: -1832

sqlstate: 429B4

SQL1881N *option-name* **is not a valid** *option-type* **option for** *object-name***.**

Explanation: The specified option might not exist, or it might not be valid for the particular data source, data source type, or database object that you're working with.

User Response: Consult the SQL Reference to verify the option that you want. Then revise and resubmit the statement that you want to run.

sqlcode: -1881

sqlstate: 428EE

SQL1882N   **The** *option-type* **option** *option-name* **cannot be set to** *option-value* **for** *object-name*.

Explanation: The value that you specified either lacks the proper delimiters or is invalid.

User Response: Consult the SQL Reference to verify the value that you want. Then revise and resubmit the statement that you want to run. Be sure to delimit the value in single quotes.

sqlcode: -1882

sqlstate: 428EF

SQL1883N   *option-name* **is a required** *option-type* **option for** *object-name*.

Explanation: You did not specify an option that DB2 requires in order to process the statement that you submitted.

User Response: Consult the documentation to find out the options that are required for the statement that you want to run. Then revise and resubmit this statement.

sqlcode: -1883

sqlstate: 428EG

SQL1884N   **You specified** *option-name* **(a** *option-type* **option) more than once.**

Explanation: You entered a statement that references the same option multiple times.

User Response: Recode the statement so that it references the option that you want only once. Then resubmit the statement.

sqlcode: -1884

sqlstate: 42853

SQL1900 - SQL1999

SQL1885N   **The** *option-type* **option** *option-name* **is already defined.**

Explanation: You entered a value for an option that already has a value.

User Response: Query the appropriate catalog view to determine what value the option is currently set to. If this value differs from the one that you want, recode the statement so that the SET keyword follows the OPTIONS keyword. To find out what catalog view contains the values for this option, consult the SQL Reference.

sqlcode: -1885

sqlstate: 428EH

SQL1886N   **The** *operation-type* **operation is not valid because the** *option-type* **option** *option-name* **has not been defined.**

Explanation: You tried to change or delete a value for an option that has not been defined for the data source, data source type, or database object that you're working with.

User Response: If you specified SET in the statement that you want to run, recode the statement, either omitting SET or replacing it with ADD (ADD is the default). Then resubmit the statement. If you specified DROP, do nothing.

sqlcode: -1886

sqlstate: 428EJ

SQL1887N   **SPECIFICATION ONLY clause required.**

Explanation: A CREATE INDEX statement for a nickname must have a SPECIFICATION ONLY clause.

User Response: Add the SPECIFICATION ONLY clause and then resubmit the statement.

sqlcode: -1887

sqlstate: 42601

SQL1900N The command completed successfully.

Explanation: The command line utility successfully completed the command.

User Response: No action is required.

SQL1901N The syntax of the command is not correct.

Explanation: The command line utility could not process the command.

SQL2000 - SQL2099

SQL2000N The drive specified in the utility command is not a valid diskette drive or fixed disk.

Explanation: The input or output drive specified in the utility command does not exist.

The utility stops processing.

User Response: Resubmit the utility command with the correct drive specifier.

SQL2001N The utility was interrupted. The output data may be incomplete.

Explanation: The user may have pressed the interrupt key sequence, or had called the utility with a terminate caller action.

This message can also be returned during a backup or restore operation on a database node when the database catalog node is down.

The utility stops processing.

User Response: Restart the application or reissue the command, if needed. The output data from the interrupted command may be incomplete, and should not be used.

User Response: Correct and resubmit the command.

SQL2002N The specified database utility command is not valid for remote databases. The database specified in the command must reside on your local workstation.

Explanation: The database utility command is only valid for local databases.

The utility stops processing.

User Response: Run the utility locally.

SQL2003C A system error occurred.

Explanation: An operating system error occurred. The return code can be found in the SQLERRD[0] field of the SQLCA.

The utility stops processing.

User Response: 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.

Explanation: The utility was using an SQL statement when an error occurred.

The utility stops processing.

User Response: 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.**

Explanation: Incomplete data was read during an I/O operation.

The utility stops processing.

User Response: Determine if the I/O error can be corrected and resubmit the command.

SQL2006C **An I/O error occurred during a write operation. The data may be incomplete.**

Explanation: Incomplete data was written during an I/O operation.

The utility stops processing.

User Response: Determine if the I/O error can be corrected and resubmit the command.

SQL2007N **Specified buffersize of** *buffersize* **4K buffers is too small for** *pagesize* **page size object.**

Explanation: To backup a *pagesize* page size database object one needs a buffer larger than the page size. When backing up a database the data is first copied to an internal buffer. Data is then written from this buffer to the backup media when the buffer is full. The buffersize of *buffersize* 4K buffers that was specified is inadequate.

User Response: Use a larger buffersize.

SQL2008N **The callerac parameter is not within valid range or the requested action is out of sequence.**

Explanation: 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.

User Response: Each utility has its own list of valid *callerac* values. Check the *Application Development Guide* 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.**

Explanation: More memory is needed to run the specified utility.

The command cannot be processed.

User Response: Possible actions include:

• Increase the UTIL_HEAP_SZ database configuration parameter. For backup and restore, this value must be at least as high as the number of buffers * buffer size. For more information on backup and restore buffers, refer to the *Command Reference*.

• Verify that your system has sufficient real and virtual memory.

• Remove background processes.

• Increase the DBHEAP database configuration parameter.

SQL2010N **The utility encountered an error** *error* **while connecting to the database.**

Explanation: The utility could not connect to the database.

The utility stops processing.

User Response: 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.**

Explanation: The utility could not disconnect from the database.

The utility stops processing.

User Response: Look at the error number in the message for more information. Make changes and resubmit the command.

SQL2012N   **Interrupt handling could not be enabled.**

Explanation: 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.

User Response: 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.**

Explanation: An error occurred while the utility was accessing the database directory.

The utility stops processing.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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. The name must be located at a valid address for the application.

The utility stops processing.

User Response: Resubmit the command with a valid database name.

SQL2016C   **The PATH command does not contain a path to** *program-name***.**

Explanation: The utility could not find the required program using the operating system Select Path.

The utility stops processing.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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 or 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: The read or write operation against the log control file failed.

The utility stops processing.

User Response: 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 file type *file-type*.

Explanation: An I/O error occurred while accessing the specified file.

A file with the extension ".BRG" is used to determine whether a RESTORE operation has terminated abnormally. The file resides in the local database directory of the database for which the RESTORE operation was attempted.

A file with the extension ".BRI" stores information about the progress of an incremental RESTORE operation. The file resides in the local

database directory of the database for which the RESTORE INCREMENTAL operation was attempted.

The file name consists of the database token concatenated with the file type extension. For example, if the database "SAMPLE" is assigned the database token "SQL00001", the BRI file will have the name "instance/NODE0000/sqldbdir/SQL00001.BRI".

The utility stops processing.

User Response: 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***.**

Explanation: An I/O error occurred while accessing a file on the specified media.

The utility stops processing.

User Response: 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.**

Explanation: The internal disconnect command failed. The SQLCODE is returned in the message.

The utility stops processing.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The utility could not enable the interrupt handler. The SQLCODE is returned in the message.

The utility stops processing.

User Response: 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.**

Explanation: An error was returned from either the specified command file, or from the operating system.

User Response: 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.**

Explanation: The specified drive does not have enough space for creating an internal subdirectory and information file.

The utility stops processing.

User Response: Free the indicated space on the specified drive and resubmit the command.

SQL2031W **Warning! Please mount the target or source media on device** *device***.**

Explanation: 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.

User Response: 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.**

Explanation: The parameter is incorrectly specified. Either the value is out of range or is incorrect.

User Response: Resubmit the command with a correct value for the parameter.

SQL2033N **The TSM error *error* occurred.**

Explanation: During the processing of a database utility, TSM was called and encountered an error.

User Response: Consult the TSM documentation for a description of the error, take corrective actions and resubmit the command.

SQL2034N **The address of the *parm* parameter is not valid.**

Explanation: 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.

User Response: 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.**

Explanation: The calling application invoked the utility with *no interrupt* mode. During the operation a warning condition was encountered.

User Response: 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 **The path for the file or device *path/device* is not valid.**

Explanation: 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.

User Response: Reissue the utility command with a path that represents a correct path or device.

SQL2037N **TSM could not be loaded.**

Explanation: A call to a database utility specified TSM as the target or source of the backup. An attempt was made to load the TSM client. Either the TSM client is not available on the system or an error was encountered in the load procedure.

User Response: Ensure that the system has TSM available. Resubmit the command after TSM is made available, or resubmit the command without utilizing TSM.

SQL2038N **A database system error *errcode* occurred during processing.**

Explanation: A database system error was encountered during the processing of one of the utilities.

User Response: 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.**

Explanation: 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.

User Response: Reissue the command after determining why the application terminated.

SQL2040N **The database alias parameter** *dbalias* **is not valid or not specified.**

Explanation: 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.

User Response: 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 8 and 16384 inclusive.**

Explanation: 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*. **Additional information (if available):** *additional-information*. **Please ensure media is mounted and positioned correctly.**

Explanation: 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 message may contain additional information to help diagnose the problem.

The utility waits for a response to continue.

User Response: Correctly mount and position the tape and return to the utility indicating if

processing continues or ends.

The error, device and the additional information (if present) can be used to diagnose and correct the problem.

SQL2043N **Unable to start a child process or thread.**

Explanation: Unable to start up the child processes or threads required during the processing of a database utility. There may not be enough available memory to create the new process or thread. On AIX-based systems, the maxuproc value set by the chdev command may be too low. On OS/2-based systems, the THREADS value set in CONFIG.SYS may be too low. The utility stops processing.

User Response: 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*

Explanation: 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.

User Response: 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! Error *error* **occurred while writing to media** *media***.**

Explanation: A database utility process encountered error *error* returned by the operating system 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.

User Response: Consult the problem determination documentation for your operating system and correct the *error* condition. 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***.**

Explanation: 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.

8 Trying to load/quiesce into a table that is not defined on this partition.

The utility stops processing.

User Response: 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, 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.

Explanation: The backup or copy image in use is corrupted

The utility stops processing.

User Response: 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***.**

Explanation: A database utility was unable to access memory during processing.

The utility stops processing.

User Response: Stop the database manager, then restart it and resubmit the utility command.

SQL2056N   An invalid media type was encountered on media *media*.

Explanation: An invalid media type was encountered during the processing of a database utility.

The utility stops processing.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: An end-of-media warning was encountered during the processing of a database utility. This error can also occur if an invalid tape device block size was specified.

The utility is waiting for a response to continue.

User Response: 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.

The tape device block size (or blocking factor) used at restore time must be the same as that used during the backup. If a variable block size is used, then the buffer size used must be less than or equal to the maximum block size of the tape device.

SQL2059W   A device full warning was encountered on device *device*.

Explanation: An device full warning was encountered during the processing of a database utility.

The utility is waiting for a response to continue.

User Response: 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.**

Explanation: An empty device was encountered during the processing of a database utility. The utility is waiting for a response to continue.

User Response: 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.**

Explanation: An attempt to access a device, file, TSM or the vendor shared library is denied during the processing of a database utility. The utility stops processing.

User Response: Ensure the device, file, TSM 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*

Explanation: An unexpected error occurred while accessing a device, file, TSM 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, TSM or the vendor shared library failed.

2   An attempt to terminate a device, file, TSM or the vendor shared library failed.

other If you are using TSM, this is an error code returned by TSM.

The utility stops processing.

User Response: Ensure the device, file, TSM 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.**

Explanation: The caller action specified is not allowed because there is only one device left connected to the database utility.

User Response: 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 utility operation.**

Explanation: The table space name specified is syntactically correct, but it does not exist in the database or cannot be used in an utility operation. If the utility in use is a backup operation, the table space may be disallowed because it is a system or user temporary table space or is in an inconsistent state.

User Response: Check the table space name and resubmit the utility command with the correct table space name.

SQL2068N **An invalid image was encountered on media** *media*. **There was no media header.**

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: Resubmit the command with correct backup or copy images.

SQL2070N **An invalid image was encountered on media** *media*. **The image contained timestamp** *timestamp*.

Explanation: 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.

User Response: 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*

Explanation: 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.

User Response: 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*.

Explanation: An error was encountered while binding the shared library during the processing of a database utility. The utility stops processing.

User Response: 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.

SQL2073N **DATALINK processing failed because of internal problems at the database server or DB2 Data Links Manager.**

Explanation: An unexpected error occurred while processing DATALINK values.

User Response: Resubmit the command. If the problem still exists, resubmit the command after shutdown and restart of DB2 and the DB2 Data Links Managers.

The Restore utility can avoid DATALINK processing by specifying WITHOUT DATALINK.

SQL2074N **DATALINK processing failed because of internal problems at the database server.**

Explanation: An unexpected error occurred while processing DATALINK values.

User Response: Resubmit the command. If the problem still exists, resubmit the command after shutdown and restart of DB2.

The Restore utility can avoid DATALINK processing by specifying WITHOUT DATALINK.

SQL2075N **DATALINK processing failed because of internal problems at the DB2 Data Links Manager.**

Explanation: An unexpected error occurred while processing DATALINK values.

User Response: Resubmit the command. If the problem still exists, resubmit the command after

shutdown and restart of the DB2 Data Links Managers.

The Restore utility can avoid DATALINK processing by specifying WITHOUT DATALINK.

SQL2076W **The DB2 Data Links Manager** *server-name* **was not registered to the database.**

Explanation: The DB2 Data Links Manager *server-name* was not registered to the database with the ADD DATALINKS MANAGER command.

User Response: For details on why the ADD DATALINKS MANAGER command failed, check the administration notification log.

SQL2077W **Reconcile processing completed successfully on Data Links Managers that were available. Reconcile processing is pending on Data Links Managers that were not available. Examine the administration notification log for more information.**

Explanation: Some or all of the Data Links Managers referred to in the table data were not available during reconcile processing. Reconcile processing completed successfully on Data Links Managers that were available. The table is placed in DataLink Reconcile Pending (DRP) state, because reconcile processing is pending on Data Links Managers that were not available.

User Response: The table will be taken out of DataLink Reconcile Pending state when reconcile is completed successfully on all Data Links Managers referred to in the table data. Bring up the Data Links Managers that were not available and run reconcile again.

SQL2078N **The DB2 Data Links Manager could not be added or dropped successfully. Reason code =** *reason-code*.

Explanation: The DB2 Data Links Manager could not be added or dropped, as indicated by

the one of the following reason codes:

01 The DB2 Data Links Manager being added is already registered to the database.

02 The DB2 Data Links Manager being dropped is not registered to the database.

03 The database manager configuration parameter DATALINKS is set to NO.

04 The database already has the maximum allowed number of DB2 Data Links Managers registered to it.

User Response: The action depends on the reason code as follows:

01 Do not attempt to add the DB2 Data Links Manager more than once.

02 Do not attempt to drop a DB2 Data Links Manager which is not registered.

03 Set the database manager configuration parameter DATALINKS to YES using the UPDATE DATABASE MANAGER CONFIGURATION command and attempt the operation again.

04 Do not attempt to add more than the maximum allowed number of DB2 Data Links Managers.

SQL2079N **An error was reported by the shared library** *shr-lib-name*. **Return code:** *return-code*.

Explanation: An error was reported from a vendor shared library during the processing of a database utility. The return code displayed corresponds to one of the valid vendor API return codes. The utility stops processing.

Possible API return codes are:

3 The DB2 and vendor products are incompatible.

4 An invalid action was requested.

8 An invalid userid was specified.

9 An invalid password was provided.

10 Invalid options were specified.

12 An invalid device handle was provided.

13 An invalid buffer size was specified.

30 A severe error was experienced inside the vendor product.

other A specific severe error returned by the vendor product.

User Response: Ensure the shared library provided is valid. Attempt to make the necessary corrective actions based on the vendor API return code and the corresponding db2diag entry and resubmit the utility command.

SQL2100 - SQL2199

SQL2150W **The table spaces contained in the backup image were restored. Depending on the request of the caller, one or more of these table spaces might have been skipped.**

Explanation: A RESTORE DATABASE command was issued. The user might have indicated that only a subset of the table spaces in the backup image were to be recovered.

User Response: No action is required.

SQL2154N **The RESTORE was not successful. The backup used to restore the table space(s) is not associated with the current log sequence for the database.**

Explanation: For table space 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.

Table space restore has stopped.

User Response: Resubmit the command with a correct backup image.

SQL2155W **Changes have been made to the recovery history file since the open scan was issued.**

Explanation: Changes have been made to the recovery history file since the file was open for a scan. The data read may not be consistent.

User Response: 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.**

Explanation: Eight recovery history file scans in this process are already open. No more than eight open scans are allowed.

The command cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

SQL2162N **Rollforward operation failed because DB2 does not have permission to access log file** *log-file-number* **for database** *database-name* **on node** *node-number*.

Explanation: DB2 cannot access the specified log file because of the file permission settings. Rollforward cannot continue.

User Response: Check the file system where the log files are stored. Make sure sufficient permission is given to the instance owner to access these log files.

SQL2165W **The SQLUHINFO structure did not provide enough TABLESPACE entries.**

Explanation: 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.

User Response: 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.

SQL2170N   The utility encountered entries with the same identifier in the recovery history file. Write fails.

Explanation: The utility has encountered an entry with the same identifier (timestamp to the second) in the recovery history file while writing to it. The write to the recovery history file terminates. Database manager ensures the uniqueness of the recovery history file identifiers and makes provisions for it even if there are more than one request per second. However, there may be cases where it will fail if numerous requests come in within a few seconds.

User Response: If the applications are running utilities (such as backup, quiesce, load) that generate numerous entries to the history file, try to pace the utility requests to prevent the problem.

SQL2171N   The update of the recovery history file failed because the specified object part does not exist in the file.

Explanation: The entry specified to be updated in the recovery history file does not exist in the file. The utility stops processing.

User Response: Resubmit the command with a valid entry.

SQL2172W   The utility has completed but it is not able to log the event in the recovery history file due to error *error*.

Explanation: The utility has encountered an error writing to the recovery history file. This warning does not affect processing.

User Response: Correct the error condition to prevent future warnings.

SQL2180N   Incorrect syntax or incorrect password key used in filtering specification.

Explanation: The filtering specification provided uses one or both of incorrect syntax or an incorrect password key.

User Response: Save the diagnostic information. Contact IBM service.

SQL2181N   An internal error occurred during filtered recovery.

Explanation: An internal error occurred during filtered recovery. Recovery is terminated.

User Response: If the error was on an attempt to change the state for a table, try filtering out the full tablespace. Save the diagnostic information. Contact IBM service.

SQL2200 - SQL2299

SQL2200N   The qualifier for the table or index name is too long.

Explanation: The authid must be 1 to 30 characters (bytes in MBCS environment).

The utility stops processing.

User Response: Resubmit the command with 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.

Explanation: The table name is required and it must be 1 to 128 characters in length (bytes in MBCS environments). The table name must be

located at a valid application address.

The utility stops processing.

User Response: 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.**

Explanation: If an index is specified, the name must be 1 to 18 characters (bytes in MBCS environments). The index must be located at a valid application address.

The utility stops processing.

User Response: Resubmit the command with a valid index name.

SQL2205N **The specified index is not valid. Reason code:** *reason-code*.

Explanation: The specified index parameter is not valid for the requested operation. Following is a list of reason codes:

1 The specified index does not exist for the specified *schema.table-name* or *schema.index-name*.

2 The specified index is an extended index. The Reorganize Table utility does not support indexes based on index extensions.

3 All indexes defined on the table must be pseudo delete enabled for the following operations: Inplace Table REORG, Online LOAD in incremental mode, REORG INDEXES with the CLEANUP ONLY option.

4 An index other than the clustering index was specified on the REORG command.

5 Inplace table REORG is not allowed if any spatial indexes exist on the table.

6 An index cannot be specified for a REORG of a table that has one or more

dimensions using the ORGANIZE BY clause of the CREATE TABLE command.

Federated system users: this situation can also be detected by the data source.

The utility or operation stops processing.

User Response: Resubmit the command with a valid index or without an index.

SQL2207N **The filepath specified by the datafile parameter is not valid.**

Explanation: 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.

User Response: Resubmit the command with a valid datafile parameter.

SQL2208N **The table space specified by the table space parameter is not valid.**

Explanation: 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.

User Response: 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.

Explanation: The table does not exist in the database. Either the table name or the authorization ID is incorrect.

The command cannot be processed.

User Response: 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.

Explanation: The Reorganize Table utility cannot be run against a view.

The utility stops processing.

User Response: Resubmit the command with a valid table name.

SQL2213N   The specified table space is not a system temporary table space.

Explanation: The REORG utility requires that any table space specified is a system temporary table space. The table space name provided is not a table space defined to hold system temporary tables.

The utility stops processing.

User Response: Resubmit the command with the name of a system temporary table space or do not use the table space name parameter. In the latter case, the REORG utility will use the table space(s) in which the table itself resides.

SQL2214N   The user does not have the authority to run the REORG utility on table *name***.**

Explanation: The user attempted to reorganize the specified table or its indexes without the appropriate authorization (either SYSADM or DBADM authority or the CONTROL privilege on the table).

The utility stops processing.

User Response: Logon as a user with the appropriate authority or privilege and resubmit the REORG utility command.

SQL2215N   SQL error *sqlcode* **occurred while committing previous work for the database.**

Explanation: 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.

User Response: 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 or its indexes.**

Explanation: An error occurred during the reorganization of a database table or its indexes.

The utility stops processing.

User Response: Look at the SQLCODE (message number) in the message for more information. Make changes and resubmit the command.

SQL2217N   The page size of the system temporary table space used by the REORG utility must match the page size of the table space(s) in which the table data resides (including the LONG and/or LOB column data).

Explanation: If the system temporary table was explicitly specified to the REORG utility then the page size of the system temporary table space used by the REORG utility must match the page size of the table space or table spaces in which the table data, including LONG and/or LOB column data, resides. One of the following has violated this restriction:

- The table's data resides in a table space that has a different page size than that of the specified system temporary table space.
- The table contains LONG and/or LOB columns whose data resides in a table space with a page size that is different than that of the system temporary table space and the table's regular data.

If the system temporary table space was not specified to the REORG utility then the utility was looking internally for a system temporary table space. A system temporary table space that uses the same page size as the table data either did not exist in the database or was not available at the time.

User Response: If the table being reorganized resides in a table space with a page size which is different than that of the table space in which the LONG and/or LOB data of the table resides, then a system temporary table space cannot be provided to the REORG utility. Reissue the REORG request without specifying a system temporary table space.

If a system temporary table space that uses the same page size as the table data does not exist in the database, please create a system temporary table space using a page size that matches the page size of the table data.

If a system temporary table space that uses the same page size as the table data exists in the database but was not available at the time of your command, please reissue the command when the system temporary table space is available.

SQL2218N **One or more parameters specified on the reorg command is incompatible.**

SQL2219N **The specified INPLACE table reorganization action on table** *table-name* **is not allowed on one or more nodes. Reason code:** *reason-code***.**

Explanation: The statement cannot be processed because it violates a restriction as indicated by

the following reason code:

1 Table reorganization cannot RESUME as the original data or index object has changed.

2 There is no reorganization to RESUME for the specified table.

3 An index cannot be specified to RESUME a non-clustering table reorganization.

4 Table reorganization cannot RESUME as the original index specified to START the reorganization has been dropped.

5 The index specified on RESUME is not the same index that was specified when the table reorganization was paused.

6 Unable to RESUME table reorganization as the Status File is corrupted or missing.

7 An I/O error occurred when attempting to access the Status File.

8 INPLACE table reorganization is already in progress for the specified table.

9 START is not allowed as a prior paused table reorganization has not been stopped.

10 PAUSE or STOP has already been issued. The specified action is asynchronous and may not be effective immediately.

11 There is no table reorganization to PAUSE or STOP.

12 INPLACE table reorganization is not allowed for tables in append mode.

User Response:

1 Issue a STOP then a START.

2 There is no outstanding table reorganization to RESUME. Issue the START action to reorganize the specified table.

3 The original table reorganization did not use an index and therefore an index

cannot be specified on RESUME. Either continue the reorganization without using an index or STOP the existing reorganization and issue a START with an index name.

4   STOP the existing paused table reorganization and issue a START.

5   Resubmit the RESUME command with the correct index. If no index is specified on RESUME the original index will be used by default.

6   Confirm whether the Status File "<tablespaceID><objectID>.OLR" exists under the database directory and is accessible then resubmit the command. If the file is damaged STOP then START the table reorganization.

7   Ensure the Status File

"<tablespaceID><objectID>.OLR" is accessible and resubmit the command.

8   START or RESUME is not allowed as table reorganization is already in progress.

9   A reorg on the table has been paused. Either RESUME the paused reorg or STOP then START the reorg.

10   Wait for the original STOP or PAUSE to complete.

11   Ensure reorganization is running for the specified table.

12   Either ALTER the table to turn APPEND mode OFF or reorganize the table offline.

SQL2300 - SQL2399

SQL2300N   **The identifier for the table name is too long or it was not specified as part of the table name.**

Explanation: The table name must be fully qualified. The format is *authid.tablename* where *authid* contains 1 to 30 characters and *tablename* contains 1 to 128 characters (bytes in MBCS environments).

The utility stops processing.

User Response: 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.**

Explanation: The table name must be fully qualified and in the format *authid.name* where *authid* contains 1 to 30 characters and *name* contains 1 to 128 characters (bytes in MBCS environments) and must be located at a valid application address.

The utility stops processing.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the command with a valid index list.

SQL2303N   **The statsopt parameter is not valid.**

Explanation: 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.

User Response: 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.**

Explanation: The *sharelev* parameter in the RUN STATISTICS command must be R for reference or C for change.

The utility stops processing.

User Response: Resubmit the command with a valid *sharelev* parameter.

SQL2305N **The table specified is a view. The utility cannot be run against a view.**

Explanation: The *tname* parameter specified a view rather than a table.

The utility stops processing.

User Response: Resubmit the command with a valid *tname* parameter.

SQL2306N **The table or index *name* does not exist.**

Explanation: 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.

User Response: 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.**

Explanation: The Run Statistics utility command may not be run against a system table.

The utility stops processing.

User Response: 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.**

Explanation: The index name must be a fully qualified name. The format is *authid.name* where *authid* contains 1 to 30 characters and *name* contains 1 to 18 characters (bytes in MBCS environments).

The utility stops processing.

User Response: 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.**

Explanation: The index name must be a fully qualified name. The format is *authid.name* where *authid* contains 1 to 30 characters and *name* contains 1 to 18 characters (bytes in MBCS environments).

The utility stops processing.

User Response: Resubmit the command with a valid index.

SQL2310N **The utility could not generate statistics. Error *sqlcode* was returned.**

Explanation: An error occurred while the utility was gathering statistics.

The utility stops processing.

User Response: 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***.**

Explanation: 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). If the table specified in RUNSTATS is the root table of a table hierarchy, then the table name returned in the message may be a subtable of the specified root table.

The utility stops processing.

User Response: 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.**

Explanation: The setting of the database configuration parameter *stat_heap_sz* is not big enough to collect non-uniform distribution statistics on the table.

User Response: 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.**

Explanation: 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.

User Response: Update the database configuration parameter *stat_heap_sz* by 20 percent and try again.

SQL2314W **Some statistics are in an inconsistent state. The newly collected** *object1* **statistics are inconsistent with the existing** *object2* **statistics.**

Explanation: Issuing RUNSTATS on the table only may result in a situation where the table level statistics are inconsistent with the already existing index level statistics. For example, if index level statistics are collected on a particular table and later a significant number of rows is deleted from this table, issuing RUNSTATS on the table only may end up with the table cardinality less than FIRSTKEYCARD which is an inconsistent state. Likewise, issuing RUNSTATS for indexes only or collecting statistics during index creation may leave the already existing table level statistics in an inconsistent state. For example, if table level statistics are collected on a particular table and later a significant number of rows is deleted from this table, issuing RUNSTATS for the indexes only or collecting statistics during index creation may end up with some columns having a COLCARD greater than the table cardinality.

User Response: If a RUNSTATS was issued on the table only, issue a RUNSTATS for the indexes as well so that table level and index level statistics are consistent. Likewise, if a RUNSTATS was issued for indexes only or statistics were collected during index creation, issue a RUNSTATS on the table as well.

sqlcode: 2314

sqlstate: 01650

SQL2400 - SQL2499

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.**

Explanation: The type must be either 0 for backing up the whole database or 1 for backing up changes only.

The utility stops processing.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the utility command with the correct type.

SQL2403N **The user does not have the authority to run the utility against the database.**

Explanation: The user attempted to execute a database utility without either SYSADM or DBADM authority.

The utility stops processing.

User Response: 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.**

Explanation: The backup target media does not have enough space for creating an internal subdirectory and information file.

The utility stops processing.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the RESTORE command.

SQL2406N **The BACKUP cannot be performed because the database needs to be rolled forward.**

Explanation: The backup failed because the database is in an inconsistent state.

The utility stops processing.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: The BACKUP command ran successfully. The indicated file was not deleted due to an I/O error.

The utility completed processing.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: During the utility operation a corrupted database page was encountered. The database is in an unpredictable state and the utility cannot continue.

User Response: Record the message number (SQLCODE).

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The tape which the utility is using has become full.

User Response: Mount another tape and continue the operation with the *callerac* parameter indicating if processing continues.

SQL2417N **Archive log is not allowed on non-recoverable databases.**

Explanation: The archive log command can only be used on databases that are in recoverable mode. A database is in recoverable mode if either LOGRETAIN or USEREXIT are enabled.

User Response: Ensure that the specified database is in recoverable mode and reissue the command.

SQL2418N **The database specified for Backup does not exist.**

Explanation: The database specified in the *dbase* parameter of the Database Backup command could not be found.

User Response: 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 *disk* has become full.**

Explanation: During the processing of a database utility, the target disk became full. The utility has stopped and the target is deleted.

User Response: 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.

On non unix-based systems, this disk full condition may be due to exceeding the maximum file size allowed for the operating system. Direct the target to other media, such as tape, or use multiple targets.

SQL2420N **Insufficient space on device** *device* **to contain initial backup control information.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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 table space locations.**

Explanation: This database contains table space(s) which have been defined in a location other than the database subdirectory. This is not compatible with the API being used.

User Response: Use the current backup API to backup the database.

SQL2423N **The database cannot be backed up because some index files are missing.**

Explanation: Some of the index files required by the backup are missing. These index files must be recreated prior to backing up the database.

User Response: Run the 'db2recri' program to recreate the missing index files and then resubmit the backup command.

SQL2424N **The backup could not succeed because asynchronous copy operations at DB2 Data Links Managers are not complete.**

Explanation: The TSM or vendor supplied archive server might not be in operational state.

User Response: Ensure TSM or the vendor supplied archive server is in an operational state and resubmit the backup command.

SQL2425W **The log file for the online backup was not truncated.**

Explanation: During an online backup, all buffered log records are forced to disk and the last active log file is truncated. During the current backup, truncation of the last active log file has failed. As a result, new log records will continue to be written into the last log file used during the backup.

User Response: No user action is required. The last active log file used during the backup will eventually become inactive when it becomes full.

SQL2426N **The database has not been configured to allow the incremental backup operation. Reason code =** *reason-code***.**

Explanation: Incremental backups are not enabled for a table space until after modification tracking has been activated for the database and a non-incremental backup has been performed on the table space.

Possible reason codes:

1. The configuration parameter TRACKMOD has not been set for the database.
2. The TRACKMOD configuration parameter has been set but at least one table space has not had a non-incremental backup taken since the TRACKMOD parameter was set.

User Response: The action is based on the reason code as follows:

1. Activate modification tracking for the database by setting the TRACKMOD database configuration parameter to on, then perform a full database backup.
2. Perform a full backup of the table space.

SQL2500 - SQL2599

SQL2501C   The database was restored but the data in the restored database was unusable.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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***.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Reinstall DB2 or reapply recent updates and resubmit the utility command.

SQL2508N   The Database Restore timestamp parameter *timestamp* **is not valid.**

Explanation: The timestamp parameter must contain either NULL or a portion of a valid timestamp, the portion consisting of complete components of a timestamp.

User Response: Resubmit the Restore operation with a valid timestamp value.

SQL2509N   The database drive parameter is not valid.

Explanation: 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.

User Response: Resubmit the utility command with a valid drive.

SQL2510N   An operating system semaphore error occurred.

Explanation: An error occurred in the wait or post semaphore.

The utility stops processing.

User Response: Stop the database manager, then restart it and resubmit the utility command.

SQL2511N   The utility encountered an error *error* **while dropping the database.**

Explanation: The utility could not drop the database.

The utility stops processing.

User Response: 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.**

Explanation: The utility could not create the
database.

The utility stops processing.

User Response: 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.**

Explanation: The utility could not rename the
database.

The utility stops processing.

User Response: Look at the error number in the
message for more information. Make changes
and resubmit the command.

SQL2514N   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.

Explanation: The release number of the restored
database is not compatible with the release
number of the installed version of the database
manager.

The utility stops processing.

User Response: Resubmit the command with a
backup that is compatible with the release level
of the database manager.

SQL2515N   The user does not have the
authority to run the RESTORE
utility against the database.

Explanation: The user attempted to execute the
RESTORE utility without SYSADM authority.

The utility stops processing.

User Response: 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.

Explanation: 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.

User Response: Wait until no databases on the
workstation are in use and resubmit the
command.

SQL2517W   The restored database was
migrated to the current release.

Explanation: 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.

User Response: No action is required.

SQL2518N   The RESTORE was not successful.
An I/O error occurred while
trying to restore the database
configuration file.

Explanation: The database configuration file
could not be restored due to an I/O error.

The utility stops processing.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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 file type** *file-type***.**

Explanation: The restore process ran successfully. The specified file was not deleted due to an I/O error.

A file with the extension ".BRG" is used to determine whether a RESTORE operation has terminated abnormally. The file resides in the local database directory of the database for which the RESTORE operation was attempted.

A file with the extension ".BRI" stores information about the progress of an incremental

RESTORE operation. The file resides in the local database directory of the database for which the RESTORE INCREMENTAL operation was attempted.

The file name consists of the database token concatenated with the file type extension. For example, if the database "SAMPLE" is assigned the database token "SQL00001", the BRI file will have the name "instance/NODE0000/sqldbdir/SQL00001.BRI".

The utility completed successfully.

User Response: Delete the .BRG or .BRI file manually. If the file cannot be deleted, contact your technical service representative.

SQL2522N **More than one backup file matches the timestamp value provided for the backed up database image.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Return to the utility with the *callerac* parameter indicating processing to continue or end.

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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The backup image being restored is corrupted making the database image non-restorable.

User Response: 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.**

Explanation: 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.

User Response: Determine the location of the correct backup image file, and resubmit the Restore command.

SQL2532N **The backup file contains the backup image of database *dbalias* taken at timestamp *timestamp*. This is not the backup image requested.**

Explanation: The file selected for restore does not contain the requested backup image. The image is for a different database than that requested.

User Response: 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 file on device** *device* **contains the image of database** *database* **taken at timestamp** *timestamp***. This is not the backup image requested.**

Explanation: The backup 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 file sequence.

User Response: Ensure that the tape is positioned at the correct backup, 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.**

Explanation: The data read from the tape position does not contain a valid backup media header.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the Database Restore command without specifying SQLUD_NOROLLFWD for the *rst_type* parameter.

SQL2538N **Unexpected End of File of Backup Image reached on media** *media***.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Ensure that the action that caused this warning to be generated has not resulted in a condition not wanted.

SQL2541W **The utility was successful, however the file containing the backup image could not be closed.**

Explanation: The utility was successful, but the file containing the backup image could not be closed.

User Response: 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** *database-alias* **and timestamp** *timestamp* **provided.**

Explanation: 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. If this error was received from an automatic incremental restore then a required image was not found based on the timestamp and location in the database history.

User Response: Ensure that the database backup image resides on the media source. Resubmit the operation by specifying a correct timestamp to result in a match.

If this error was received from an automatic incremental restore, check the database history for the corresponding backup entry and verify that the location listed matches the actual location of the backup image. Either update the database history and retry the operation to result in a match or issue a RESTORE INCREMENTAL ABORT command to cleanup any resources that may have been created during processing.

SQL2543N **The target directory specified for the database is not valid.**

Explanation: 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.

User Response: Reissue the Backup or Restore command with a valid target directory.

SQL2544N **The directory where the database is being restored has become full.**

Explanation: 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.

User Response: 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 TSM server is currently stored on mountable media. The time required to make it available is unknown.**

Explanation: The backup image is not immediately accessible by the TSM server. The restore process can continue and make the request to the server to retrieve the data. The time required is unknown.

User Response: 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.**

Explanation: 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.

User Response: 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 requires rollforward recovery.**

Explanation: The physical log file formats have changed between these releases, making rollforward impossible.

User Response: 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 code page indicated within the backup image** *code-page* **is invalid or not supported. The restore operation has failed.**

Explanation:

1. The backup image you are restoring may have been created on a server with a newer FixPak level than the FixPak level that has been applied to this server. In this case, it is possible that the image contains a newer code page that is unsupported.

2. The backup image is corrupt and contains invalid code page information.

User Response: If attempting to restore an image from a server with a newer FixPak level to a server with a lesser FixPak level, then ensure that the code page is supported by both servers. Alternatively, consider applying the newer FixPak to the server you are restoring to.

SQL2549N   **The database was not restored because either all of the table spaces in the backup image are inaccessible, or one or more table space names in list of table space names to restore are invalid.**

Explanation: The containers which were used by the table spaces in the backup image are not available, are already in use, or one or more table space names specified in the list on the restore command do not exist in the backup image.

User Response: Use redirected restore to redefine the containers for the table spaces in this backup image, or specify a list of valid table space names to restore.

SQL2550N   **Database backed up on node** *node1* **cannot be restored to node** *node2***.**

Explanation: The backup image used for the restore is a backup of a database from a different node. You can only restore a backup to the same node.

User Response: Ensure that you have the

correct backup image for the node then issue the request again.

SQL2551N **Database with catalog node** *node1* **cannot be restored to a database with catalog node** *node2*.

Explanation: The catalog node can exist on only one node and there is a discrepancy between the backup image and the node being restored to. This can occur in the following cases:

1. The backup image specified catalog node *node1* and the restore was attempted to an existing database whose catalog node is node *node2*

2. The restore was attempted to a new database and the catalog node has not been restored first.

User Response: Verify that the correct backup image is being restored.

If you are restoring to an existing database and want to change the catalog node to *node2*, the existing database must be dropped first.

If you are restoring to a new database, restore the catalog node *node1* first.

SQL2552N **Invalid report file name specified in the restore command.**

Explanation: The length of the report file name exceeded the allowed limit of 255.

User Response: Specify a report file name whose length is within the allowed limit and resubmit the restore command.

SQL2553I **The RECONCILE utility completed successfully.**

Explanation: The utility completed successfully.

User Response: No action necessary.

SQL2554N **The RECONCILE utility failed with reason code** *reason-code*. **Possible problem with** *component*.

Explanation: Possible Reason codes:

1   Unknown connection to a DB2 Data Links Manager.

2   Table/DATALINK column not defined on the DB2 Data Links Manager.

3   DB2 Data Links Manager may be down.

4   I/O problems.

5   The exception table contains datalink columns that are defined with file link control.

6   Table is in "Datalink Reconcile Not Possible" state.

7   Exception table not allowed for reconcile on typed tables.

8   ALTER TABLE failed to put the table into "Datalink Reconcile Pending" or "Datalink Reconcile Not Possible" state, or failed to take the table out of "Datalink Reconcile Pending" or "Datalink Reconcile Not Possible" state.

9   Data Links support is not turned on.

10   Table is in check pending state.

11   A required DB2 Data Links Manager was not available during exception handling. Table is placed in DataLink Reconcile Pending state.

12   Reconcile processing could not complete on the DB2 Data Links Manager(s) in the allotted time period. Check the administration notification log for details.

User Response: Possible Solutions:

1   Make sure that the DB2 Data Links Manager is up and running, and that it has been registered with the database using the ADD DATALINKS MANAGER command. Try connecting to the database and make a sure a corresponding connection has been established on the DB2 Data Links Manager.

2   The table does not seem to exist on the DB2 Data Links Manager. Nothing to reconcile.

3	DB2 Data Links Manager may be down. Try starting the DB2 Data Links Manager.
4	Ensure you have file permissions and enough space for the report file. A fully qualified path is required for the DLREPORT parameter. Ensure that the table being reconciled is not damaged.
5	Redefine all datalink columns in the exception table as "NO LINK CONTROL".
6	Take the table out of "Datalink Reconcile Not Possible" state using the SET INTEGRITY command. Repeat the reconcile command.
7	Do not specify an exception table.
8	Try issuing a SET INTEGRITY command to put the table into "Datalink Reconcile Pending" state or to reset the "Datalink Reconcile Pending" or "Datalink Reconcile Not Possible" state.
9	The value of the database manager configuration parameter DATALINKS is set to NO. To use RECONCILE, you must set the value of the parameter DATALINKS to YES.
10	To run reconcile on a table, the table cannot be in check pending state. To remove the check pending state, use the SET INTEGRITY command.
11	Run reconcile again.
12	The reconcile command has timed out waiting for verification from the DB2 Data Links Manager(s). Try the reconcile command again.

SQL2560N   The target database is not identical to the source database for a restore from a table space level backup.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Return to the utility with the *callerac* parameter indicating processing to continue or end.

SQL2562N   Table space level restore from a full database backup is not allowed.

Explanation: For a restore from a full database level backup, the restore type cannot be table space level.

User Response: 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 were not restored.

Explanation: The restore process has completed successfully. One or more table spaces that are in the backup were not restored for one of the following reasons:

- an error was encountered while accessing the containers of the table spaces. If the table spaces have been dropped since the backup was taken, no further action is necessary.
- the restore command was invoked with a list of table spaces so that only a subset of the table spaces in the backup were restored. No further action is necessary.

User Response: If this message is not due to a subset restore, 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 about recovering the table space.

SQL2564N The pagesize of one or more table spaces in the backup image do not match that of the corresponding destination table spaces.

Explanation: The target table space pagesize must match that of the table space pagesize in the backup image. Restoring to a table space of a different pagesize is not supported. The default pagesize is 4K.

User Response: Ensure that the table space being restored into has the same pagesize as the table space in the backup image.

SQL2565N The option specified for RESTORE is not allowed with the backup image provided.

Explanation: 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.

User Response: Resubmit the command with the correct database name.

SQL2566W The restore process has completed successfully, but one or more table spaces have their tables put in DRP/DRNP state. Check the db2diag.log file for details.

Explanation: One or more table spaces have tables in DATALINK Reconcile Pending (DRP) or DATALINK Reconcile Not Possible (DRNP) because of one of the following reasons:

• Restoring to a database that has a different database name, alias, hostname, or instance than the backup image. If no rollforward follows the restore, any tables with DATALINK columns are placed in DRNP state.

• Restore is done specifying the WITHOUT DATALINK option and no rollforward follows the restore. Any tables with DATALINK columns are placed in DRP state.

• Restoring from a backup image that has become unusable. If no rollforward follows the restore, any tables with DATALINK columns are placed in DRNP state.

• DATALINK column information does not exist at the DB2 Data Links Manager(s). Affected tables are placed in DRNP state.

• Restore involving fast reconciliation failed while trying to relink files at the DB2 Data Links Manager(s). Affected tables are placed in DRNP state.

User Response: Look in the administration notification log to find out what tables are put in DRP/DRNP state. See the Administration Guide for information on reconciling tables that are in DRP/DRNP state.

SQL2570N A database cannot be restored on a platform that does not match the platform on which the backup image was created.

Explanation: You created the backup image on one type of platform and are trying to restore it on a different platform. However, backup images must be restored on a system whose machine type and operating system match the system on which the backup image was created.

User Response: To use this particular backup image, restore it on a system that corresponds to the system on which the backup was taken.

To move a database from one type of platform to another, use the db2move utility described in the *Data Movement Utilities Guide and Reference*.

SQL2571N Automatic incremental restore is unable to proceed. Reason code: *reason-code***.**

Explanation: An error was encountered during the automatic incremental restore process. The utility was unable to complete as intended. The utility stops processing. This error is returned after the initial definitions have been restored and the processing of the required incremental restore set cannot be completed successfully, corresponding to the following reason codes:

1 The backup image corresponding to the specified timestamp could not be found in the database history.

2 An error occurred trying to determine which table spaces to restore.

3 A required backup image could not be found in the database history.

User Response: Issue a RESTORE INCREMENTAL ABORT command to cleanup any resources that may have been created during processing. Perform a manual incremental restore to restore the database from this backup image.

SQL2572N Attempted an incremental restore of an out of order image. The restore of tablespace *tablespace-name* **encountered an error because the backup image with timestamp** *timestamp-value* **must be restored before the image that was just attempted.**

Explanation: When restoring images produced with an incremental backup strategy, restore the images in the following order:

1. Restore the final image first to indicate to DB2 the increment to which you want to restore the database.

2. Restore the full database or tablespace image which precedes the set of incremental images.

3. Restore the set of incremental and delta images in the chronological order in which they were produced.

4. Restore the final image for a second time.

Each tablespace in the backup image is aware of the backup image that must be restored before the backup image that failed can be successfully restored. You must restore the image with the timestamp reported in this message before you can successfully restore the image that invoked this message. There might be additional images to restore before the indicated image, but this was the first tablespace to encounter an error.

User Response: Ensure the order of the set of incremental backup images is correct and continue the incremental restore process.

SQL2573N An incremental backup image must be restored as part of an incremental RESTORE operation.

Explanation: A RESTORE operation was attempted using an incremental backup image. An incremental backup cannot be restored on its own; it can be restored only as part of an incremental RESTORE operation.

The utility stops processing.

User Response: To restore this backup image, reissue the RESTORE command using the INCREMENTAL modifier. To perform a non-incremental RESTORE operation, specify a non-incremental backup image.

SQL2574N A backup image restored as part of an incremental RESTORE operation cannot be newer than the target image.

Explanation: The target image is the first image to be restored as part of an incremental RESTORE operation. This image contains the tablespace definitions and other control structures for the database that is being restored. The RESTORE utility cannot restore an image that is newer than the target image during an incremental RESTORE operation, because it could corrupt the database.

The incremental RESTORE operation failed because it attempted to restore a backup image with a timestamp that is more recent than the timestamp of the target image.

The utility stops processing.

User Response: Reissue the command, specifying a backup image with a timestamp that is older than the timestamp of the target image.

SQL2575N **The timestamp of the specified incremental backup image is older than the timestamp of the last image that was restored for tablespace** *tablespace-number*. **The last backup image has a timestamp of** *timestamp*.

Explanation: To perform an incremental RESTORE operation, the backup images must be restored in chronological order of oldest to newest for each tablespace. An incremental RESTORE operation specified a backup image with a timestamp older than the timestamp of the previous image that was restored for the specified tablespace.

The utility stops processing.

SQL2600 - SQL2699

SQL2600W **The input parameter pointer to the authorization block is invalid or the block's size is incorrect.**

Explanation: 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.

User Response: Correct the value of the input parameter and resubmit the command.

SQL2650N **An invalid parameter** *parameter_name* **was passed to the** *API-name* **read log API. Reason Code =** *reason-code*.

Explanation: An invalid parameter was passed to a read log API. The possible reason codes are as follows:

01 Invalid action specified.

User Response: Reissue the command, specifying a backup image with a timestamp that is newer than the timestamp of the last image that was restored for the tablespace.

SQL2576N **Tablespace** *tablespace-name* **is being restored as part of an incremental RESTORE operation, but the RESTORE command did not specify the INCREMENTAL clause.**

Explanation: To incrementally restore a tablespace, each RESTORE command must specify the INCREMENTAL clause.

The utility stops processing.

User Response: Reissue the RESTORE command and include the INCREMENTAL clause.

02 The starting log sequence number is greater than 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.

10	The filter option is not valid.
11	The logfile path is not valid.
12	The overflow logfile path is not valid.
13	The retrieve logs option is not valid.
14	The specified number of bytes for memory allocation is too small.

User Response: For reason code 01, ensure that the action is either DB2READLOG_QUERY, DB2READLOG_READ, DB2READLOG_READ_SINGLE.

For reason code 02, reduce the value of the starting log sequence number.

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 the next start 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.

For reason code 10, ensure that the filter option is either DB2READLOG_FILTER_OFF or DB2READLOG_FILTER_ON.

For reason code 11, ensure that the logfile path field is defined and points to a valid directory.

For reason code 12, ensure that the overflow logfile path field is defined and points to a valid directory.

For reason code 13, ensure that the retrieve option is either DB2READLOGNOCONN_RETRIEVE_OFF, DB2READLOGNOCONN_RETRIEVE_LOGPATH, or DB2READLOGNOCONNRETRIEVE_OVERFLOW.

For reason code 14, increase the amount of bytes that may be allocated internally.

SQL2651N   The log records associated with the database can not be asynchronously read.

Explanation: 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.

User Response: 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.

Explanation: The allocation of the internal buffers used by the asynchronous read log API has failed.

User Response: 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***.**

Explanation: 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.

User Response: No action is required.

SQL2654W   The database log has been asynchronously read to the end of the current active log.

Explanation: All log records in the database active log have been read by the asynchronous log reader.

User Response: No action is required.

SQL2655N   The Asynchronous Log Reader encountered log file *name* **which is not associated with the database being asynchronously read.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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   The Asynchronous Log Reader requires log file *name* **which does not currently reside in the database log directory.**

Explanation: 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.

User Response: Move the specified log file into the database log directory of the database being asynchronously read. If the database log path was changed, the log file might be found in the old log path. Resubmit the call to the Asynchronous Read Log API.

SQL2700 - SQL2799

SQL2701N   Invalid command line option for *progname***. Reason code:** *reason-code*

Explanation: Command line option for the data splitting utility is not valid. Valid options are:

- -c "Configuration file name"
- -d "Distribution file name"
- -i "Input file name"
- -o "Output file prefix"
- -h Usage message

User Response: Given for each reason code:

1   There is an option not started with '-' sign.

2   Each option except 'h' (or 'H') must be followed by an argument.

3   There is an invalid option.

4   The argument of an option is too long (maximum 80 characters).

SQL2702N   **Failed to open the configuration file** *config-file***.**

Explanation: The utility cannot read the configuration file *config-file*.

User Response: Please ensure the configuration file exists and is readable.

SQL2703N   **Failed to open the log file** *log-file***.**

Explanation: The utility cannot open the log file *log-file* for writing or appending.

User Response: Please ensure the log file exists and is writable.

SQL2704N   **Failed to open the input data file** *input-data-file***.**

Explanation: The utility cannot read the input data file *input-data-file*.

User Response: Please ensure the input data file exists and is readable.

SQL2705N   **Failed to open the input partitioning map file** *in-map-file***.**

Explanation: The utility cannot read the input partitioning map file *in-map-file*.

User Response: Please ensure the input partitioning map file exists and is readable.

SQL2706N   **Failed to open the output partitioning map file** *out-map-file***.**

Explanation: The utility cannot open the output partitioning map file *out-map-file* for writing.

User Response: Please ensure the output partitioning map file is writable.

SQL2707N   **Failed to open the distribution file** *dist-file***.**

Explanation: The utility cannot open the distribution file *dist-file* for writing.

User Response: Please ensure the distribution file is writable.

SQL2708N   **Failed to open the output data file** *out-data-file***.**

Explanation: The utility cannot open the output data file *out-data-file* for writing.

User Response: Please ensure the output data file is writable.

SQL2709N   **Syntax error at line** *line* **of the configuration file.**

Explanation: There is a syntax error in the specification of a keyword and its argument.

User Response: A keyword and its argument must be delimited by an '=' sign.

SQL2710N   **Invalid keyword at line** *line* **of the configuration file.**

Explanation: There is an undefined keyword in the configuration file.

User Response: Valid keywords (case insensitive) are:

- DESCRIPTION, CDELIMITER, SDELIMITER, NODES, TRACE, MSG_LEVEL, RUNTYPE, OUTPUTNODES, NODES, OUTPUTNODES, OUTPUTTYPE, PARTITION, MAPFILI, INFILE, MAPFILO, OUTFILE, DISTFILE, LOGFILE, NEWLINE, HEADER, FILETYPE

SQL2711N   **Invalid column delimiter (CDELIMITER) at line** *line* **of the configuration file.**

Explanation: The column delimiter (CDELIMITER) specified in the configuration file is not valid.

User Response: Make sure the column delimiter (CDELIMITER) is a single byte character.

SQL2712N   **Invalid string delimiter (SDELIMITER) at line** *line* **of the configuration file.**

Explanation: The string delimiter (SDELIMITER) specified in the configuration file is not valid.

User Response: The string delimiter (SDELIMITER) cannot be a period.

SQL2713N Invalid run type (RUNTYPE) at line *line* **of the configuration file.**

Explanation: The value for run type (RUNTYPE) specified in the configuration file is not valid.

User Response: Valid run type (RUNTYPE) can be either PARTITION or ANALYZE (case insensitive).

SQL2714N Invalid message level (MSG_LEVEL) at line *line* **of the configuration file.**

Explanation: The value for message level (MSG_LEVEL) specified in the configuration file is not valid.

User Response: Valid message level (MSG_LEVEL) can be either CHECK or NOCHECK (case insensitive).

SQL2715N Invalid check level (CHECK_LEVEL) at line *line* **of the configuration file.**

Explanation: The value for check level (CHECK_LEVEL) specified in the configuration file is not valid.

User Response: Valid check level (CHECK_LEVEL) can be either CHECK or NOCHECK (case insensitive).

SQL2716N Invalid record length (RECLEN) *reclen* **at line** *line* **of the configuration file.**

Explanation: The value for record length (RECLEN) *reclen* specified in the configuration file is not valid.

User Response: The record length (RECLEN) must be between 1 and 32767.

SQL2717N Invalid node specification (NODES) at line *line* **of the configuration file. Reason code** *reason-code***.**

Explanation: The node specification (NODES) in the configuration file is not valid.

User Response: Given for each reason code:

1 NODES has already been defined.

2 The format is not valid. A valid example: NODES=(0,30,2,3,10-15,57)

3 Each entry must be numeric data between 0 and 999.

4 Range specification must be from a lower number to a higher number.

SQL2718N Invalid output node specification (OUTPUTNODES) at line *line* **of the configuration file. Reason code** *reason-code***.**

Explanation: The output node specification (OUTPUTNODES) in the configuration file is not valid.

User Response: Given for each reason code:

1 OUTPUTNODES has already been defined.

2 The format is not valid. A valid example: OUTPUTNODES=(0,30,2,3,10-15,57)

3 Each entry must be numeric data between 0 and 999.

4 Range specification must be from a lower number to a higher number.

SQL2719N Invalid output type (OUTPUTTYPE) at line *line* **of the configuration file.**

Explanation: The output type (OUTPUTTYPE) in the configuration file is not valid.

User Response: Valid output types (OUTPUTTYPE) are W (write) or S (stdin), case insensitive.

SQL2720N Number of partition keys exceeded the maximum "256". This error was detected at line *line* **of the configuration file.**

Explanation: The number of partitioning keys defined cannot exceed the maximum limit: 256.

User Response: Remove one or more partitioning keys defined in the configuration file.

SQL2721N Invalid partitioning key specification (PARTITION) at line *line* **of the configuration file. Reason code** *reason-code*.

Explanation: A partitioning key specification (PARTITION) in the configuration file is not valid. Valid format:

```
PARTITION=<key name>,
          <position>,
          <offset>,
          <len>,
          <nullable>,
          <datatype>
```

If a delimited-data file, <position> must be defined; otherwise, <offset> and <len> must be defined.

User Response: Given for each reason code:

1   Fields must be delimited by the ',' character.

2   <position>, <offset> and <len> must be positive integers.

3   <nullable> must take a value from {N,NN,NNWD}.

4   Valid <data type> includes: SMALLINT, INTEGER, CHARACTER, VARCHAR, FOR_BIT_CHAR, FOR_BIT_VARCHAR, FLOAT (for binary numerics only), DOUBLE (for binary numerics only), DATE, TIME, TIMESTAMP, DECIMAL(x,y).

5   For DECIMAL data type, precision (x) and scale (y) must be specified and they must be positive integers.

6   For CHARACTER or VARCHAR data type, <len> must be specified.

SQL2722N Invalid log file specification (LOGFILE) at line *line* **of the configuration file.**

Explanation: The log file specification (LOGFILE) in the configuration file is not valid.

User Response: The log file specification (LOGFILE) must be in one of the following two formats:

- LOGFILE=<log file name>,<log type>
- LOGFILE=<log file name>

<log type> can only be W (write) or A (append), case insensitive.

SQL2723N Invalid trace specification (TRACE) at line *line* **of the configuration file.**

Explanation: The trace specification (TRACE) in the configuration file is not valid.

User Response: The trace specification (TRACE) must be a positive integer between 0 and 65536 (exclusive).

SQL2724N Invalid node list specification.

Explanation: The node list specification is not valid.

User Response: One and only one of the two parameters: NODES and MAPFILI (input partition map) must be specified in the configuration file.

SQL2725N Filename for output partition map was not specified.

Explanation: If run type is ANALYZE, a filename for the output partition map must be defined.

User Response: Specify a filename for the output partition map.

SQL2726N   There is no partitioning key defined.

Explanation: At least one partitioning key must be defined.

User Response: Specify one or more partitioning keys.

SQL2727N   Partition key *key-name* **exceeds the record length** *reclen***.**

Explanation: For non-delimited data, starting position for a key must be less than the record length.

User Response: Make sure starting position for a key is less than the record length.

SQL2728N   Output node *out-node* **is not defined in the node list.**

Explanation: Output node list must be a subset of the node list, which is derived from NODES or the input partition map file.

User Response: Make sure all output nodes are defined in the node list.

SQL2729N   Invalid input partition map.

Explanation: There is at least one error in the input partition map file.

User Response: The input partition map must contain no less than 4096 data entries, and each data entry must be a number between 0 and 999.

SQL2730N   Error while writing header to output data file *out-data-file***.**

Explanation: An I/O error occurred while writing the header to an output data file.

User Response: Check your operating system documentation for file I/O errors and ensure that there is sufficient space on the output device.

SQL2731N   Error while reading from input data file *filename***.**

Explanation: An I/O error occurred while reading from input data file.

User Response: Check your operating system documentation for file I/O errors.

SQL2732N   Line *line* **of the input data file contains binary data.**

Explanation: Binary data is not permitted on the host versions of this utility program.

User Response: Check your input data file.

SQL2733N   Run type (RUNTYPE) was not defined in the configuration file.

Explanation: The run type (RUNTYPE) must be defined as either PARTITION or ANALYZE.

User Response: Please specify run type (RUNTYPE) in the configuration file.

SQL2734N   Invalid specification for parameter 32KLIMIT at line *line* **of the configuration file.**

Explanation: The specification for parameter 32KLIMIT in the configuration file is not valid.

User Response: Parameter 32KLIMIT can be either YES or NO, case insensitive.

SQL2735W   The record *rec-no* **of the input data file was discarded because it was an empty record.**

Explanation: The record *rec-no* of the input data is discarded because it contains nothing but spaces.

User Response: Check record *rec-no* in the input data file.

SQL2736N **sqlugrpi_api returned an error when processing record at line** *line* **of the input data file.**

Explanation: Partitioning key fields contain invalid data.

User Response: Check the input data file at line *line*.

SQL2737N **Failed to write output data file for output node** *out-node* **when processing record at line** *line* **of input data file.**

Explanation: An I/O error occurred while writing a record to the output data file for node *out-node*.

User Response: Check your operating system documentation for file I/O errors and ensure that there is sufficient space on the output device.

SQL2738W **Record at line** *line* **of input data file has been truncated when writing to the output data file for node** *out-node*. **Expected writing length is** *reclen* **while actual writing length is** *real-len*.

Explanation: Expected writing length (RECLEN) does not match the actual writing length.

User Response: Adjust record length value defined in the configuration file.

SQL2739N **The record length was not specified for a binary numerical data file.**

Explanation: If a binary numerical input data file, the record length has to be defined.

User Response: Please specify the record length in your configuration file.

SQL2740N **Float data type is not allowed in a non-binary input data file.**

Explanation: Float data type is only supported when the file type is BIN (binary).

User Response: Make sure data types and the type of input data file match.

SQL2741N **Invalid file type specification at line** *line* **of the configuration file.**

Explanation: The file type specification (FILETYPE) in the configuration file is not valid.

User Response: Valid values for the file type parameter are as follows:

- ASC (positional ascii data file)
- DEL (delimited ascii data file)
- BIN (ASC file with all numeric data in binary format)
- PACK (ASC file with all decimal data in packed decimal format)
- IMPLIEDDECIMAL (DEL file with decimal data in implied decimal format).

All the values are case insensitive.

SQL2742N **The length of partition key** *partition-key* **does not match its precision.**

Explanation: In a binary input data file, the length of a partition key with decimal data type must conform to the equation: $LENGTH=(PRECISION+2)/2$ (integer division), because it is a packed decimal.

User Response: Match the length of a decimal-typed partition key with its precision if the input data file is a binary data file.

SQL2743N **The length of partition key** *partition-key* **does not match its data type.**

Explanation: In a binary input data file, the length of a partition key with integer, small integer, float, and double data type must be a predefined constant, i.e. 4 for integer, 2 for small integer, 4 for float, and 8 for double.

User Response: Check partition key definitions in the configuration file.

SQL2744N   Illegal file name specification for *file* **at line** *line* **of the configuration file.**

Explanation: The maximum length for a file name is 80 bytes.

User Response: Check the configuration file.

SQL2745N   Invalid NEWLINE flag at line *line* **of the configuration file.**

Explanation: The NEWLINE flag must be either YES or NO, and if not given, it is NO.

User Response: Check the specification of NEWLINE flag in the configuration file.

SQL2746N   An incomplete record was found while reading record *record-number* **from the input data file.**

Explanation: If a fixed-length positional ASC file or a binary numerical data file, each record has to be exactly the same length as the value of the RECLEN parameter in the configuration file.

User Response: Make sure the input data file is complete.

SQL2747N   A record was too long while reading record *rec-no* **from the input data file.**

Explanation: For a positional ASC input data file or a delimited data file with parameter 32KLIMIT on, the maximum record length can not exceed the 32k (bytes) limit.

User Response: Check your input data file and make sure the record length is less than 32k bytes.

SQL2748N   Record *record-number* **has a length of** *length* **bytes, which is too short to hold the partitioning key** *key*.

Explanation: For a positional ASC input data file or a binary numerical data file, each record has to be long enough to hold all partitioning keys.

User Response: Check the record length of your input data file.

SQL2749N   Partitioning key *key-no* **of record** *rec-no* **was not in the first 32k bytes of the record.**

Explanation: If a record is greater than 32k bytes long in a delimited data file, all partitioning keys of each record have to be within the first 32k bytes of the record.

User Response: Check record *rec-no* in the input data file.

SQL2750N   The length of line *line-number* **in the configuration was more than 255 bytes.**

Explanation: The maximum length of a line in the configuration file must be less than 255 bytes.

User Response: Check your configuration file and make sure all lines are less than 255 bytes.

SQL2751N   The actual length *actual-reclen* **of record** *rec-no* **did not match the expected length** *exp-reclen*.

Explanation: If new line checking is required for a fixed length ASC data file (NEWLINE parameter is YES and RECLEN parameter is not zero), the actual length of each record has to match the expected record.

User Response: Check record *rec-no* in the input data file.

SQL2752N   Invalid codepage specification *codepage* **at line** *line* **of the configuration file.**

Explanation: The codepage specification is not valid. It must be a positive integer.

User Response: Correct the codepage specification in the configuration file.

SQL2753N   **Failed to get the territory code
and codepage for the application.
Return code from function**
function-name **is** *rc***.**

Explanation: The program failed to get the
territory code and codepage of its environment.

User Response: Check with your database
system administrator.

SQL2754N   **Codepage** *source-cp* **cannot be
converted to codepage** *target-cp***.**

Explanation: The database does not support
codepage conversion between those two
codepages.

User Response: Please ensure your data is in a
convertible codepage.

SQL2755N   **Decimal data can not be in both
IMPLIEDDECIMAL and
PACKEDDECIMAL format.**

Explanation: Valid format of decimal data is
SQL_PACKEDDECIMAL_FORMAT,
SQL_CHARSTRING_FORMAT, or
SQL_IMPLIEDDECIMAL_FORMAT, mutually
exclusive.

User Response: Verify the format of your
decimal data, correct the format value, and
resubmit the command.

SQL2761N   **Only one of table name or
nodegroup can be specified.**

Explanation: Specify either the table name or
the nodegroup name, but not both.

User Response: Check your command line
options.

SQL2762N   **The utility failed to find the
database installation path.**

Explanation: The utility needs to know where
the database manager is installed, then it can
find its bind file.

User Response: Make sure your database
manager is properly installed.

SQL2763N   **The table** *tbl-name* **was not found.**

Explanation: The table *tbl-name* cannot be
located in sysibm.systables.

User Response: Make sure the table exists in
the database.

SQL2764N   **The nodegroup** *nodegroup* **was not
found.**

Explanation: The nodegroup *nodegroup* cannot
be located in sysibm.sysnodegroupdef.

User Response: Make sure the nodegroup exists
in the database.

SQL2765W   **The utility failed to open output
partition map file** *out-map-file***.**

Explanation: The utility cannot open the output
partition map file for writing. It will write the
output to stdout.

User Response: Check your file access
permissions.

SQL2766N   **The partition map is not of the
correct size** *map-size***.**

Explanation: The size of the partition map is
not correct. Data is corrupted in the database.

User Response: Contact the Database
Administrator to resolve the problem.

SQL2767N   **A command line option is not
valid.**

Explanation: There is an invalid command line
option.

User Response: Make sure to specify the correct
command line options.

SQL2800 - SQL2899

SQL2800N **CREATE, DROP, or CHANGE NODE failed. Reason code** *reason-code*.

Explanation: The utility failed to add, drop, or change the node because the input parameters specified are invalid as indicated by the following reason codes:

- (1) The node number is not specified
- (2) The TCP/IP host name is not specified
- (3) The computer name is not specified
- (4) The node number is invalid.
- (5) The port number is invalid.
- (6) The TCP/IP host name is invalid.
- (7) The port value is not defined for your instance in the services file.
- (8) The port value is not in the valid range defined for your instance in the services file.
- (9) The node number is not unique.
- (10) The host name/port couple is not unique.
- (11) The host name value has no corresponding port 0.

User Response: The action corresponding to the reason code is:

- (1) Ensure that the node number is specified
- (2) Ensure that the TCP/IP host name is specified
- (3) Ensure that the computer name is specified
- (4) Ensure that the node number is between 0 and 999.
- (5) Ensure that the port number is between 0 and 999.
- (6) Ensure that the host name specified is both defined on the system and is operational
- (7) Ensure that the TCP/IP services on your system contains an entry for your instance
- (8) Ensure that you only use port values that are specified in the services file on your system
- (9) Ensure that the node number specified is unique.
- (10) Ensure that the new host name/port is not already defined in the db2nodes.cfg file.

- (11) Ensure that the port value 0 has been defined for host name specified.

SQL2801N **The syntax of the DB2NCRT command is incorrect.**

Explanation: The DB2NCRT utility creates a new node in a partitioned database system.

```
DB2NCRT /n:node
        /u:username,password
        [/i:instance]
        [/h:host]
        [/m:machine]
        [/p:port]
        [/o:instance owning
          machine]
        [/g:netname]
```

The command arguments are:

- /n Specify the node number
- /u to specify the account name and password for the DB2 service.

The command options are:

- /i Specify the instance if different from the default/current instance
- /h Specify the TCP/IP host name if the host name is not the machine default TCP/IP host name
- /m Specify the workstation name if the node is created on a remote machine
- /p Specify the logical port number if this is not the first node on the machine
- /o Specify the computer name of the instance owning machine when creating the first node on a machine.
- /g Specify the network name or IP Address.

User Response: Enter the command again using valid parameters.

SQL2802N **The syntax of the DB2NCHG command is incorrect.**

Explanation: The DB2NCHG utility changes or updates node configuration for a given node in a partitioned database system.

```
DB2NCHG  /n:node
         [/h:host]
         [/m:machine]
         [/p:port]
         [/i:instance]
         [/u:username,
           password]
         [/g:netname]
```

The command options are:

- /h Change the TCP/IP host name
- /m Change the workstation name
- /p Change the logical port number
- /i Specify the instance if different from the default/current instance
- /u Change the Logon Account name and password
- /g Specify the network name or IP Address

User Response: Issue the DB2NCHG command with one of the valid command options.

SQL2803N   The syntax of the DB2NDROP command is incorrect.

Explanation: The DB2NDROP utility drops a node from a partitioned system.

```
DB2NDROP /n:node
         [/i:instance]
```

The command options are:

- /i Specify the instance if different from the default/current instance

User Response: Issue the DB2NDROP command with one of the valid command options.

SQL2804N   The syntax of the DB2NLIST command is incorrect.

Explanation: The DB2NLIST utility list all nodes in a partitioned system.

```
DB2NLIST [/i:instance]
         [/s]
```

The command options are:

- /i Specify the instance if different from the default/current instance

- /s Display node status

User Response: Issue the DB2NLIST command with one of the valid command options.

SQL2805N   A service error has occurred. Reason code *reason-code***.**

Explanation: A service error has occurred during CREATE, DROP or ADD NODE processing as indicated by the following reason codes:

- (1) Unable to register the service
- (2) Unable to set the required User Rights Policy
- (3) Unable to set the Logon Account for the service
- (4) Unable to delete the service

User Response: The action corresponding to the reason code is:

- (1) If the workstation name is specified during DB2NCRT, ensure that the workstation name is correct.
- (2) Ensure that the user name specified is valid.
- (3) Ensure that the user name and the password specified are valid.
- (4) If the node resides on another machine, ensure that the machine is running.

If the problem persists, contact IBM Service.

SQL2806N   Node *node* **not found for instance** *instance***.**

Explanation: DB2NDROP failed because the node does not exist.

User Response: Ensure the node number is correct and reissue the command.

SQL2807N   Node *node* **already exists for instance** *instance***.**

Explanation: DB2NCRT failed because the node already exists.

User Response: Ensure the node number is

correct and reissue the command.

SQL2808W **Node** *node* **for instance** *instance* **has been deleted.**

Explanation: The DB2NDROP processing has completed successfully.

User Response: No further action is required.

SQL2809W **Node:** *node* **added to instance:** *instance* {**Host:** *host-name* **Machine:** *machine-name* **Port:** *port-num*}

Explanation: The DB2NCRT processing has completed successfully.

User Response: No further action is required.

SQL2810W **Node:** *node* **changed in instance:** *instance* {**Host:** *host-name* **Machine:** *machine-name* **Port:** *port-num*}

Explanation: The DB2NCHG processing has completed successfully.

User Response: No further action is required.

SQL2811N **The command is invalid because the instance is not a partitioned database instance.**

Explanation: The command is only valid if the instance is a partitioned database instance.

User Response: Ensure that the specified instance name is valid. If the instance name is not specified on the command line, ensure that the DB2INSTANCE environment is set to a valid partitioned database instance.

SQL2812N **An invalid argument was entered for the db2drvmp command.**

Explanation: Usage:

```
db2drvmp add    node_number
                from_drive to_drive
        drop    node_number
                from_drive
        query   [node_number]
```

```
                [from_drive]
        reconcile [node_number]
                [from_drive]
```

Valid arguments for this command are:

add assign a new database drive map.

drop remove an existing database drive map.

query query a database map.

reconcile
repair a database map drive when the registry contents are damaged.

node_number
the node number. The parameter is required for add and drop operations.

from_drive
the drive letter to map from. This parameter is required for add and drop operations.

to_drive
to drive letter to map to. This parameter is required only for add operation. It is not applicable for any other operations.

User Response: Enter the command again using valid arguments.

SQL2813I **The drive mapping from drive** *drive-1* **to drive** *drive-2* **was added for node** *node*.

Explanation: The drive mapping was added successfully.

User Response: No action necessary.

SQL2814I **The drive mapping from drive** *drive* **was deleted for node** *node*.

Explanation: The drive mapping was deleted successfully.

User Response: No action necessary.

SQL2815I   **The drive mapping for node** *node*
                is: *drive-1* **-** *drive-2***.**

Explanation: Informational message.

User Response: No action necessary

SQL3000 - SQL3099

SQL3001C   **An I/O error (reason =** *reason***)**
                occurred while opening the
                output file.

Explanation: A system I/O error occurred while
opening the output file.

The command cannot be processed. No data is
processed.

User Response: For IMPORT/LOAD, verify
that the output file exists. For EXPORT, 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.

Explanation: 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.

User Response: 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.

Explanation: A system I/O error occurred while
closing the output data file.

The file is not closed.

User Response: If the output data file is
incomplete, erase it and resubmit the command.

SQL3004N   **The filetype parameter is not**
                valid.

Explanation: 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.

IXF files can not be used to load into a table
defined on a multinode nodegroup.

The command cannot be processed.

User Response: Resubmit the command with a
valid *filetype* parameter.

SQL3005N   **Processing was interrupted.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: A system I/O error occurred while writing to the message file.

Processing may not have finished.

User Response: If the message file is incomplete, erase it and resubmit the command.

SQL3008N   The utility encountered an error error while connecting to the database.

Explanation: The IMPORT or EXPORT utility could not connect to the database.

No data is imported or exported.

User Response: Look at the error number in the message for more information. Make changes and resubmit the command.

SQL3009N   The Action String parameter is not valid.

Explanation: The *Action String* (e.g. "REPLACE into ...") parameter in the command is not valid. The *Action String* pointer may be incorrect. The structure that the *Action String* points to may be incorrect. The *Action String* structure may contain characters that are not valid.

The command cannot be processed.

User Response: Verify the *Action String* pointer and the structure it points to. Resubmit the command with a valid *Action String*.

SQL3010N   The METHOD parameter is not valid.

Explanation: The *METHOD* parameter in the command is not valid. The *METHOD* pointer may be incorrect. The structure that the *METHOD* points to may be incorrect. The *METHOD* structure may contain characters that are not valid.

The command cannot be processed.

User Response: Examine the *METHOD* pointer and the structure it points to. Resubmit this command with a valid *METHOD*.

SQL3011C   There is not enough storage to process the command.

Explanation: A memory allocation error occurred. Not enough memory is available to process the command, or there was an error freeing storage.

The command cannot be processed.

User Response: 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.
- The *util_heap_sz* is shared by the LOAD, BACKUP and RESTORE utilities, so have fewer concurrent instances of these utilities running.

SQL3012C   A system error occurred.

Explanation: An operating system error occurred.

The command cannot be processed.

User Response: 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.

Explanation: 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.

User Response: Examine the *filetmod* pointer and the structure it points to. Resubmit the

command with a valid *filetmod*.

SQL3014C   An I/O error occurred while closing the message file.

Explanation: A system I/O error occurred while closing the message file.

The message file is not closed.

User Response: If the message file is incomplete, resubmit the command.

SQL3015N   An SQL error *sqlcode* occurred during processing.

Explanation: An SQL error occurred during the utility call.

The utility stops processing.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: For Delimited ASCII (DEL), the period cannot be specified as the character string delimiter.

The command cannot be processed.

User Response: Resubmit the command with valid delimiter overrides.

SQL3019N   No Action String parameter was specified in the command.

Explanation: No *Action String* (e.g. "REPLACE into ...") parameter is specified for this utility call. This parameter is required.

The command cannot be processed.

User Response: Resubmit the command with an *Action String* parameter.

SQL3020N   The user does not have the authority to run the specified Export command.

Explanation: 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.

User Response: 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***.**

Explanation: The user attempted to import data without the appropriate authorization for the specified option and table.

Import using the INSERT option requires one of the following:

- SYSADM or DBADM authority
- CONTROL privilege on the table, view, or entire hierarchy
- INSERT and SELECT privilege on the table, view, or entire hierarchy.

Note: entire hierarchy refers to all subtables or object views in the hierarchy.

Import to an existing table or view using the INSERT_UPDATE, REPLACE, or the REPLACE_CREATE option, requires one of the following:

- SYSADM or DBADM authority
- CONTROL privilege on the table, view, or entire hierarchy.

Note: entire hierarchy refers to all subtables or object views in the hierarchy.

Import to a table that does not exist, using the CREATE or REPLACE_CREATE option, requires one of the following:

- SYSADM or DBADM authority
- CREATETAB authority on the database and one of:
  - IMPLICIT_SCHEMA authority on the database, if the schema name of the table does not exist
  - CREATEIN privilege on the schema, if the schema of the table exists

The import operation is not performed.

User Response: Ensure that the user has the authorization to perform the import operation.

SQL3022N **An SQL error** *sqlcode* **occurred while processing the SELECT string in the Action String parameter.**

Explanation: In IMPORT or EXPORT, during processing of the SELECT string from the *Action String* (e.g. "REPLACE into ...") structure, an SQL error occurred.

The command cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the statement with a valid database name parameter.

SQL3024N **An I/O error occurred while opening the format file.**

Explanation: A system I/O error occurred while opening the format file. This error can refer to a problem on either the client or the server.

A possible cause may be that the format file is in use by another application.

The command cannot be processed.

User Response: Ensure that the format file is valid, including the path, and that it is not currently in use by another application.

Resubmit the command. Look in the diagnostics log file for additional information.

SQL3025N   A parameter specifying a filename or path is not valid.

Explanation: The parameter contains a path, directory, or file name that is not valid.

For IMPORT and EXPORT, verify the *datafile* parameter.

For LOAD, verify that 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 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.

User Response: Resubmit the statement with a valid parameter.

SQL3026N   The msgfile or tempfiles path parameter is not valid.

Explanation: For IMPORT or EXPORT, the *msgfile* parameter contains a path, directory, or file name that is not valid.

For LOAD, either the *msgfile* parameter contains a path, directory, or file name that is not valid on the client, or the tempfiles path 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.

User Response: Resubmit the statement with a valid *msgfile* and/or tempfiles path parameter.

SQL3027N   An I/O error occurred while reading from the format file.

Explanation: A system I/O error occurred while reading from the format file. This error can refer to a problem on either the client or the server.

The command cannot be processed.

User Response: Verify that the format file is readable.

SQL3028N   The export method indicator is not valid. It must be either 'N' or 'D'.

Explanation: The export method indicator must be either N for Names or D for Default.

The command cannot be processed.

User Response: Resubmit the statement with a valid method indicator.

SQL3029N   No filetype parameter was specified.

Explanation: 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.

User Response: Resubmit the statement with a valid *filetype*.

SQL3030C   An I/O error (reason = *reason*) occurred while opening the input file.

Explanation: 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.

A possible cause may be that the input file is in use by another application.

The command cannot be processed.

User Response: Ensure that the input file is valid, including the path, and that it is not currently in use by another application.

Resubmit the command.

SQL3031C   An I/O error occurred while reading from the input file.

Explanation: 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.

User Response: Verify that the input file is readable.

SQL3032N   The LOAD/IMPORT method indicator is not valid for the specified filetype. It must be either 'N', 'P', or 'default'.

Explanation: For the IXF and WSF filetypes, the LOAD/IMPORT method indicator must be either N for Names, P for Positional, or D for Default.

The command cannot be processed.

User Response: Resubmit the command with a valid method indicator.

SQL3033N   A keyword such as INSERT, REPLACE, CREATE, INSERT_UPDATE, or REPLACE_CREATE is missing from the target specification or it is misspelled.

Explanation: For IMPORT, the *Action String* (e.g. "REPLACE into ...") parameter does not contain the keyword INSERT, REPLACE, CREATE, INSERT_UPDATE, or REPLACE_CREATE. For LOAD, the *Action String* 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.

User Response: Resubmit the command with a valid *Action String* parameter.

SQL3034N   The keyword INTO is missing from the target specification or it is misspelled.

Explanation: The INTO keyword is missing or spelled incorrectly. INTO must be followed by at least one blank.

The command cannot be processed.

User Response: Resubmit the command with a valid *Action String* (e.g. "REPLACE into ...") parameter.

SQL3035N   The tablename parameter in the target specification is not valid.

Explanation: For IMPORT, the *tablename* in the *Action String* (e.g. "REPLACE into ...") is not valid. For LOAD, the *tablename* or the *exception tablename* in the *Action String* is not valid.

The command cannot be processed.

User Response: Resubmit the command with a valid *tablename*. Also ensure that all command keywords and parameters are entered in the correct order.

SQL3036N   The tcolumn-list in the target specification does not have a closed parenthesis.

Explanation: The *tcolumn-list* must be delimited by parentheses. The list was not terminated by the closed parenthesis.

The command cannot be processed.

User Response: Resubmit the statement with a valid, complete column list in the *Action String* (e.g. "REPLACE into ...") parameter.

SQL3037N   An SQL error *sqlcode* occurred during Import processing.

Explanation: An SQL error occurred during processing of the *Action String* (e.g. "REPLACE into ...") parameter.

The command cannot be processed.

User Response: Look at the SQLCODE (message number) in the message for more

information. Make changes and resubmit the command.

SQL3038N   The Action String parameter contains unexpected characters.

Explanation: For IMPORT, there are characters other than blanks after the closing parenthesis of the column list in the *Action String* (e.g. "REPLACE into ...") parameter. For LOAD, there are characters other than blanks after the closing parenthesis of the column list and/or the exception tablename in the *Action String* parameter.

The command cannot be processed.

User Response: Resubmit the command with a valid *Action String* parameter.

SQL3039W   The memory available to LOAD for DATA BUFFER prohibits full LOAD parallelism. Load parallelism of *value* will be used.

Explanation: The LOAD utility attempted to determine the optimal level of CPU parallelism for SMP exploitation based on the system configuration, or the user specified a value for LOAD parallelism when invoking the utility. However, one of the following limitation was encountered:

1. The amount of free memory in the utility heap prohibits this degree of parallelism.
2. The DATA BUFFER parameter was specified with a value too small to allow for the specified or available parallelism.

A lesser degree of parallelism, requiring less memory, will be used.

User Response:
1. Ignore this message, and the LOAD will complete normally using the smaller value for LOAD parallelism. However, Load performance may be less than optimal.
2. Specify a smaller value for LOAD parallelism when invoking the utility.
3. Increase the size of the utility heap.

4. Increase the size of the data buffer parameter, or leave the parameter blank and let the LOAD utility determine a default based on the freespace in the utility heap.

SQL3040N   Unable to use the lobpath/lobfile parameters as specified. Reason code: *reason-code*.

Explanation: 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.

User Response: 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.

SQL3041N   The date value specified is not supported in Sybase.

Explanation: Sybase does not support date values prior to 1753.

User Response: Please give a valid date value

---

and issue the command again.

SQL3042N   The LINKTYPE specified for a DATALINK column is invalid.

Explanation: The value specified for LINKTYPE for a DATALINK column is invalid.

User Response: Check the LINKTYPE specified. Correct the value and resubmit the command.

SQL3043N   The DATALINK SPECIFICATION for DATALINK column(s) is invalid.

Explanation: The DATALINK SPECIFICATION for DATALINK column(s) is invalid for one of the following reasons:

- No value is specified for DL_URL_REPLACE_PREFIX.
- No value is specified for DL_URL_DEFAULT_PREFIX.
- No value is specified for DL_URL_SUFFIX.
- Contains keywords other than DL_URL_REPLACE_PREFIX or DL_URL_DEFAULT_PREFIX or DL_URL_SUFFIX.

User Response: Correct the specification and resubmit the command.

SQL3044N   The DATALINK SPECIFICATION for a DATALINK column has duplicate prefix specification.

Explanation: The DATALINK SPECIFICATION for a DATALINK column has duplicate specifications for DL_URL_REPLACE_PREFIX or DL_URL_DEFAULT_PREFIX.

User Response: Remove the duplicate specification and resubmit the command.

SQL3045N   A dcolumn position in the METHOD parameter is less than one or greater than the maximum number of columns (1024) in a Delimited ASCII file.

Explanation: 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.

User Response: Resubmit the command with valid *dcolumn* positions.

SQL3046N   The number of columns in the METHOD parameter is less than one.

Explanation: For *METHOD* methods other than Default, the number of columns specified must be a positive number (greater than 0).

The command cannot be processed.

User Response: Resubmit the command with a valid number of columns in the *METHOD* parameter.

SQL3047N   The LOAD/IMPORT method specified in METHOD is not valid for Delimited ASCII files. It must be either 'P' or 'default'.

Explanation: The only valid LOAD/IMPORT methods for Delimited ASCII files are P for Positions or D for Default.

The command cannot be processed.

User Response: 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.

Explanation: When fewer columns are specified in the *METHOD* 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: This is a warning only.

SQL3053N **There are more than 8191 rows to be exported to a worksheet formatted file.**

Explanation: 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.

User Response: 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.**

Explanation: 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 LOAD/IMPORT utility stops processing. No data is imported.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Verify that the input file is correct.

SQL3058N   The identifier field in the H record is not IXF.

Explanation: 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.

User Response: Examine the identifier field in the H record.

SQL3059N   The version field in the H record is not valid.

Explanation: The version field in the H record contains a value that is not valid.

The utility stops processing. No data is loaded.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Examine the single-byte and double-byte code page field in the H record, change them to appropriate values as specified in the *Application Development Guide* and resubmit the command.

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.

Explanation: 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.

User Response: Examine the double-byte code page field in the H record, change it to an appropriate value as specified in the *Application Development Guide* 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The system encountered an error while determining the code page of the application.

The utility stops processing. No data is loaded or unloaded.

User Response: Contact your technical service representative.

SQL3066N **While reading or looking for the T record, the end of the file was reached.**

Explanation: 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.

User Response: 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.**

Explanation: The length field in the T record is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

User Response: Examine the length field in the T record.

SQL3068N **The value in the length field of the T record is too small.**

Explanation: 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.

User Response: Examine the length field in the T record.

SQL3069N **The first non-A record following the H record is not a T record.**

Explanation: 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.

User Response: Examine the records following the H record.

SQL3070N **The length field in an A record is out of range.**

Explanation: 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.

User Response: Examine the length field in the A records.

SQL3071N **The data-convention field in the T record is not C.**

Explanation: The data-convention field in the T record is a value other than C.

The utility stops processing. No data is loaded.

User Response: Examine the data-convention field in the T record.

SQL3072N **The data-format field in the T record is not M.**

Explanation: The data-format field in the T record is a value other than M.

The utility stops processing. No data is loaded.

User Response: Examine the data-format field in the T record.

SQL3073N **The machine format field in the T record is not PCbbb (where b = blank).**

Explanation: The machine format 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.

User Response: Examine the machine format field in the T record.

SQL3074N **The data-location field in the T record is not I.**

Explanation: The data-location field in the T record is a value other than I.

The utility stops processing. No data is loaded.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: Examine the CCNT field in the T record.

SQL3078N **The length field in an A record cannot be converted to a numeric value.**

Explanation: The length field in an A record is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

User Response: Examine the length fields in the A records.

SQL3079N   The length field in a C record cannot be converted to a numeric value.

Explanation: The length field in a C record is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

User Response: Examine the length field in the C records.

SQL3080N   The value in the length field of the C record is too small.

Explanation: 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.

User Response: Examine the length field in the C records.

SQL3081N   Not enough C records were found.

Explanation: 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.

User Response: Examine the T and C records.

SQL3082N   While reading or looking for a C record, the end of the file was reached.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Check the *METHOD* parameter for erroneous names or positions or for fewer entries in the *METHOD* parameter than the *Action String* (e.g. "REPLACE into ...") 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: Some record other than a D record was found where a D record was expected.

The utility stops processing. No data is loaded.

User Response: Examine the D records in the file.

SQL3090N The length field in a D record cannot be converted to a numeric value.

Explanation: The length field in a D record is not an ASCII representation of a number.

The utility stops processing. No data is loaded.

User Response: Examine the length field in the D records.

SQL3091N The length field in a D record is out of range.

Explanation: 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.

User Response: Examine the length field in the D records.

SQL3092N The identifier field in a D record does not contain the expected value.

Explanation: 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.

User Response: Examine the identifier field in the D records.

SQL3093N **The input file is not a valid WSF file.**

Explanation: 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.

User Response: Verify that the file is a valid WSF file and that the name was entered correctly.

SQL3094N **The input column *name* was not found, but the corresponding database column is not nullable.**

Explanation: 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

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Regenerate the WSF file with a supported level of the Lotus product.

SQL3100 - SQL3199

SQL3100W   Column number *column-number* **(identified as** *name***) in the output DEL format file is longer than 254 bytes.**

Explanation: 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.

User Response: 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***.**

Explanation: 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.

User Response: 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 METHOD parameter is greater than the number of columns in the Action String (e.g. ″REPLACE into ...″) parameter.

Explanation: 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.

User Response: Review the data in the output table or file.

SQL3103W   The number of columns in the METHOD parameter is less than the number of columns in the Action String (e.g. ″REPLACE into ...″) parameter.

Explanation: 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 input table or file will be processed. Data in the excess output columns is not processed.

User Response: Review the data in the output table or file.

SQL3104N   The Export utility is beginning to export data to file *name***.**

Explanation: This is the normal beginning message.

User Response: No action is required.

SQL3105N The Export utility has finished exporting *number* **rows.**

Explanation: This is the summary message for the export utility 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.

User Response: 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 and resubmit the command if desired. If a negative sqlcode was returned, an error occurred during the export 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.

Explanation: The error message may be incomplete or formatted incorrectly.

User Response: No action is required.

SQL3107W There is at least one warning message in the message file.

Explanation: At least one warning was received during processing.

This warning does not affect processing.

User Response: Review the message file warning.

SQL3108W Unable to access file referenced by a DATALINK value in row *row-number* **and column** *column-number*. **Reason code =** *reason-code*.

Explanation: The possible causes of this message depend on the value of *reason-code*:

1 DATALINK value data location format is not valid.

2 DATALINK value DB2 Data Links Manager is not registered with the database.

3 DATALINK linktype value is not valid.

4 DATALINK value referenced file cannot be found by the DB2 Data Links Manager.

5 DATALINK value referenced file is already linked to the database.

6 DATALINK value referenced file cannot be accessed for linking. It may be a symbolic link or a file with permission bit for set user ID (SUID) on or set group ID (SGID) on.

7 DATALINK value URL or comment is too long.

8 DATALINK value referenced file cannot be linked to the database. The DB2 Data Links Manager does not authorize the DB2 user to link this file.

User Response: The action is based on the *reason-code* as follows:

1 Correct the data location format. If hostname is not specified, DB2 can use the local hostname as default only if support for FILE LINK CONTROL is enabled. Refer to the Administration Guide for information on enabling this support.

2 Ensure that the correct DB2 Data Links Manager is specified and if so have it registered to the database. Registered DB2 Data Links Managers are ignored unless support for FILE LINK CONTROL is enabled. Refer to the Administration Guide for information on enabling this support.

3 Correct the linktype value.

4 Check that the correct file is specified and that the file exists

5 Either unlink the existing reference to the file or do not specify the file in this statement.

6   Linking of directories is not allowed. Use the actual file name, not the symbolic link. If SUID or SGID is on, this file cannot be linked using a DATALINK type.

7   Reduce the length of the data location value or the comment.

8   Contact the DB2 Data Links Manager administrator to get the necessary authorization

---

SQL3109N   The utility is beginning to load data from file *name***.**

Explanation: 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.

User Response: No action is required.

SQL3110N   The utility has completed processing. *number* **rows were read from the input file.**

Explanation: This is the normal ending message.

User Response: No action is required.

SQL3111C   An I/O error occurred while closing the input data file.

Explanation: 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.

User Response: Examine the input file for an I/O error.

SQL3112W   There are fewer input file columns specified than database columns.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The field value is longer than 32700 bytes.

The value was truncated after 32700 bytes.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The field data in the input file was shorter than the database column.

The data on the right is padded with blanks.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: This is a warning only. No action is required.

SQL3133W **The field in row** *row-number* **and column** *column-number* **contains invalid DATALINK value. A null was loaded.**

Explanation: The DATALINK value in the specified field is invalid. 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.

User Response: Examine the input value. If necessary, correct the input file and resubmit the command or edit the data in the table.

SQL3134W **The field in row** *row-number* **and column** *column-number* **contains invalid DATALINK value, but the target column is not nullable. The row was not loaded.**

Explanation: The DATALINK value in the specified field is invalid. 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.

User Response: Examine the input value. If necessary, correct the input file and resubmit the command.

SQL3135N **The number of columns in the METHOD parameter is greater than the number of columns in the target table.**

Explanation: The number of data columns in the METHOD parameter must be less than or equal to the number of data columns in the actual table.

User Response: Specify the correct number of input columns in the METHOD parameter and resubmit the command.

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.**

Explanation: 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.

User Response: 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 **The ending character string delimiter was not found before the end of the input data file was reached.**

Explanation: 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.

User Response: 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.**

Explanation: The IMPORT or EXPORT utility could not disconnect from the database.

The output data may be incomplete.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The number of rows exported exceeds 2048. The first generation products cannot support more than 2048 rows.

Continue processing.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: Information is given about the product that created the PC/IXF file and when it was created.

User Response: 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.**

Explanation: No data conversion will be performed from the IXF code page to the application code page because the FORCEIN option was specified.

User Response: 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 inserted because the FORCEIN option was specified.**

Explanation: The double-byte code page values in the record and the application are not compatible. Because the *FORCEIN* option was used, the data is inserted.

User Response: No action is required.

SQL3153N **The T record in the PC/IXF file has name** *name*, **qualifier** *qualifier*, **and source** *source*.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The null field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The column length field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

User Response: 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.**

Explanation: The precision field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

User Response: 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.**

Explanation: The scale field in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

User Response: 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.**

Explanation: The column length field in the C record for the indicated column is blank.

A column length of 00008 is used.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The column type in the C record for the indicated column is not valid.

Data from the indicated column is not loaded.

User Response: 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.**

Explanation: 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.

User Response: No action is required. If nulls are not acceptable for this column, check the *METHOD* parameter for erroneous names or positions or for fewer entries in the *METHOD* parameter than columns stated in or implied by the *Action String* (e.g. "REPLACE into ...") parameter.

SQL3167W The PC/IXF column specified to be loaded into database column *name* **is not valid. Nulls will be inserted.**

Explanation: 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.

User Response: 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.**

Explanation: The source PC/IXF and the target database column types or lengths may not be compatible.

Null values are loaded for the indicated column.

User Response: 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*.

Explanation: This is for information only about the optional use of the *FORCEIN* option.

User Response: No action required.

SQL3170W The end of the file was reached within a row of data. The partial row of data was not loaded.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.**

Explanation: The specified input column was not found in the input spreadsheet file. The database column is nullable and contains null values.

User Response: Verify the specified input column name.

SQL3173N The inserted data for column *name* **will always contain fewer characters than the column width.**

Explanation: The database column width is larger than the maximum worksheet format (WSF) label record.

Continue processing.

User Response: 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.**

Explanation: The database column type is not valid for a worksheet format (WSF) file. Because the column is nullable, nulls are imported for the column.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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 inserting into a non-nullable column in the database. The row was not inserted.**

Explanation: 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 inserted.

Processing continues with the next row. The row is not inserted.

User Response: Either edit the data in the table or verify that the data in the spreadsheet file is valid for inserting into a database manager database.

SQL3180W   Insert diskette *number* **into drive** *drive***.**

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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   Insert diskette *number* **into drive** *drive***. The diskette that is currently inserted is not the right diskette or the continuation diskette is not valid.**

Explanation: 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.

User Response: 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.

Explanation: 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 LOAD/IMPORT or export of delimited ASCII (DEL) files.

The utility stops processing. The erroneous delimiter overrides are ignored.

User Response: Resubmit the command with a correct *filetmod* parameter.

SQL3185W   The previous error occurred while processing data from row *row-number* **of the input file.**

Explanation: This message provides the identification of the row where the error occurred for the previous message listed in the message file (for example, SQL3306).

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

Federated system users: in addition to the previously listed cause, this error can occur when a CREATE NICKNAME statement was issued at federated server 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 federated server catalog. The *sqlcode* in the message provides further information about the problem encountered.

User Response: 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.

Federated system users: select from the data source catalog and the federated server catalog to determine which index or indexes were not created in the federated server 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 federated server 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.

Explanation: When running LOAD/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.

User Response: Resubmit the command.

SQL3189N   The previous message refers to index *name* **with columns** *column-list***.**

Explanation: 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.

User Response: Manually create the index with the CREATE INDEX command.

SQL3190N   The indexixf option is not valid for this Import operation.

Explanation: 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 *METHOD* 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.

User Response: Resubmit the command either without the INDEXIXF option or with the other parameters that are valid with the INDEXIXF option.

SQL3191N **The field in row** *row-number*, **column** *column-number* **which begins with** *string* **does not match the user specified DATEFORMAT, TIMEFORMAT, or TIMESTAMPFORMAT. The row will be rejected.**

Explanation: The data does not match the user specified format. This may be caused by missing fields, mismatched column separators, or a value out of range.

User Response: Examine the input value. Correct the input file or specify a DATEFORMAT, TIMEFORMAT, or TIMESTAMPFORMAT that matches the data and resubmit the command.

SQL3192N **In the filetmod a user specified format** *keyword* **beginning with the string** *string* **is not valid.**

Explanation: The user specified format is not valid because it may be specified more than once, or it may contain an invalid character.

The formats must be enclosed in double-quotes.

Valid DATEFORMAT specifiers include "YYYY" and the "M", and "D" characters.

Valid TIMEFORMAT specifiers include "AM", "PM", "TT", and the "H", "M", and "D" characters.

Valid TIMESTAMPFORMAT specifiers include all of the specifiers for DATEFORMAT and TIMEFORMAT, as well as "UUUUUU". However, "M" cannot be next to both a date format specifier and a time format specifier.

A field separator is necessary if the corresponding value in the data file can have a variable length.

The utility stops processing.

User Response: Examine the format specifier. Correct the format, and resubmit the command.

SQL3193N **The specified view or materialized query table cannot be updated. You cannot LOAD/IMPORT into this view or LOAD into this materialized query table.**

Explanation: The LOAD/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 LOAD utility can be run against a materialized query table only if the materialized query table is not replicated. The specified table is a replicated materialized query table.

The IMPORT utility can be run against a materialized query table only if the materialized query table is a user maintained materialized query table. The specified table is a system maintained materialized query table.

The LOAD/IMPORT utility stops processing. No data is inserted.

User Response: 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.**

Explanation: The utility cannot be run against a system table.

The utility stops processing. No data is loaded.

User Response: 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.**

Explanation: 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.

User Response: Prompt the user for the diskette and then return to the utility with the *callerac* parameter indicating whether processing continues or ends.

SQL3196N **The input file was not found.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the second operation when no other processes are attempting to execute the same utility.

SQL3200 - SQL3299

SQL3201N **The specified table cannot be replaced because another table is dependent on it.**

Explanation: A table that is a parent in a relationship with another table cannot be replaced.

The command cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Select a different option, such
as INSERT, or choose a different target for the
utility operation.

**SQL3207N   Invalid table-list provided.
             Reason code** *reason-code***.**

Explanation: The traversal-order-list/subtable-
list provided is invalid. The explanation for the
reason code is:

1. Tables specified in the traversal-order-list are
    not in PRE-ORDER fashion.

2. Tables specified in the traversal-order-list are
    not connected.

3. Mismatch of schema name found in
    traversal-order-list/subtable-list.

4. With REPLACE option, some subtables are
    missing in the traversal-order-list.

5. Subtable-list is not equal to or a subset of the
    traversal-order-list.

User Response: The user action, based on the
reason code, is:

1. Make sure the traversal-order-list is in
    PRE-ORDER fashion.

2. Make sure all tables in the traversal-order-list
    are connected.

3. Make sure schema names are consistent.

4. When REPLACE option is used, make sure
    all subtables in the hierarchy are included.

5. Make sure subtable-list is equal to or is a
    subset of the traversal-order-list.

**SQL3208W   Importing data from typed tables
             into regular table.**

Explanation: User specified to import data from
typed tables into regular table. Please note,
object_id column will not be cast while
importing.

User Response: Please make sure such
operation is intentional.

**SQL3209N   Import with CREATE option does
             not allow renaming subtable
             name nor attribute names.**

Explanation: When CREATE option is used,
neither subtable name nor attribute names can be
renamed.

User Response: Please check your IMPORT
command and make sure no subtable-list is
specified.

SQL3210N   Option *option* **is not compatible
             with hierarchy in** *command-name***.**

Explanation: *option* is not compatible with
hierarchy in EXPORT, IMPORT, or LOAD.

User Response: Please check the command
syntax for hierarchical support.

**SQL3211N   LOAD does not support typed
             tables.**

Explanation: LOAD does not support typed
tables. Please consider IMPORT instead.

User Response: Please use IMPORT to bring
hierarchical data into the database.

SQL3212N The TERMINATE option of the LOAD command is not currently supported for tables with DATALINK column(s), or table spaces in delete pending state.

Explanation: An attempt was made to terminate a crashed, interrupted, or forced LOAD operation against a table which contains DATALINK column(s), or a table which resides in table space(s) in delete pending state. These are not currently supported.

User Response: Use the RESTART option of the LOAD command to recover a crashed, interrupted, or forced LOAD operation.

SQL3213I The indexing mode is *mode***.**

Explanation: The indexing mode values are a follows:

REBUILD
indexes will be completely rebuilt

INCREMENTAL
indexes will be extended

DEFERRED
indexes will not be updated, but will be marked as requiring refresh prior to next access.

User Response: No action is required.

SQL3214N The LOAD utility does not support deferred indexing for tables with unique indexes.

Explanation: An indexing mode of DEFERRED was specified for a table that has unique indexes. This is not valid.

User Response: Specify an indexing mode of AUTOSELECT, REBUILD or INCREMENTAL, and reissue the command.

SQL3215W The Load utility currently does not support INCREMENTAL indexing when loading into a DMS table space on tables where the table's index object resides in the same table space as any of the other target table's objects, and the COPY option is also specified. The REBUILD indexing mode will be used instead.

Explanation: INCREMENTAL indexing mode is not supported for this operation. REBUILD indexing mode is used instead.

User Response: The user can avoid this warning by defining the index in a table space that is not shared by other objects in the table to be loaded. Alternatively, avoid use of the COPY option. Please refer to the DB2 documentation for a full list of alternatives to the COPY option.

SQL3216W The table's index object was not consistent at the time the load utility began. INCREMENTAL indexing cannot be performed during this load utility operation. The REBUILD indexing mode will be used instead.

Explanation: INCREMENTAL indexing can only be used on tables that have a consistent index object at the time the load utility begins. Loading with indexing mode REBUILD causes the table index to be rebuilt in a consistent manner.

User Response: No action required.

SQL3217W The INCREMENTAL indexing mode is supported only when using LOAD to append data using the INSERT INTO action. The current LOAD action is *action***. The utility will use indexing mode of** *mode* **instead.**

Explanation: INCREMENTAL indexing can only be used when appending data to a table using the load INSERT action. This feature is not supported with when loading with REPLACE, RESTART, or TERMINATE actions.

User Response: No action necessary.

SQL3218C **The LOAD utility cannot proceed because it has encountered one or more damaged index files. Restart the database and resubmit the LOAD command.**

Explanation: The LOAD utility cannot proceed because one or more index on disk data structures for the target table are in an inconsistent state.

User Response: Terminate all applications and issue the RESTART DATABASE command for the affected database to rebuild the damaged index or indexes. Then resubmit the LOAD command.

SQL3219N **LOAD utility failed to disable constraint checking for the target table.**

Explanation: The LOAD utility encountered problems during its attempt to disable constraint checking for the target table.

User Response:

- Issue the SET INTEGRITY OFF command on the target table prior to running the LOAD utility.
- If attempting a LOAD in the REPLACE mode after a previously failed LOAD operation, use the LOAD TERMINATE command to put the tablespace into an accessible state prior to issuing a LOAD REPLACE command.

SQL3220W **Volume** *volume-name* **was not found in the** *directory-name* **directory. Copy the volume into this directory and continue the LOAD/IMPORT.**

Explanation: An attempt to LOAD/IMPORT a multiple IXF file was made, but one of the files is missing from the directory specified. LOAD/IMPORT tries to find the parts in the same directory as the first part.

The import will terminate.

User Response:

- Find the part and put it in the same directory as the first part. Then call LOAD/IMPORT again with *callerac* of SQLU_CONTINUE. The LOAD/IMPORT will continue processing the file.
- Terminate the LOAD/IMPORT by calling LOAD/IMPORT with a callerac of SQLU_TERMINATE.

SQL3221W **...Begin COMMIT WORK. Input Record Count =** *count*

Explanation: The import is about to attempt a COMMIT of the work performed.

User Response: 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.**

Explanation: The COMMIT was successful.

User Response: None necessary if you get this message.

SQL3223N **The parameter of type pointer to** *parameter* **is not specified properly.**

Explanation: The parameter of type <parameter> is not specified properly. The types are one of "struct sqluimpt_in", "struct sqluimpt_out", "struct sqluext_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 sqluext_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.

User Response: 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.

Explanation: 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.

User Response: Ensure that the RESTARTCOUNT value is correct or call the utility again with the REPLACE or INSERT option and a correct RESTARTCOUNT value.

SQL3227W   Record token *token1* refers to user record number *token2*.

Explanation: An error or warning was encountered during LOAD, IMPORT or EXPORT of a table. CPU parallelism was greater than 1 at the time the problem was encountered, and an SQL message was written which identified the user record with a special unique token. This message serves to map the unique record token to the record number of the source user data.

User Response: Refer to the original SQL message returned for appropriate action.

SQL3228N   DEFERRED INDEXING is not supported for tables with DATALINK columns.

Explanation: The "indexing mode" option for the load utility was specified as "deferred". This option is not supported with tables having DATALINK columns.

User Response: Reissue the load command specifying a different indexing mode.

SQL3250N   COMPOUND=*value* is an invalid value or is not allowed in conjunction with other import parameters.

Explanation: 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).

User Response: Change the filetmod option to correct the usage of COMPOUND=x.

SQL3251N   ″number″ more errors occurred during the import.

Explanation: 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.

User Response: 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.

SQL3260N   An unexpected error occurred when accessing the LDAP directory. Error code = *error-code*.

Explanation: An unexpected error occurred when accessing the LDAP directory. The command can not be processed.

User Response: Record the message number (SQLCODE) and the error code. Use the Independent Trace Facility to obtain the DB2 trace. Then contact your IBM service representative.

SQL3261N **The REGISTER LDAP command did not complete successfully because a required input parameter was not specified. Reason code =** *reason-code***.**

Explanation: The REGISTER LDAP command did not complete successfully because a required input parameter was not specified as indicated by the following reason codes:

1 The Network Id parameter was not specified.

2 The Partner LU parameter was not specified.

3 The Transaction Program (TP) name parameter was not specified.

4 The Mode parameter was not specified.

5 The Netbios NNAME parameter was not specified.

6 The TCP/IP hostname parameter was not specified.

7 The TCP/IP service name parameter was not specified.

8 The IPX Address was not specified.

9 The computer name was not specified.

10 The instance name was not specified.

User Response: Re-submit the command with the required input parameter

---

SQL3262N **The TCP/IP service name** *name* **is not valid.**

Explanation: The TCP/IP service name specified is not valid.

User Response: Ensure that the TCP/IP service name is configured and reserved in the local etc/services file, then re-submit the command. Alternatively, specify the port number that is assigned to the TCP/IP service name.

SQL3263N **The protocol type is not supported.**

Explanation: The protocol type specified is not supported for the command.

User Response: Re-submit the command using a supported protocol type.

SQL3264N **The DB2 server has not been registered in LDAP.**

Explanation: The command did not complete successfully because the DB2 server had not been registered in LDAP.

User Response: Register the DB2 server in LDAP using the REGISTER LDAP command. Then re-submit the command.

SQL3265N **An unexpected error occurred during LDAP authentication.**

Explanation: Unable to authenticate the LDAP user due to unexpected LDAP system error.

User Response: Use the Independent Trace Facility to obtain the DB2 trace. Then contact your IBM service representative.

SQL3266N **The LDAP user password is incorrect.**

Explanation: The specified password is not a correct password for the specified user Distinguished Name (DN).

User Response: Re-submit the command using the correct password.

SQL3267N *authid* **does not have sufficient authority to perform the requested command.**

Explanation: The command did not complete successfully because the LDAP user did not have sufficient authority to perform the required task.

User Response: Ensure that the LDAP user has the authorization to perform the operation.

SQL3268N **The LDAP schema is not compatible with the current release of DB2.**

Explanation: The LDAP schema defined at the server does not contain the definition for the DB2 object classes and/or attributes used in the current release of DB2.

User Response: Refer to the Administration Guide on how to extend the LDAP schema with DB2 object classes and attributes.

SQL3269N **The LDAP server is not available.**

Explanation: DB2 was not able to access information in the LDAP directory because the LDAP server was not available.

User Response: Perform the following actions:

1. Ensure that the LDAP server is active.
2. Verify that TCP/IP is configured correctly on your machine
3. Verify that the DB2LDAPHOST registry variable is set to the TCP/IP hostname and port number of the LDAP server by running the "db2set DB2LDAPHOST" command. If the DB2LDAPHOST is not set, you can set it by using the "db2set DB2LDAPHOST=<host-name>:<port-number>" command, where <host-name> is the TCP/IP hostname of the LDAP server and <port-number> is the TCP/IP port number of the LDAP server. The default port number is 389.

SQL3270N **The LDAP user's Distinguished Name (DN) is not valid.**

Explanation: The LDAP user's Distinguished Name (DN) is not valid.

User Response: Re-submit the command using a valid LDAP user's DN.

SQL3271N **The LDAP user's Distinguished Name (DN) and/or password is not defined for the current logon user.**

Explanation: When setting user preferences such as CLI configuration or DB2 registry variable, the LDAP user's DN and password must be defined for the current logon user.

User Response: Refer to the IBM eNetwork LDAP documentation on how to configure the LDAP user's DN and password for the current logon user.

SQL3272N **The node** *node-name* **was not found in the LDAP directory.**

Explanation: The command did not complete successfully because the node *node-name* was not found in the LDAP directory.

User Response: Ensure the node name is correct and re-submit the command.

SQL3273N **The database** *database-alias* **was not found in the LDAP directory.**

Explanation: The command did not complete successfully because the database *database-alias* was not found in the LDAP directory.

User Response: Ensure the database name is correct and re-submit the command.

SQL3274W **The database was created successfully. However, the database was not cataloged in the LDAP directory. SQLCODE =** *sqlcode***.**

Explanation: The database was created successfully. However, the database could not be cataloged in the LDAP directory because there was an error occurred during the operation.

User Response: Correct the error as indicated in the SQLCODE. Then catalog the database in the LDAP directory using the CATALOG LDAP DATABASE command.

SQL3275W **The database was dropped successfully. However, the database was not uncataloged in the LDAP directory. SQLCODE =** *sqlcode***.**

Explanation: The database was dropped successfully. However, the database could not be uncataloged in the LDAP directory because there was an error occurred during the operation.

User Response: Correct the error as indicated in the SQLCODE. Then uncatalog the database in the LDAP directory using the UNCATALOG LDAP DATABASE command.

SQL3276N **Unable to obtain the LDAP naming context.**

Explanation: Unable to query for the LDAP naming context for the LDAP server.

User Response: Contact your LDAP directory administrator to obtain the LDAP naming context for the LDAP server that you are using. If you are using IBM eNetwork Directory V2.1, this is the name of the LDAP suffix. Then set the naming context for the current machine using the ″db2set DB2LDAP_BASEDN=<naming-context>″ command.

SQL3277N **The database** *database-alias* **already exists in the LDAP directory.**

Explanation: The command did not complete successfully because another database of the same name already exists in the LDAP directory.

User Response: Re-submit the command using a different alias name.

SQL3278N **The node** *node* **already exists in the LDAP directory.**

Explanation: The command did not complete successfully because another node of the same name already exists in the LDAP directory.

User Response: Re-submit the command using a different alias name.

SQL3279N **The command did not complete successfully because LDAP is disabled.**

Explanation: The command did not complete successfully because the LDAP support is disabled on the current machine.

User Response: If the LDAP support is installed, enable the LDAP support by running the command: ″db2set DB2_ENABLE_LDAP=YES″.

If the LDAP support is not installed, you need to run the setup program and select to install the LDAP support.

SQL3280N **The attempt to connect to a DRDA server failed.**

Explanation: The attempt to connect to a DRDA server failed because DB2 Connect is not installed on this DB2 client and the gateway-node was not specified when cataloging this LDAP database.

User Response: Either install DB2 Connect on this DB2 client or recatalog this LDAP database with a valid gateway-node.

SQL3281N **The OSTYPE parameter is not valid.**

Explanation: The database server was not registered in LDAP because the OSTYPE parameter specified was not valid. The OSTYPE parameter describes the operating system type of the server.

User Response: Resubmit the command specifying an operating system type (OSTYPE) that is supported by DB2.

SQL3282N **The supplied credentials are not valid.**

Explanation: Either or both the distinguished name (DN) of the user and the password that were specified were not valid.

User Response: Resubmit the command using valid values for both the distinguished name

(DN) of the user and the password.

SQL3283W **The database manager configuration was updated successfully. However, the protocol information was not updated in the LDAP directory. SQLCODE =** *sqlcode-value***.**

Explanation: The database manager configuration was updated successfully. However, the protocol information could not be updated in the LDAP directory because an error occurred during the LDAP operation.

User Response: Correct the error as indicated in

the SQLCODE. Then update the protocol information in the LDAP directory using the UPDATE LDAP NODE command.

SQL3284N **The nodetype parameter is not valid.**

Explanation: The database server was not registered in LDAP because the nodetype parameter specified was not valid.

User Response: Use a valid nodetype when registering the database server in LDAP. Valid values for the nodetype parameter are: SERVER, MPP, and DCS.

SQL3300 - SQL3399

SQL3300N **The records in the input file are not in the correct sequence.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Regenerate the WSF file with a supported level of the Lotus product.

SQL3302N **An EOF record was found before any data was imported.**

Explanation: 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.

User Response: 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 Action String parameter.**

Explanation: File types other than IXF are not allowed with the CREATE or REPLACE_CREATE keywords in the *Action String* (e.g. "REPLACE into ...") parameter.

The IMPORT utility stops processing. No data is imported.

User Response: Change the file type to IXF, or use INSERT, INSERT_UPDATE or REPLACE.

SQL3304N   The table does not exist.

Explanation: The parameters specified on the command require that the table exists.

The utility stops processing. No data is loaded.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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 METHOD parameter does not match the number of entries in the Action String parameter, or a column specified in the METHOD parameter does not exist.

Explanation: The CREATE or REPLACE_CREATE option was specified on the IMPORT command. Either:

- If the NAMES or POSITIONS method indicator was specified in the *METHOD* parameter, the number of columns explicitly specified in *METHOD* is not equal to the number of columns explicitly specified in the *Action String* (e.g. "REPLACE into ...") parameter.
- If the DEFAULT method indicator was specified in the *METHOD* parameter, the number of columns in the PC/IXF file is less than the number of columns specified in the *Action String* parameter.
- One column specified in the *METHOD* parameter does not exist in the PC/IXF file.

The IMPORT utility stops processing. No table is created.

User Response: Either correct the columns specified in the *METHOD* and *Action String* parameters or correct the columns specified in the *METHOD* 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Verify the column definition information in the input file.

SQL3313N **The disk is full. Processing was ended.**

Explanation: 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.

User Response: 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 **The date and time fields in an A record do not match the date and time fields in the H record.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the command.

SQL3317N **The string pointed to by the filetmod parameter contains conflicting information.**

Explanation: 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.

User Response: Change the *filetmod* string to define only one generation and product family. Resubmit the command.

**SQL3318N   There are duplicate keywords in
the filetmod parameter.**

Explanation: 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.

User Response: Resubmit the command with a
correct *filetmod* parameter.

**SQL3319N   An SQL error *sqlcode* occurred
while creating the table.**

Explanation: An SQL error occurred while
creating the table.

The IMPORT utility stops processing. The table
was not created. No data was imported.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Stop and restart DB2 and
resubmit the utility.

**SQL3324N   Column *name* has a type of *type*
which is not recognized.**

Explanation: A column of data returned by the
SQL statement is not supported.

Federated system users: The data type of the
data that you want is not supported by either the
federated server or the data source that you want
to access.

User Response: Correct the error and retry the
command.

Federated system users: Specify a supported data
type and resubmit the program.

**SQL3325W   All the columns in row
row-number have NULL values; the
row will not be included in the
WSF data file.**

Explanation: 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.

User Response: No action required. This is an informational message.

SQL3326N   The column list following the table name in the Action String parameter is invalid.

Explanation: When you call IMPORT or LOAD with an *Action String* (e.g. "REPLACE into ...") parameter which contains a column list after the table name, this message will result if it is invalid. For example, the following *Action String* 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.

User Response: Change the *Action String* 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*).

Explanation: A system error occurred during processing.

The utility stops processing.

User Response: 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.

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.

Explanation: Only even length character fields can be loaded into graphic database columns.

The row is not loaded.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: This message may accompany another error message. It indicates that the

maximum number of open files has been reached.

The command cannot be processed.

User Response: Terminate other applications to reduce the number of files open and retry the utility.

SQL3333C   The file or directory does not exist.

Explanation: 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.

User Response: Resubmit the command with the correct file name, including a valid path.

SQL3334C   There is not enough storage available.

Explanation: This message may accompany another error message. It indicates there is not enough storage available to open the file.

The command cannot be processed.

User Response: 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.

Explanation: This message may accompany another error message. It indicates there is no available space for writing on the device.

The command cannot be processed.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Query the remote database to determine if the utility operation was successful.

SQL3340N   Unable to perform a load with concurrent read access to the table. Reason code = *reason-code*.

Explanation: The ALLOW READ ACCESS option of the load command is not supported in the following cases as specified in the *reason-code*:

1. When using LOAD REPLACE.
2. When using INDEXING MODE DEFERRED.
3. When the target table is in the CHECK PENDING state and is not in the READ ACCESS ONLY state.
4. When the indexes are marked invalid.
5. When using LOAD TERMINATE or LOAD RESTART on a load that did not use the ALLOW READ ACCESS option.

User Response: Resubmit the command using

the ALLOW NO ACCESS option.

SQL3341N   Invalid table space name provided with the USE option. Reason code = *reason-code*.

Explanation: Only system temporary table spaces can be used for rebuilding the indexes in a table space other than the index table space. The page size of the system temporary table space must match the page size of the index table space.

User Response: Resubmit the command with a table space name referring to a system temporary table space with the correct page size. The *reason-code* describes the failure as follows:

1. Table space name in USE clause not found.
2. Table space must be a system temporary table space.
3. Page size in system temporary table space must match page size in index table space.

SQL3346W   USE TablespaceName option ignored. Reason code = *reason-code*.

Explanation: USE TablespaceName is only valid for ALLOW READ ACCESS loads with INDEXING MODE REBUILD. If INDEXING MODE AUTOSELECT is specified, load will use the alternate table space only if it chooses to rebuild the indexes.

User Response: Refer to the reason code for an explanation.

1. Indexing mode incompatible with USE TablespaceName option.
2. Table has no indexes.
3. LOAD TERMINATE does not need to use a separate table space.
4. USE TablespaceName is only supported for ALLOW READ ACCESS loads.

SQL3400 - SQL3499

SQL3400N   The method specified in METHOD is not valid for Non-Delimited ASCII files. It must be 'L' for locations.

Explanation: When loading from a non-delimited ASCII file, columns must be selected by locations in the file.

The command cannot be processed.

User Response: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3401N   The method specified in METHOD is not valid for any filetype.

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3403N The beginning-ending location pair for inserting into column *name* **is not valid.**

Explanation: 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.

User Response: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3404N The beginning-ending location pair for inserting into column *name* **is not valid for a number.**

Explanation: 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.

User Response: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3405N The beginning-ending location pair for inserting into column *name* **is not valid for a date.**

Explanation: 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.

User Response: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3406N The beginning-ending location pair for inserting into column *name* **is not valid for a time.**

Explanation: 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.

User Response: Resubmit the command with a valid set of locations for the columns in the source file.

SQL3407N The beginning-ending location pair for inserting into column *name* **is not valid for a timestamp.**

Explanation: 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.

User Response: Resubmit the command with a valid set of locations for the columns in the file.

SQL3408W The beginning-ending location pair for inserting into column *name* **defines a field larger than the target column. Data may be truncated.**

Explanation: 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.

User Response: No action is required.

SQL3409W **The beginning-ending location pair for inserting into column** *name* **defines a field shorter than the target fixed-length column. Data will be padded.**

Explanation: 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.

User Response: No action is required.

SQL3410N **The beginning-ending location pair for inserting into column** *name* **is not valid for a graphic column.**

Explanation: The field specification for locating the input data in the ASCII file that is to be inserted into the indicated database column defines a field with an odd number of bytes.

The command cannot be processed.

User Response: 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 inserted.**

Explanation: 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 inserted.

User Response: 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 inserted.**

Explanation: 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 inserted 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 inserted.

User Response: 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 inserted.**

Explanation: 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 inserted.

User Response: 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.**

Explanation: 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.

User Response: Depending on where the interrupt occurred, it may be possible to restart the load at the build phase.

SQL3415W   **Failed to convert the field value in row** *row-number* **and column** *column-number* **from the codepage of the input data file to the codepage of the database. A null value was loaded.**

Explanation: The value in the field indicated cannot be converted from the codepage of the input data file to the codepage of the database.

User Response: If a null value is not acceptable, correct the input data file and resubmit the

command, or edit the data in the table.

SQL3416W   **Failed to convert the field value in row** *row-number* **and column** *column-number* **from the codepage of the input data file to the codepage of the database. The row was not loaded.**

Explanation: The value in the field indicated cannot be converted from the codepage of the input data file to the codepage of the database.

User Response: No action is required. If the row is necessary, correct the input data file and resubmit the command, or edit the data in the table.

SQL3500 - SQL3599

SQL3500W   **The utility is beginning the** *phase* **phase at time** *timestamp***.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: No action is required.

SQL3502N   **The utility has encountered** *number* **warnings which exceeds the total number of warnings allowed.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: No response required.

SQL3504W **A consistency point is being established.**

Explanation: 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.

User Response: 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.**

Explanation: In the *filetmod* parameter, the RECLEN option was specified for an ASC file. The length specified is not valid.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Correct the parameter and resubmit the command.

SQL3508N **Error in accessing a file or path of type** *file-type* **during load or load query. Reason code:** *reason-code*. **Path:** *path/file*.

Explanation: An error occurred while trying to access a file during load or load query processing. The utility stops processing.

User Response: If doing a load and the table space is not in load pending state, correct the problem and invoke the load utility again. If the table space is in load pending state, then invoke the load utility in RESTART or REPLACE mode, or restore a backup of the table space(s). The state of the table space 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 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 an 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 TEMPFILES_PATH.

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.

TEMPFILES_PATH

Ensure that the tempfiles path parameter is specified properly. See the *Data Movement Utilities Guide and Reference* for more details on this parameter.

SQL3509W   The utility has deleted *number* **rows from the table.**

Explanation: 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.

User Response: No action is required.

SQL3510N   The working directory for the sort phase cannot be accessed.

Explanation: One or more of the specified working directories for the sort phase either does not exist or has no read/write permission.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: DB2CS files cannot be loaded into a database with a codepage that differs from the original database.

User Response: 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*.

Explanation: A system error occurred during database utility processing.

User Response: 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 vendor sort utility.

  Try the load again using the DB2 Client/Server sort utility instead of the vendor sort. To do this, reset the Profile Registry value at the server to blank. You may have to restart the Database Manager in order to pick up the new Profile Registry value. If the error persists, provide the function, reason code and error code to your vendor sort technical service representative.

SQL3515W **The utility has finished the** *phase* **phase at time** *timestamp*.

Explanation: This is an informational message indicating that a phase has finished. The phases are (in order of appearance):

- LOAD
- BUILD
- DELETE

User Response: No action is required.

SQL3516N   The utility could not restart the load as specified.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: Copy the record 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.

Explanation: 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.

User Response: 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***.**

Explanation: The Load Utility is about to attempt to perform a Consistency Point to commit the table data that has already been loaded.

User Response: 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.

Explanation: The Consistency Point performed by Load was successful.

User Response: This is an informational message only. No response is required.

SQL3521N   The input source file *sequence-num* **was not provided.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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*.

Explanation: No messages are returned from a query of the LOAD temporary message file. The possible return codes are:

1   No LOAD temporary message file exists.

2   There are no messages in the LOAD temporary message file.

User Response: Ensure that a valid table name has been specified. If the table name 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 LOAD temporary message file will not be created until the LOAD utility is in progress and it will be deleted after the completion of the LOAD utility.

SQL3524N **Option** *option* **has an invalid value of** *value*.

Explanation: The value provided must be an integer value. Its range for each option is as follows:

1. TOTALFREESPACE: the value must be in the range of 0 to 100, and is interpreted as a percentage of the total pages in the table that are to be appended to the end of the table as free space.

2. PAGEFREESPACE: the value must be in the range of 0 to 100, and is interpreted as a percentage of each data page that is to be left as free space.

3. INDEXFREESPACE: the value must be in the range of 0 to 99, and is interpreted as a percentage of each index page that is to be left as free space when loading indexes.

The utility stops processing.

User Response: Correct the value and resubmit the command.

SQL3525N **The** *option-1* **option is incompatible with the** *option-2* **option.**

Explanation: Incompatible options have been provided to the utility.

User Response: Remove or modify one of the options and resubmit the command. See the Command Reference for more information on valid options.

SQL3526N **The modifier clause** *clause* **is inconsistent with the current load options. Reason code:** *reason-code*.

Explanation: The load file type mode (modifier) indicated, is incompatible with your load/import/export command. This is because of one of the following reasons:

1. The current option requires you to specify the RECLEN and NOEOFCHAR file type modifiers. One or more of these options are missing from your command.

2. The indicated option, such as DEL or ASC, is inconsistent with the format of the input or output data file.

3. Generated or identity related file type modifiers have been specified but the target table contains no such columns.

4. When a pre-Version 8 client is used, it is not possible to load a table having a non-identity generated column in a unique index, unless the column is specified in the INCLUDE clause of the CREATE INDEX statement or

the GENERATEDOVERRIDE file type modifier is used. For pre-Version 8 clients, it is also necessary to specify the GENERATEDOVERRIDE file type modifier when loading a table that has a non-identity generated column in its ORGANIZE BY clause.

5. The IDENTITYOVERRIDE file type modifier cannot be used when loading a table with a GENERATED BY DEFAULT identity column.

6. The LOBSINFILE file type modifier cannot be specified in a partitioned database load when a generated column, that is part of the partitioning key, is defined in terms of a long field or LOB column.

7. Use of the GENERATEDMISSING or IDENTITYMISSING modifier in the current load or import command would result in all columns in the table being excluded from the load operation.

User Response: Check the requirements of the option you are using. Reissue the command using modifier clauses (file type modes) and utility options that are consistent.

SQL3527N   **The number specified in the FILETMOD parameter for the CODEPAGE option is not valid.**

Explanation: In the FILETMOD parameter, the CODEPAGE option was invalid.

User Response: Correct the codepage number and resubmit the command.

SQL3528W   **The delimiter (column delimiter, string delimiter, or decimal point) specified in the CLP command may get translated from the application codepage to the database codepage.**

Explanation: When a CLP command is sent from the client to the server, it may get translated from the codepage of client to the codepage of server, if they are different.

User Response: To ensure that delimiters are not converted, they should be specified in hex format.

SQL3529N   **The *operation-name* operation encountered the unsupported data type *data-type* in column *column-number*.**

Explanation: The *operation-name* operation does not support data type *data-type* found in column *column-number*.

User Response: Please check your table definition and the Data Movement Guide for supported data types.

SQL3530I   **The Load Query utility is monitoring *agenttype* progress on partition *partitionnumber*.**

Explanation: The Load Query utility was invoked in an MPP environment.

User Response: No action is necessary.

SQL3531I   **A LOAD RESTART has occurred.**

Explanation: The RESTART option was given for the Load currently being queried.

User Response: No action is necessary.

SQL3532I   **The Load utility is currently in the *phase* phase.**

Explanation: This is an informational message indicating the phase of the Load currently being queried.

User Response: No action is necessary.

SQL3533I   **The Load utility is currently building index *number* of *number*.**

Explanation: This is an informational message returned if the Load currently being queried is in the BUILD phase.

User Response: No action is necessary.

SQL3534I   **The Load DELETE phase is approximately** *number* **percent complete.**

Explanation: This is an informational message returned if the Load currently being queried is in the DELETE phase.

User Response: No action is necessary.

SQL3535W   **The LOAD command parameter** *parameter-name* **is no longer supported; its value will be ignored by the LOAD utility.**

Explanation: Your LOAD command contains a parameter that is no longer supported.

User Response: Please refer to the DB2 documentation for information on temporary sort space for LOAD and suggestions related to LOAD performance tuning.

SQL3536N   **System temporary table space** *table-space-name* **is full.**

Explanation: The LOAD utility has encountered a table space full condition while sorting index keys.

User Response: Ensure that there is enough space allocated to the system temporary table space *table-space-name*, at least two times the total size of the indexes being built. Restart the LOAD utility.

SQL3537N   **Sort memory could not be allocated during the execution of the LOAD utility.**

Explanation: Insufficient process virtual memory is available to the LOAD utility for sort processing.

User Response: Terminate the application on receipt of this message. Ensure there is enough virtual memory available for sort processing.

Possible solutions include:

• Disconnect all applications from the database and decrease the size of the sort heap

parameter (sortheap) in the corresponding database configuration file.

• Remove background processes and/or terminate other currently executing applications.

• Increase the amount of virtual memory available.

SQL3538N   **The LOAD QUERY utility failed because multiple LOADs are using the same temporary files path.**

Explanation: At least one other LOAD was invoked using the same TEMPFILES PATH as the queried LOAD and is still in progress. The LOAD QUERY utility cannot uniquely determine which LOAD to query.

User Response: Use the TABLE parameter of LOAD QUERY instead.

SQL3539N   **LOAD RESTART cannot proceed because LOAD TERMINATE has been attempted at least once.**

Explanation: LOAD TERMINATE can only be followed by LOAD TERMINATE.

User Response: User can proceed with LOAD TERMINATE only.

SQL3550W   **The field value in row** *row-number* **and column** *column-number* **is not NULL, but the target column has been defined as GENERATED ALWAYS.**

Explanation: A non NULL field value was encountered in the input file. Because the target column is of type GENERATED ALWAYS, the value cannot be loaded. The column number specifies the field within the row of the missing data.

User Response: For LOAD, an explicit, non NULL field value can be loaded into a GENERATED ALWAYS identity column only if the identityoverride file type modifier is used. For non-identity GENERATED ALWAYS columns, the generatedoverride file type modifier

can be used to load explicit, non NULL values into a row. If using these modifiers is not appropriate, the field value must be replaced with a NULL if LOAD is to accept the row.

For IMPORT there is no way to override a GENERATED ALWAYS column. The field value must be removed and replaced with a NULL if the utility is to accept the row.

SQL3551W   **The table contains at least one GENERATED ALWAYS column that the utility will override.**

Explanation: An "override" file type modifier (for example, IDENTITYOVERRIDE or

GENERATEDOVERRIDE) has been specified.

For IDENTITYOVERRIDE, this may violate the uniqueness property of an identity column defined as GENERATED ALWAYS.

For GENERATEDOVERRIDE, this may result in a non-identity GENERATED ALWAYS column containing a value that does not correspond to its column definition.

User Response: No action is required.

SQL3600 - SQL3699

SQL3600N   **The IMMEDIATE CHECKED option of the SET INTEGRITY statement is not valid since the table** *table-name* **is a user maintained materialized query table or not in the check pending state.**

Explanation: The data is checked for constraint violations only if the table has been placed in the check pending state, is not a user maintained materialized query table, or if one of its ancestors that has been placed into the check pending state using the CASCADE DEFERRED option is also in the invocation list and all intermediate ancestors are also in the list.

User Response:

1. Use the SET INTEGRITY statement with the OFF option to place a table in the check pending state.

2. Use the IMMEDIATE UNCHECKED option for a user maintained materialized query table.

3. Include an ancestor of the table to the list of tables to be checked. The ancestor must be in the check pending state and all intermediate ancestors must also be in the list.

4. If the table is in the no data movement mode, specify the SET INTEGRITY...FULL ACCESS statement to force the table into the full

access mode. Note that all dependent immediate materialized query tables that remain in the check pending state are also forced to be fully recomputed in subsequent REFRESH statement, and all dependent immediate staging tables that remain in the check pending state can no longer be used to refresh their dependent materialized query tables.

sqlcode: -3600

sqlstate: 51027

SQL3601W   **The statement caused one or more tables to automatically be placed in the check pending state.**

Explanation: Dependent tables, descendent tables, descendent refresh immediate materialized query tables or staging tables were put in the check pending state. This is done to enforce referential integrity constraints or to enforce the data relationship between the base table, the dependent materialized query tables, and the staging tables.

User Response: To validate the integrity for all the dependent tables, descendent tables, descendent refresh immediate materialized query tables, or staging tables, execute the SET INTEGRITY...IMMEDIATE CHECKED statement

for these tables. To determine which tables are in the check pending state, issue the following query:

```
SELECT TABSCHEMA, TABNAME, STATUS
FROM SYSCAT.TABLES
WHERE STATUS = 'C'
```

sqlcode: +3601

sqlstate: 01586

SQL3602W   **Check data processing found constraint violations and moved them to exception tables.**

Explanation: There are rows that violate constraints that were specified to be checked by the SET INTEGRITY statement execution. These were moved to exceptions tables.

User Response: 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 INTEGRITY statement has found integrity violation involving a constraint with name** *name*.

Explanation: A row has been found to violate a constraint defined on a table specified to be checked by the SET INTEGRITY statement. The name *name* is either a constraint name or the name of a generated column.

User Response: The row has not been deleted from the table since the FOR EXCEPTION option was not used.

It is advisable to execute the SET INTEGRITY 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 INTEGRITY statement or LOAD utility does not have the proper structure, has been defined with unique indexes, constraints, generated columns, or triggers, or is in the check pending state itself.**

Explanation: 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 may not be any generated columns in the exception table. 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.

User Response: Create the exception table as indicated in the relevant section of the documentation and re-run the statement or utility.

sqlcode: -3604

sqlstate: 428A5

SQL3605N   **Table** *table-name* **named in the SET INTEGRITY statement is either not listed to be checked or is an exception table specified more than once.**

Explanation: When the FOR EXCEPTION clause is specified in the SET INTEGRITY 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

User Response: Correct the table names and execute the command again.

sqlcode: -3605

sqlstate: 428A6

SQL3606N **There is a mismatch in the number of tables being checked and in the number of exception tables specified in the SET INTEGRITY statement.**

Explanation: There must be a one-to-one correspondence between the exception tables and the tables that are not materialized query tables or staging tables provided in the invocation list. The tables in the invocation list that are materialized query tables or staging tables must not have corresponding exception tables.

User Response: Create the missing exception table for the table that is not a materialized query table or staging table if it does not already exist and specify it in the invocation list in order to execute the command again. Do not specify exception tables for materialized query tables and staging tables.

sqlcode: -3606

sqlstate: 428A7

SQL3608N **Cannot check a dependent table** *dependent-table-name* **using the SET INTEGRITY statement while the parent table or underlying table** *parent-table-name* **is in the check pending state or if it will be put into the check pending state by the SET INTEGRITY statement.**

Explanation: The parent table or underlying table must not be in the check pending state either before or after the SET INTEGRITY statement, or it must be included in the invocation list in order to check a dependent table.

User Response: Ensure that the parent table is not in the check pending state by executing the SET INTEGRITY statement to check the parent table. If the dependent table is a materialized query table or staging table, ensure that the underlying table is not in the check pending state by executing the SET INTEGRITY statement to check the underlying table.

If the dependent table is not a materialized query table or staging 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 statement could still fail if there are rows in the parent table which violate the constraint and these rows are not deleted and put into an exception table. This can happen if the FOR EXCEPTION option is not used.

If the dependent table is a materialized query table or staging table, it is recommended to check the underlying table first. It is also possible to refresh the dependent materialized query table and include the underlying table in the invocation list. In this case, the statement could still fail if there are rows in the parent table which violate the constraint and these rows are not deleted and put into an exception table. This can happen 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

SQL3700 - SQL3799

SQL3700W **Device** *device* **is full. There are** *active-devices* **other active devices. Please mount new media or take appropriate action.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: No action required.

SQL3702W **Warning. SQLCODE** *sqlcode* **was received for device** *device***. Unload will attempt to continue without this device.**

Explanation: 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.

User Response: 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 Message Reference.

SQL3703W *xxx* **of** *yyy* **number of type** *type* **pages have been unloaded and sent to be written out to the target media.**

Explanation: 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.

User Response: This is an informational message. No action is required.

SQL3704N **The num_buffers parameter specified is invalid.**

Explanation: 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.

User Response: 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.**

Explanation: 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 16384.

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.

User Response: Reissue the command with a valid buffer size.

SQL3706N **A disk full error was encountered on** *path/file***.**

Explanation: A disk full error was encountered during the processing of a database utility. The utility stops.

User Response: 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***.**

Explanation: The sort memory size is not large enough for sorting the keys of the index(es).

User Response: 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 **An error was encountered while opening the copy location file. Open file error code is** *errcode***.**

Explanation: During load recovery, an error was encountered while opening the copy location file. The operating system file open return code is returned.

User Response: 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*

Explanation: 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.

User Response: 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*
on node *node-number* **failed due to**
error *sqlcode* **with additional**
information *additional-info*.

Explanation: A severe error occurred during
load recovery. The utility stops processing.

(Note : if you are using a partitioned database
server, the node number indicates which node
the error occurred on. Otherwise, it is not
pertinent and should be ignored.)

User Response: 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.

Explanation: Load recovery is in progress and
one of the parameters passed in is not valid for
the current state of load recovery.

SQL3800 - SQL3899

SQL3802N **An invalid quiesce mode**
quiesce-mode **was encountered.**

Explanation: An invalid quiesce mode was
passed to the quiesce API.

User Response: Resubmit the command with
the correct parameters.

SQL3804N **The index is invalid.**

Explanation: An invalid index was encountered
during the processing of a utility command.

User Response: Check in the *Administration*
Guide to determine the appropriate way to
revalidate the index and resubmit the command
after the condition is corrected.

User Response: 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*
on node *node-number* **is pending**
due to warning *sqlcode* **with**
additional information
additional-info.

Explanation: A warning condition is
encountered during load recovery. The utility
waits for a response to continue.

(Note : if you are using a partitioned database
server, the node number indicates which node
the error occurred on. Otherwise, it is not
pertinent and should be ignored.)

User Response: 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.

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*.

Explanation: 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 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 table space cannot be quiesced.

06 load copy not allowed when table space is in backup pending state.

07 Attempt to restart a load at an incorrect phase

User Response: Corresponding to each of the above reasons:

01 Resubmit the command with the correct loadapi action or quiescemode, or correct the state of the table spaces for the table.

02 Resubmit the command with the correct load action or complete the current logical unit of work by issuing either a COMMIT or ROLLBACK.

03 Resubmit the command with the correct load action.

04 Determine which table spaces for the table have reached the maximum number of quiescers. QUIESCE RESET these table spaces.

05 Resubmit the command with a table which is not in the system catalog table space.

06 Resubmit the command omitting the copy parameter

SQL3900 - SQL3999

SQL3901N   A non-severe system error occurred. Reason code *reason-code***.**

Explanation: Processing ended because of a non-severe system error.

User Response: If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Then contact your technical service representative with the following information:

- Problem description

07 Determine at which phase the load should be restarted and resubmit the command with the correct phase.

SQL3806N   Not all table constraints are off for the table to be loaded.

Explanation: One or more constraints for the table to be loaded are still on when the load API is invoked.

User Response: Resubmit the command after all table constraints are turned off.

SQL3807N   Instance or database *name* **quiesce is pending.**

Explanation: Another user has submitted the quiesce command and it has not yet completed.

User Response: Wait for the quiesce to complete.

SQL3808N   Instance or database *name* **unquiesce is pending.**

Explanation: Another user has submitted the unquiesce command and it has not yet completed.

User Response: Wait for the unquiesce to complete.

- SQLCODE and embedded reason code
- SQLCA contents if possible
- Trace file if possible.

SQL3902C   A system error occurred. Further processing is not possible. Reason code = *reason-code***.**

Explanation: A system error occurred.

User Response: If trace was active, invoke the Independent Trace Facility at the operating

system command prompt. 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.

SQL3910I **Synchronization session
completed successfully.**

Explanation:

User Response:

SQL3911I **Test synchronization session
completed successfully.**

Explanation:

User Response:

SQL3912I **STOP completed successfully.**

Explanation:

User Response:

SQL3913I **STOP issued, but no
synchronization session is
currently active.**

Explanation:

User Response:

SQL3914I **A user interrupt was issued. The
synchronization session stopped
successfully.**

Explanation:

User Response:

SQL3915I **A user interrupt was issued
before the results were uploaded
to the satellite control server. The
results will be uploaded during
the next synchronization session.**

Explanation:

User Response:

SQL3916I **A STOP request was received.
Synchronization session stopped
successfully.**

Explanation:

User Response:

SQL3917I **A STOP request was received
before the results were uploaded
to the satellite control server. The
results will be uploaded during
the next synchronization session.**

Explanation:

User Response:

SQL3918I **Synchronization progress
information was obtained
successfully.**

Explanation:

User Response:

SQL3919I **A STOP request was received
before the satellite could contact
the satellite control server.
Synchronization stopped
successfully.**

Explanation:

User Response:

SQL3920I **The application version on this satellite does not match one that is available for the group of this satellite. Synchronization cannot occur.**

Explanation: The application version reported by the satellite does not exist at the satellite control server.

User Response: Contact the help desk or your system administrator.

SQL3921I **This satellite is disabled at the satellite control server. Synchronization cannot occur.**

Explanation: When disabled, the satellite cannot synchronize.

User Response: Contact the help desk or your system administrator.

SQL3930W **There are no synchronization scripts to execute.**

Explanation: No synchronization scripts were downloaded to the satellite for execution.

User Response: Contact the help desk or your system administrator to ensure that synchronization scripts are available on the satellite control database for this satellite.

SQL3931W **The test synchronization session completed successfully. The satellite ID, however, could not be found in the satellite control database.**

Explanation: Either the satellite ID is not defined correctly on the satellite or this satellite has not been defined in the satellite control database.

User Response: If you use the DB2SATELLITEID registry variable, ensure that it is set to the unique ID for the satellite. If you are using the operating system logon ID as the satellite ID, ensure you log on using it.

SQL3932W **The test synchronization session completed successfully. The satellite's application version, however, is not set locally, or does not exist for this satellite's group at the satellite control server.**

Explanation: The application version on the satellite is different from any that are available for this satellite's group.

User Response: Ensure that the application version on the satellite is set to the correct value.

SQL3933W **The test synchronization session completed successfully. The release level of the satellite, however, is not supported by the release level of the satellite control server.**

Explanation: The release level of the satellite must be within the range of one level above to two levels below that of the satellite control server.

User Response: Contact the help desk or your system administrator.

SQL3934W **The test synchronization session completed successfully. This satellite, however, is disabled at the satellite control server.**

Explanation: The satellite has been placed in the disabled state at the satellite control server.

User Response: Contact the help desk or your system administrator.

SQL3935W **The test synchronization session completed successfully. This satellite, however, is in the failed state at the satellite control server.**

Explanation: The satellite is in the failed state at the control server.

User Response: Contact the help desk or your system administrator.

SQL3936W No progress information is available.

Explanation: The synchronization session has not reached a stage where the progress information has been logged, or there is no active synchronization session for this satellite.

User Response: Ensure the synchronization session is active, or try querying for progress information later.

SQL3937W The application version for this satellite does not match one that is available for this satellite's group.

Explanation: A satellite can only synchronize with a particular application version for its group. The application version for this satellite is not available for the satellite's group at the control server.

User Response: Ensure that the application version on the satellite is set to the correct value.

SQL3938W An interrupt was issued during script execution. The synchronization session was stopped, but the satellite may be in an inconsistent state.

Explanation: An interrupt was issued while the script execution phase of the synchronization process was being performed. The synchronization session was stopped but the satellite may be in an inconsistent state because a script may have been halted at an inappropriate place.

User Response: Contact the help desk or your system administrator.

SQL3942I Synchronization session identifier was set successfully for the satellite.

Explanation: The session identifier was set successfully for the satellite.

User Response: No action is required.

SQL3943N Synchronization session identifier exceeds the maximum length of *length* characters.

Explanation: The specified synchronization session identifier is longer than the allowed length of *length* characters.

User Response: Ensure that the identifier is no more than nnn characters.

SQL3944I Synchronization session identifier was reset successfully for the satellite.

Explanation: The session identifier was reset successfully for the satellite.

User Response: No action is required.

SQL3945I Synchronization session identifier for the satellite was retrieved successfully.

Explanation: The session identifier for this satellite was found and returned successfully.

User Response: No action is required.

SQL3946N Synchronization session identifier operation failed.

Explanation: The synchronization session identifier operation has failed for an unknown reason.

User Response: Ensure that the product is installed correctly. If the failure persists, contact DB2 service.

SQL3950N A synchronization session is active. Only one synchronization session can be active.

Explanation: Only one synchronization session can be active at a time.

User Response: Wait until the current synchronization session completes successfully before initiating another session.

SQL3951N **The satellite ID cannot be found locally.**

Explanation: Either the operating system logon was bypassed or the DB2SATELLITEID registry variable was not set.

User Response: If you are using the operating system logon ID as the satellite ID, log on to the operating system. If you use the DB2SATELLITEID registry variable, ensure that it is set to the unique ID for the satellite.

SQL3952N **The satellite ID could not be found at the satellite control server.**

Explanation: Either the satellite ID is not defined correctly on this satellite or this satellite has not been defined at the satellite control server.

User Response: If you use the DB2SATELLITEID registry variable, ensure that it is set to the unique ID for the satellite. If you are using the operating system logon ID as the satellite ID, ensure you log on using it. Otherwise, contact the help desk or your system administrator.

SQL3953N **This satellite has been disabled at the satellite control server.**

Explanation: The satellite ID has been disabled at the satellite control server.

User Response: Contact the help desk or your system administrator.

SQL3954N **This satellite is in the failed state at the satellite control server.**

Explanation: The satellite is in the failed state because a previous synchronization session failed.

User Response: Contact the help desk or your system administrator.

SQL3955N **The satellite control database name or its alias could not be found.**

Explanation: The satellite control database is not cataloged correctly.

User Response: Contact the help desk or you system administrator.

SQL3956N **The application version for this satellite is not defined locally.**

Explanation: The application version is not defined locally on this satellite or is not defined correctly.

User Response: Ensure that the application version is set to the correct value.

SQL3957N **Cannot connect to the satellite control server because of a communication failure: SQLCODE=*sqlcode* SQLSTATE=*sqlstate*, tokens=*token1*, *token2*, *token3*.**

Explanation: An error has been detected by the communication subsystem. Refer to *sqlcode* for more details.

User Response: Contact the help desk or your system administrator.

SQL3958N **Error occurred during synchronization session: SQLCODE=*sqlcode* SQLSTATE=*sqlstate*, tokens=*token1*, *token2*, *token3*.**

Explanation: An unknown error has been detected by the communication subsystem. Refer to *sqlcode* for more details.

User Response: Contact the help desk or your system administrator.

SQL3959N **Synchronization session cannot start because of a communication failure: SQLCODE=***sqlcode* **SQLSTATE=***sqlstate***, tokens=***token1***,** *token2***,** *token3***.**

Explanation: An error has been detected by the communication subsystem. Refer to *sqlcode* for more details.

User Response: Contact the help desk or your system administrator.

SQL3960N **Cannot upload results to the satellite control server because of a communication failure: SQLCODE=***sqlcode* **SQLSTATE=***sqlstate***, tokens=***token1***,** *token2***,** *token3***.**

Explanation: An error has been detected by the communication subsystem. Refer to *sqlcode* for more details.

User Response: Contact the help desk or your system administrator.

SQL3961N **Cannot authenticate with the satellite control server.**

Explanation: An authentication error was detected while trying to connect to the satellite control database.

User Response: Either the remote administrative user ID or the password required to connect to the satellite control database was not correct, or both. Either provide the correct user ID and password, or contact the help desk or your system administrator.

SQL3962N **Synchronization could not be started because of a database error: SQLCODE=***sqlcode* **SQLSTATE=***sqlstate***, tokens=***token1***,** *token2***,** *token3***.**

Explanation: An error preventing synchronization occurred at the satellite control server.

User Response: Attempt to synchronize again.

If failure persists, contact the help desk or your system administrator.

SQL3963N **Results cannot be uploaded because of a database error: SQLCODE=***sqlcode* **SQLSTATE=***sqlstate***, tokens=***token1***,** *token2***,** *token3***.**

Explanation: An error occurred while uploading results to the satellite control server.

User Response: Contact the help desk or your system administrator.

SQL3964N **Synchronization failed because the release level of the satellite is not supported by the satellite control server.**

Explanation: The release level of the satellite must be within the range of one level above to two levels below that of the satellite control server.

User Response: Contact the help desk or your system administrator.

SQL3965N **Cannot download the synchronization scripts because of a satellite control server failure: SQLCODE=***sqlcode* **SQLSTATE=***sqlstate***, tokens=***token1***,** *token2***,** *token3***.**

Explanation: The satellite cannot download the scripts that are required to synchronize the satellite. One possible reason for this failure is that the control server cannot instantiate parameterized scripts with the attributes of the satellite. Another reason is that the satellite control server was temporarily unable to complete the request because of resource constraints.

User Response: Try the request again. If the failure persists, contact the help desk or your system administrator.

SQL3966N **Synchronization session failed. Reason code** *reason-code*.

Explanation: The synchronization session failed to complete for one of the following reasons:

- (01) Missing authentication information.
- (02) Some scripts required for synchronization are missing.
- (03) System files are corrupted or missing.
- (04) A system error prevented scripts from executing.

User Response: Try the request again. If failure persists, contact the help desk or your system administrator.

SQL3967N **Progress information cannot be retrieved.**

Explanation: Progress on this satellite's synchronization session cannot be retrieved. Either the data is corrupt, or it does not exist.

User Response: If a synchronization session is active and no progress information is available, contact the help desk or your system administrator.

SQL3968N **Synchronization failed to complete successfully because of a script failure. The results of the execution, however, were sent to the satellite control server.**

Explanation: One of the synchronization scripts failed during execution. Either a return code was not in the defined success code set, or the script failed to execute.

User Response: Contact the help desk or your system administrator.

SQL3969N **Synchronization failed because an interrupt was received during script execution.**

Explanation: Script execution fails when an interrupt is received. An abnormal termination of this type will cause the synchronization session to fail because the system may be in an inconsistent state.

User Response: Contact the help desk or your system administrator.

SQL3970N **Synchronization session failed with, SQLCODE** *sqlcode* **SQLSTATE** *sqlatate* **The error was detected at location** *location*.

Explanation: Script execution failed because of an unknown error.

User Response: Contact the help desk or your system administrator.

SQL4000 - SQL4099

SQL4001N **The character** *character* **in column** *column* **on line** *line* **is not valid.**

Explanation: The specified character is not a valid character in SQL statements.

The statement cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Repeat the precompile process with the current version of the precompiler.

SQL4004N **The package name is not valid.**

Explanation: 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.

User Response: 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***.**

Explanation: A syntax error in the SQL statement was detected at the specified token *token*.

The statement cannot be processed.

User Response: Examine the statement, especially around the specified token. Correct the syntax.

SQL4006N **The structures are nested too deeply.**

Explanation: The number of nested structures exceeded the maximum of 25.

The statement cannot be processed.

User Response: Reduce the number of nested structures.

SQL4007N **The host structure** *host-structure* **has no fields.**

Explanation: No field can be found within the host structure *host-structure*.

The statement cannot be processed.

User Response: Add a field to the host structure.

SQL4008N **The host variable** *name* **cannot be uniquely referenced even with full qualification.**

Explanation: The host variable *name* matches with at least one other unqualified or partially qualified host variable even with full qualification.

The command cannot be processed.

User Response: Rename the host variable.

SQL4009N **The expression for data length is invalid.**

Explanation: The expression for data length has syntax errors or is too complex.

The statement cannot be processed.

User Response: Check the syntax of the size expression.

SQL4010N **Illegal nesting of Compound SQL statements.**

Explanation: This error is returned when a BEGIN COMPOUND clause is detected as a sub-statement of a Compound SQL statement.

User Response: Resubmit the precompilation

without the nested BEGIN COMPOUND.

SQL4011N   Invalid SQL sub-statement in a Compound SQL statement.

Explanation: 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.
- RELEASE TO SAVEPOINT; only for non-atomic Compound SQL
- ROLLBACK TO SAVEPOINT; only for non-atomic Compound SQL
- SAVEPOINT; only for non-atomic Compound SQL

User Response: Resubmit the precompilation without the invalid sub-statement.

sqlcode: -4011

sqlstate: 42910

SQL4012N   Invalid use of COMMIT in a Compound SQL statement.

Explanation: This error is returned when a sub-statement is detected after a COMMIT has been found in a COMPOUND SQL statement.

User Response: 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.

Explanation: This error is returned when an END COMPOUND statement has been found without a preceding BEGIN COMPOUND.

User Response: Either remove the END COMPOUND, or add a BEGIN COMPOUND, and resubmit the precompilation.

SQL4014N   Invalid Compound SQL syntax.

Explanation: 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).

User Response: Fix the syntax error and re-precompile.

SQL4015N   An error has occurred during preprocessing.

Explanation: The external preprocessor has terminated with one or more errors.

The command cannot be processed.

User Response: Please refer to the ".err" file for the corresponding source file for more details.

SQL4016N   The preprocessor specified cannot be found.

Explanation: The preprocessor specified through the PREPROCESSOR option cannot be found.

The command cannot be processed.

User Response: Ensure that the preprocessor can be executed from the current directory, and also check the syntax of the PREPROCESSOR option.

SQL4017W Preprocessing has completed successfully.

Explanation: Preprocessing of the input file has completed successfully using the external command specified through the PREPROCESSOR option.

User Response: No action is required.

SQL4018W Starting to process the preprocessed file *preprocessed-file***.**

Explanation: The precompiler is now processing the preprocessed file.

User Response: No action is required.

SQL4019W Completed processing the preprocessed file *preprocessed-file***.**

Explanation: The precompiler has completed processing the preprocessed file.

SQL4100 - SQL4199

SQL4100I The *sqlflag-type* **SQL language syntax is used for syntax checking by the flagger.**

Explanation: The SQL statements that have passed the precompiler checking will be subject to checking against the specified syntax by the flagger. Warning message will be issued against the statement when there is a syntax deviation.

Processing continues.

User Response: None. This is an informational message only.

SQL4102W An SQL syntax deviation has occurred at the token starting with the text *text***.**

Explanation: The flagger has detected a deviation from the SQL language syntax specified in the SQLFLAG precompiler option.

Processing continues.

User Response: Correct the SQL statement.

User Response: No action is required.

SQL4020N The 'long' host variable *token-1* **is not valid. Use 'sqlint32' instead.**

Explanation: If the LONGERROR YES precompile option is in effect or the precompile option LONGERROR has not been specified and the platform has 8-byte 'long's, then the INTEGER host variable must be declared with the data type 'sqlint32', not 'long'.

On 64 bit platforms with 8-byte 'long' types the precompile option LONGERROR NO can be used to specify that 'long' host variables are to be used for BIGINT data types. For maximum portability the use of 'sqlint32' and 'sqlint64' are recommended for INTEGER and BIGINT data types respectively.

User Response: Replace the current data type of the host variable with the data type specified in the message.

SQL4103W A data definition statement is not within a CREATE SCHEMA statement.

Explanation: The FIPS standard requires that all data definition statements are contained within a CREATE SCHEMA statement.

Processing continues.

User Response: Correct the SQL statement.

SQL4104W There is a non-standard embedded comment.

Explanation: A comment embedded in an SQL statement does not meet the requirements of the standard being flagged. The comment does not begin with at least two consecutive hyphens

Processing continues.

User Response: Correct the SQL statement.

SQL4105W **An SQL syntax deviation has occurred. The statement is not complete.**

Explanation: The SQL statement has terminated before all required elements have been found.

Processing continues.

User Response: Correct the SQL statement.

SQL4106W **The identifier** *identifier* **is more than 18 characters.**

Explanation: Authorization identifier, table identifier, column name, correlation name, module name, cursor name, procedure name, or parameter name has a length of more than 18 characters.

Processing continues.

User Response: Correct the SQL statement.

SQL4107W **Column** *column* **has an invalid length, precision, or scale attribute.**

Explanation: One of the following conditions has not been met:
- Length value should be greater than zero.
- Precision value should be greater than zero.
- Scale value should not be greater than precision.

Processing continues.

User Response: Correct the SQL statement.

SQL4108W **An indicator variable has data type other than exact numeric, or has a non-zero scale.**

Explanation: Data type of an indicator variable should be exact numeric with a scale of zero.

Processing continues.

User Response: Correct the SQL statement.

SQL4109W **The SET FUNCTION SPECIFICATION references column** *column***.**

Explanation: One of the following conditions has not been met:
- The COLUMN REFERENCE of a DISTINCT SET FUNCTION cannot reference a column derived from a SET FUNCTION SPECIFICATION.
- COLUMN REFERENCEs in the VALUE EXPRESSION of an ALL SET FUNCTION cannot reference a column derived from a SET FUNCTION SPECIFICATION.

Processing continues.

User Response: Correct the SQL statement.

SQL4110W **A VALUE EXPRESSION containing** *column* **cannot include operators.**

Explanation: A VALUE EXPRESSION cannot include operators in an OUTER REFERENCE COLUMN REFERENCE.

Processing continues.

User Response: Correct the SQL statement.

SQL4111W **The COLUMN REFERENCE is missing or is invalid for the ALL column function** *function***.**

Explanation: The VALUE EXPRESSION of an ALL SET FUNCTION SPECIFICATION should include a COLUMN REFERENCE.

Processing continues.

User Response: Correct the SQL statement.

SQL4112W **Column** *column* **is not unique and requires qualification.**

Explanation: The specified column is not unique within the current scope. Qualification must be given to uniquely identify the required column.

Processing continues.

User Response: Correct the SQL statement.

SQL4113W A VALUE EXPRESSION may not contain a SET FUNCTION SPECIFICATION.

Explanation: The VALUE EXPRESSION of an ALL SET FUNCTION may not contain SET FUNCTION SPECIFICATION.

Processing continues.

User Response: Correct the SQL statement.

SQL4114W Column *column* does not identify a column of a table within the current scope.

Explanation: One of the following conditions has not been met:

- The table or correlation name used as a qualifier does not exist.
- The column name does not exist within the current scope or the scope of the qualifier.

Processing continues.

User Response: Correct the SQL statement.

SQL4115W A column function that contains the OUTER REFERENCE column *column* is not in the subquery of a HAVING clause.

Explanation: A column function must be contained in a subquery of a HAVING clause if it contains an OUTER REFERENCE COLUMN REFERENCE.

Processing continues.

User Response: Correct the SQL statement.

SQL4116W The result of a SUM or AVG function cannot be a character string.

Explanation: Character string is not valid for the result of a SUM or AVG SET FUNCTION SPECIFICATION.

Processing continues.

User Response: Correct the SQL statement.

SQL4117W Operator *operator* is invalid in this context.

Explanation: One of the following conditions has not been met:

- A VALUE EXPRESSION containing a DISTINCT SET FUNCTION cannot include dyadic operators.
- First character following a monadic operator cannot be a plus or minus sign.
- VALUE EXPRESSION should not contain operators if PRIMARY is of type character string.

Processing continues.

User Response: Correct the SQL statement.

SQL4118W The *exptype* EXPRESSION is comparing incompatible data types.

Explanation: The data types of one of the following (identified by exptype) do not match:

- exptype = COMPARISON - Comparison operators must match
- exptype = BETWEEN - The three VALUE EXPRESSIONs must match
- exptype = IN - VALUE EXPRESSION, subquery and all VALUE SPECIFICATIONs must match
- exptype = QUANTIFIED - VALUE EXPRESSION and subquery must match

Processing continues.

User Response: Correct the SQL statement.

SQL4119W The operand in the LIKE predicate is not a character string.

Explanation: One of the following conditions has not been met:

- Data type of the column in a LIKE predicate is not a character string.

- Data type of the pattern in a LIKE predicate is not a character string.

Processing continues.

User Response: Correct the SQL statement.

SQL4120W **The ESCAPE character must be a one-byte character string.**

Explanation: Escape character in a LIKE predicate must have the data type of character string with a length of 1.

Processing continues.

User Response: Correct the SQL statement.

SQL4121W **The WHERE clause, GROUP BY clause, or HAVING clause is invalid for the grouped view** *schema-name*.*view*.

Explanation: If the table identified in the FROM clause is a grouped view, then the TABLE EXPRESSION must not contain a WHERE clause, GROUP BY clause, or HAVING clause.

Processing continues.

User Response: Correct the SQL statement.

SQL4122W *schema-name*.*name* **occurs more than once in a FROM clause.**

Explanation: One of the following conditions has not been met:
- Table name occurs more than once in a FROM clause.
- Correlation name is identical to a table name or to another correlation name in a FROM clause.

Processing continues.

User Response: Correct the SQL statement.

SQL4123W **Only one table reference is allowed in the FROM clause for a GROUPed view.**

Explanation: If the table identified by table name is a GROUPed view, then the FROM clause must contain exactly one table reference.

Processing continues.

User Response: Correct the SQL statement.

SQL4124W **A reference to column** *column* **derived from a column function is invalid in a WHERE clause.**

Explanation: A VALUE EXPRESSION directly contained in the SEARCH CONDITION of a WHERE clause must not include a reference to a column derived from a column function.

Processing continues.

User Response: Correct the SQL statement.

SQL4125W **A HAVING clause must contain the WHERE clause when the WHERE clause has a column function.**

Explanation: If a VALUE EXPRESSION directly contained in the SEARCH CONDITION is a column function, then the WHERE clause must be contained in a HAVING clause.

Processing continues.

User Response: Correct the SQL statement.

SQL4126W **The COLUMN REFERENCE for** *column* **must be an OUTER REFERENCE.**

Explanation: If a VALUE EXPRESSION directly contained in the SEARCH CONDITION is a function, then the COLUMN REFERENCE in the column function expression must be an OUTER REFERENCE.

Processing continues.

User Response: Correct the SQL statement.

SQL4127W **Column** *column* **is duplicated within the current scope.**

Explanation: The specified column is duplicated within the current scope.

Processing continues.

User Response: Correct the SQL statement.

SQL4128W **The COLUMN REFERENCE for** *column-name* **should be a grouping column or should be specified within a column function.**

Explanation: Each COLUMN REFERENCE contained in a subquery in the SEARCH CONDITION of a HAVING clause must reference a grouping column or be specified within a column function.

Processing continues.

User Response: Correct the SQL statement.

SQL4129W **DEGREE of table** *schema-name.table* **should be 1 when using a SELECT LIST of *.**

Explanation: The DEGREE of the TABLE EXPRESSION must be 1 if a SELECT LIST of * is specified in a subquery of any predicate other than an EXISTS predicate.

Processing continues.

User Response: Correct the SQL statement.

SQL4130W **The column function is not valid for the TABLE EXPRESSION beginning with table** *schema-name.table*.

Explanation: One of the following conditions has not been met:

- If the TABLE EXPRESSION is a grouped view, the SELECT LIST of a subquery cannot contain a SET FUNCTION SPECIFICATION.
- If the TABLE EXPRESSION is a grouped view, the SELECT LIST of a QUERY SPECIFICATION cannot contain a column function.

Processing continues.

User Response: Correct the SQL statement.

SQL4131W **The COLUMN REFERENCE for** *column* **is invalid.**

Explanation: One of the following conditions has not been met:

- For a GROUPed table, the COLUMN REFERENCE must reference a GROUPing column or be specified within a SET FUNCTION SPECIFICATION.
- If not a GROUPed table and VALUE EXPRESSION includes a SET FUNCTION SPECIFICATION, then each COLUMN REFERENCE must be specified within a SET FUNCTION SPECIFICATION.

Processing continues.

User Response: Correct the SQL statement.

SQL4132W **DISTINCT is specified more than once.**

Explanation: One of the following conditions has not been met:

- DISTINCT must not be specified more than once in a QUERY SPECIFICATION, excluding any subquery of that QUERY SPECIFICATION.
- DISTINCT must not be specified more than once in a subquery, excluding any subquery contained in that subquery.

Processing continues.

User Response: Correct the SQL statement.

SQL4133W **The COMPARISON PREDICATE subquery cannot contain a GROUP BY or HAVING clause.**

Explanation: If a subquery is specified in a COMPARISON PREDICATE, then the TABLE EXPRESSION whose FROM clause identifies the named table must not contain a GROUP BY clause or a HAVING clause.

Processing continues.

User Response: Correct the SQL statement.

SQL4134W **The COMPARISON PREDICATE subquery cannot identify a GROUPed view.**

Explanation: If a subquery is specified in a COMPARISON PREDICATE, then the TABLE EXPRESSION whose FROM clause identifies the named table must not identify a GROUPed view.

Processing continues.

User Response: Correct the SQL statement.

SQL4135W **There is an invalid AUTHORIZATION IDENTIFIER** *authid*.

Explanation: The AUTHORIZATION IDENTIFIER prefixed to the table name is invalid.

Processing continues.

User Response: Correct the SQL statement.

SQL4136W **Table or view** *schema-name.name* **already exists.**

Explanation: The table name or view name specified already exists in the catalog.

Processing continues.

User Response: Correct the SQL statement.

SQL4137W **The COLUMN DEFINITION is missing.**

Explanation: At least one COLUMN DEFINITION should be specified for a CREATE TABLE.

Processing continues.

User Response: Correct the SQL statement.

SQL4138W **The data type** *type1* **of the target is not compatible with the data type** *type2* **of the source.**

Explanation: Data types should match:
- In a FETCH statement, between source and target.
- In a SELECT statement, between source and target.

Processing continues.

User Response: Correct the SQL statement.

SQL4139I **You must specify a VIEW COLUMN LIST for** *schema-name.table*.

Explanation: If any two columns in the table specified by the QUERY SPECIFICATION have the same column name, or if any column of that table is an unnamed column, then a VIEW COLUMN LIST must be specified.

Processing continues.

User Response: Correct the SQL statement.

SQL4140W **An error has occurred causing the flagger to stop. Module name =** *module-name*. **Internal error code =** *error-code*.

Explanation: 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.

User Response: Record this message number (SQLCODE), module name and error code in the message. Contact your technical service representative with the information.

SQL4141W **An error has occurred when attempting to produce message** *message-number* **in module** *module-name*.

Explanation: The FLAGGER has attempted to produce an undefined message.

Processing continues.

User Response: 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*.

Explanation: 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.

User Response: 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*.

Explanation: 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.

User Response: Restart precompilation if flagging is required.

SQL4144W   Internal error on call to the FLAGGER found in module *module-name*. **Internal error code =** *error-code*.

Explanation: The FLAGGER encountered an internal error.

Preprocessing continues but FLAGGER operation is discontinued.

User Response: Record this message number (SQLCODE), module name and error code in the message. Contact your technical service representative with the information.

SQL4145W   Error during FLAGGER access to the system catalog. Flagging continues to check syntax only. SQLCODE = *nnn*   **SQLERRP =** *modname*   **SQLERRD =** *nnn*   **Creator =** *creatorname*   **Table =** *tablename*

Explanation: An internal error has occurred during FLAGGER access to the system catalog.

Processing continues but with flagger syntax checking only.

User Response: Record this message number (SQLCODE), module name and error code in the message. Contact your technical service representative with the information.

SQL4146W   Internal error occurred causing the semantics processing to stop. Module name = *module-name*. **Internal error code =** *error-code*.

Explanation: The FLAGGER has encountered a severe internal error in a semantics analysis routine.

Processing continues but with flagger syntax checking only.

User Response: Record this message number (SQLCODE), module name and error code in the message. Contact your technical service representative with the information.

SQL4147W   Invalid flagger version number.

Explanation: 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.

User Response: Provide a valid flagger version number. Refer to the *Command Reference* manual.

SQL4170W **You must declare column** *column* **as NOT NULL.**

Explanation: All columns identified as UNIQUE must be defined with the NOT NULL option.

Processing continues.

User Response: Correct the SQL statement.

SQL4171W **The viewed table** *schema-name*.*table* **must be updatable.**

Explanation: When the WITH CHECK OPTION clause is specified then the viewed table must be updatable.

Processing continues.

User Response: Correct the SQL statement.

SQL4172W **Invalid number of column names.**

Explanation: The number of column names in the VIEW COLUMN LIST should be the same as the DEGREE of the table specified by the QUERY SPECIFICATION.

Processing continues.

User Response: Correct the SQL statement.

SQL4173W **You must declare cursor** *cursor* **before using it.**

Explanation: The cursor specified has not been declared in a DECLARE CURSOR statement.

Processing continues.

User Response: Correct the SQL statement.

SQL4174W **Cursor** *cursor* **is already declared.**

Explanation: Cursor specified has already been declared in a DECLARE CURSOR statement.

Processing continues.

User Response: Correct the SQL statement.

SQL4175W **Only * or column names are valid in this context.**

Explanation: When UNION is specified, the SELECT LIST for the specification of the two TABLE EXPRESSIONs identified by the QUERY EXPRESSION and QUERY TERM should consist of * or COLUMN REFERENCE(s).

Processing continues.

User Response: Correct the SQL statement.

SQL4176W **The descriptions of tables identified by the QUERY EXPRESSION beginning with** *schema-name1*.*table1* **and the QUERY TERM beginning with** *schema-name2*.*table2* **should be identical.**

Explanation: When UNION is specified, the descriptions of the two tables should be identical, except for the column names.

Processing continues.

User Response: Correct the SQL statement.

SQL4177W **The SORT SPECIFICATION** *number* **is outside the DEGREE of cursor** *cursor*.

Explanation: If the SORT SPECIFICATION contains an unsigned integer, it should be greater than 0 and not greater than the number of columns in the table.

Processing continues.

User Response: Correct the SQL statement.

SQL4178W **Table** *schema-name*.*table* **is a read-only table.**

Explanation: A DELETE, an INSERT, or an UPDATE was specified on a read-only table.

Processing continues.

User Response: Correct the SQL statement.

SQL4179W   Table *schema-name*.*table* **must not be identified in a FROM clause of any subquery contained in a SEARCH CONDITION.**

Explanation: A table specified in a DELETE or an UPDATE cannot be used in a FROM clause of any subquery contained in its SEARCH CONDITION.

Processing continues.

User Response: Correct the SQL statement.

SQL4180W   Table *schema-name1*.*table1* **is not the first table** *schema-name2*.*table2* **specified in the DECLARE CURSOR statement.**

Explanation: Table specified in a DELETE or an UPDATE statement should be the first table specified in the FROM clause of the DECLARE CURSOR statement.

Processing continues.

User Response: Correct the SQL statement.

SQL4181W   **The number of TARGET SPECIFICATIONs does not match the DEGREE of cursor** *cursor*.

Explanation: The number of TARGET SPECIFICATIONs in the FETCH statement does not match the DEGREE of the table specified.

Processing continues.

User Response: Correct the SQL statement.

SQL4182W   **Target table** *schema-name*.*table* **of the INSERT statement is also in the FROM clause or in the subquery.**

Explanation: The named table must not be identified in a FROM clause of a QUERY SPECIFICATION or of any subquery contained in the QUERY SPECIFICATION.

Processing continues.

User Response: Correct the SQL statement.

SQL4183W   **The number of columns specified does not match the number of values specified.**

Explanation: In an INSERT statement, the number of columns specified does not match the number of values specified.

Processing continues.

User Response: Correct the SQL statement.

SQL4184W   **The number of columns specified does not match the DEGREE of the QUERY SPECIFICATION beginning with table** *schema-name*.*table*.

Explanation: In an INSERT statement, the number of columns specified does not match the DEGREE of the table specified in the QUERY SPECIFICATION

Processing continues.

User Response: Correct the SQL statement.

SQL4185W   **There is a data type or length mismatch between column** *column* **and the INSERT or UPDATE item.**

Explanation: One of the following conditions has not been met:

- If the data type of the column name is character string, then the corresponding item in the INSERT or UPDATE statement should be character string of length equal to or less than the length of the column name.
- If the data type of the column name is exact numeric, then the corresponding item in the INSERT or UPDATE statement should be exact numeric.
- If the data type of the column name is approximate numeric, then the corresponding item in the INSERT or UPDATE statement should be approximate numeric or exact numeric.

Processing continues.

User Response: Correct the SQL statement.

SQL4186W **You cannot use a GROUP BY or HAVING clause or identify a GROUPed view in this context.**

Explanation: The table identified in the FROM clause of the TABLE EXPRESSION in a SELECT statement should not include a GROUP BY or HAVING clause and should not identify a GROUPed view.

Processing continues.

User Response: Correct the SQL statement.

SQL4187W **The number of elements specified in the SELECT LIST should match the number in the SELECT TARGET LIST.**

Explanation: In the SELECT statement, the number of elements specified in the SELECT LIST should match the number of elements in the SELECT TARGET LIST.

Processing continues.

User Response: Correct the SQL statement.

SQL4188W **A column function is not allowed in a SET clause of an UPDATE statement.**

Explanation: A VALUE EXPRESSION in a SET clause of an UPDATE statement should not include a column function.

Processing continues.

User Response: Correct the SQL statement.

SQL4189W **You cannot specify NULL for NOT NULL column** *column***.**

Explanation: If NULL is specified in the SET clause of the UPDATE statement, then the

corresponding column should allow nulls.

Processing continues.

User Response: Correct the SQL statement.

SQL4190W **A host variable of an unrecognized data type is referenced. Host variable position is** *position***.**

Explanation: The host variable reference at position *position* is of a data type not recognized by the standard.

Processing continues.

User Response: Correct the SQL statement.

SQL4191W **The data type of column** *column-name* **is not recognized.**

Explanation: The data type of the column is not recognized by the standard.

Processing continues.

User Response: Correct the SQL statement.

SQL4192W **Table** *schema-name.table* **is not found in the catalog.**

Explanation: The named table or view does not exist in the system catalog.

SQL4300 - SQL4399

SQL4300N **Java support is not installed or properly configured on this platform.**

Explanation: Support for Java stored procedures and user-defined functions is not installed and configured on this server.

User Response: Ensure that a compatible Java Runtime Environment or Java Development Kit for the server is installed. Ensure that the "JDK_PATH" configuration parameter is correctly set.

sqlcode: -4300

sqlstate: 42724

SQL4301N **Java interpreter startup or communication failed, reason code** *reason-code***.**

Explanation: An error occurred while attempting to start or communicate with a Java interpreter. The reason codes are:

1 Java environment variables or Java database configuration parameters are invalid.

2 A Java Native Interface call to the Java interpreter failed.

3 The "db2java.zip" file may be corrupt or missing.

4 The Java interpreter has terminated itself and cannot be restarted.

User Response: Ensure that the Java database configuration parameters (jdk_path and java_heap_sz) are correctly set. Ensure that a supported Java runtime environment is installed. Ensure that internal DB2 classes (COM.ibm.db2) are not overridden by user classes.

sqlcode: -4301

sqlstate: 58004

SQL4302N **Java stored procedure or user-defined function** *name***, specific name** *spec-name* **aborted with an exception** *string***.**

Explanation: The Java stored procedure or user-defined function aborted with a Java exception. The administration notification log contains a Java stack traceback for the aborted method.

User Response: Debug the Java method to eliminate the exception.

sqlcode: -4302

sqlstate: 38501

SQL4303N **Java stored procedure or user-defined function** *name***, specific name** *spec-name* **could not be identified from external name** *string***.**

Explanation: The CREATE PROCEDURE or CREATE FUNCTION statement that declared this stored procedure or user-defined function had a badly formatted EXTERNAL NAME clause. The external name must be formatted as follows: "package.subpackage.class!method".

User Response: Submit a corrected CREATE PROCEDURE or CREATE FUNCTION statement.

sqlcode: -4303

sqlstate: 42724

SQL4304N **Java stored procedure or user-defined function** *name***, specific name** *spec-name* **could not load Java class** *class***, reason code** *reason-code***.**

Explanation: The Java class given by the EXTERNAL NAME clause of a CREATE PROCEDURE or CREATE FUNCTION statement could not be loaded. The reason codes are:

1 The class was not found on the CLASSPATH.

2 The class did not implement the required interface

("COM.ibm.db2.app.StoredProc" or "COM.ibm.db2.app.UDF") or lacked the Java "public" access flag.

3   The default constructor failed or was unavailable.

4   Could not load driver for "jdbc:default:connection".

5   Could not establish default context.

User Response: Ensure that the compiled ".class" file is installed in the CLASSPATH, for example under "sqllib/function". Ensure it implements the required Java interfaces and is "public".

sqlcode: -4304

sqlstate: 42724

SQL4306N **Java stored procedure or user-defined function** *name***, specific name** *spec-name* **could not call Java method** *method***, signature** *string***.**

Explanation: The Java method given by the EXTERNAL NAME clause of a CREATE

PROCEDURE or CREATE FUNCTION statement could not be found. Its declared argument list may not match what the database expects, or it may not be a "public" instance method.

User Response: Ensure that a Java instance method exists with the "public" flag and the expected argument list for this call.

sqlcode: -4306

sqlstate: 42724

SQL4400 - SQL4499

SQL4400N *authorization-ID* **does not have the authority to perform the administration task at the DB2 Administration Server.**

Explanation: The user does not have the necessary authority at the DB2 Administration Server to perform the attempted administrative action.

User Response: Submit the request to the DB2 Administration Server using a user ID with DASADM authority. DASADM authority is required to perform any administrative actions against the DB2 Administration Server. The DASADM group is a DB2 Administration Server configuration parameter. Verify the current setting of the DASADM group by viewing the DB2 Administration Server configuration parameters using the GET ADMIN CONFIGURATION command. To modify the

value of any configuration parameter, use the UPDATE ADMIN CONFIGURATION command.

SQL4401C **The DB2 Administration Server encountered an error during startup.**

Explanation: An error was encountered during startup of the DB2 Administration Server.

User Response: Refer to the DB2 Administration Server's First Failure Data Capture Log for additional information. Take the appropriate action as necessary and attempt to restart the DB2 Administration Server.

If the problem persists contact your technical service representative.

SQL4402W The DB2ADMIN command was successful.

Explanation: All processing completed normally.

User Response: No further action is required.

SQL4403N The syntax of the command is invalid.

Explanation: The command was entered using invalid arguments or an invalid number of parameters.

User Response: Re-submit the command with valid argument(s).

SQL4404N The DB2 Administration Server does not exist.

Explanation: The DB2 Administration Server was not found on the machine.

User Response: Create the DB2 Administration Server on the machine.

- On Windows operating systems, issue the following command:

  db2admin create

- On UNIX platforms, ensure that you have root authority, and issue the following command from the DB2DIR/instance directory, where DB2DIR is the installation path of DB2. <ASName> represents the name of the Administration Server:

  dascrt <ASName>

SQL4405W The DB2 Administration Server already exists.

Explanation: The DB2 Administration Server already exists on the machine

User Response: No further action is required.

SQL4406W The DB2 Administration Server was started successfully.

Explanation: All processing completed normally.

User Response: No further action is required.

SQL4407W The DB2 Administration Server was stopped successfully.

Explanation: All processing completed normally.

User Response: No further action is required.

SQL4408N The DB2 Administration Server was not dropped because it is active.

Explanation: The DB2 Administration Server must be stopped before it can be dropped.

User Response: To stop the DB2 Administration Server, enter the following command:

  DB2ADMIN STOP

SQL4409W The DB2 Administration Server is already active.

Explanation: The DB2ADMIN START command cannot be processed because the DB2 Administration Server is already active.

User Response: No further action is required.

SQL4410W The DB2 Administration Server is not active.

Explanation: The DB2ADMIN STOP command cannot be processed because the DB2 Administration Server is not active.

User Response: No further action is required.

SQL4411N The requested operation is not allowed because the server instance is not the DB2 Administration Server.

Explanation: The requested operation is only valid when issued against the DB2 Administration Server.

User Response: The DB2 Administration Server should be set up using the DB2ADMIN command. Refer to the *Quick Beginnings* for

further information about the DB2ADMIN command.

SQL4412N   The logon user account for the DB2 Administration Server is invalid.

Explanation: To perform the requested task, the DB2 Administration Server must be running under a valid logon user account. This error is either caused by the account not being setup, or the logon user account does not contain a valid DB2 user ID.

User Response: If a logon user account has been set up, then ensure that the account uses a valid DB2 user ID.

You can set up the logon user account for the DB2 Administration Server by using the following command:

```
DB2ADMIN SETID <userid>
              <password>
```

SQL4413W   Usage: DB2ADMIN creates, drops, starts, or stops the DB2 Administration Server.

Explanation: The DB2ADMIN command syntax is:

```
DB2ADMIN CREATE [
            /USER:<username>
            /PASSWORD:<password>
            ]
        DROP
        START
        STOP [/FORCE]
        SETID <username>
              <password>
        SETSCHEDID <username> <password>
        /h
```

The command options are:

CREATE
        create the DB2 Administration Server

DROP   delete the DB2 Administration Server

START   start the DB2 Administration Server

STOP   stop the DB2 Administration Server

SETID   set the logon account for the DB2 Administration Server

SETSCHEDID
        set the logon account used by the scheduler to connect to the tools catalog database. Only required if the scheduler is enabled and the tools catalog database is remote to the DB2 Administration Server.

/USER   specify the logon account name during DB2ADMIN CREATE

/PASSWORD
        specify the logon account password during DB2ADMIN CREATE

/FORCE
        stop the DB2 Administration Server during DB2ADMIN STOP regardless of whether or not it is in the process of servicing any requests

/h   display the usage information

User Response: Issue the DB2ADMIN command with one of the valid command options.

SQL4414N   The DB2 Administration Server is not active.

Explanation: The request cannot be processed unless the DB2 Administration Server is active.

User Response: Start the DB2 Administration Server by issuing the command DB2ADMIN START, and reissue the request.

SQL4900 - SQL4999

SQL4901N Precompiler Services requires reinitialization because of a previous error.

Explanation: 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.

User Response: Call the sqlainit function to reinitialize Precompiler Services

---

SQL4902N At least one of the characters in parameter *n* of function *function* is not valid.

Explanation: The specified parameter in the specified function contains at least one character that is not valid.

The function cannot be completed.

User Response: Correct the specified parameter and call the function again.

SQL4903N The length of parameter *n* of function *name* is not valid.

Explanation: The length of the specified parameter in the specified function is not valid.

The function cannot be completed.

User Response: Correct the specified parameter and call the function again.

SQL4904N The pointer to parameter *n* of function *function* is not valid.

Explanation: The pointer to the specified parameter in the specified function is not valid.

The function cannot be completed.

User Response: Correct the specified parameter and call the function again.

SQL4905N The value of parameter *n* of function *function* is not within the valid range.

Explanation: 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.

User Response: Correct the specified parameter and call the function again.

SQL4906N The list of table space names specified is an incomplete set for the rollforward operation.

Explanation: The list of table space names is incomplete for one of the following reasons:

- For point-in-time table space recovery, a table space list must be specified.
- For point-in-time table space recovery, a self-contained list of table space names must be specified. The table spaces in the list must contain all objects of every table included in the table spaces.
- Point-in-time table space recovery is not allowed for the system catalogs.
- End-of-logs table space recovery is allowed for the system catalogs but it can be the only table space name in the list.
- The CANCEL option for rollforward must have a table space list if there are no table spaces in "rollforward-in-progress" state.

User Response: Check the table space list and resubmit the rollforward command with the complete table space list.

SQL4907W **Database** *name* **is recovered but one or more of the tables in the table space list included for the rollforward operation are placed in check pending states.**

Explanation: One or more of the tables involved in the point-in-time tablespace recovery have referential constraints with tables outside of the table space list used for recovery. All these tables are placed in check pending state. Rollforward operation is otherwise completed successfully.

User Response: Check the state for the tables in the table spaces and take appropriate actions if necessary.

SQL4908N **The table space list specified for roll-forward recovery on database** *name* **is invalid on node(s)** *node-list***.**

Explanation: If starting a new table space rollforward, one or more of the table spaces specified in the list to be rolled forward is not in rollforward pending state on the specified nodes. If continuing a table space rollforward that is already in progress, one or more of the table spaces specified in the list to be rolled forward is not in rollforward in progress state on the specified nodes.

If ″,...″ is displayed at the end of the node list, see the administration notification log for the complete list of nodes.

Roll-forward recovery stops.

(Note : if you are using a partitioned database server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: Use the LIST TABLESPACES SHOW DETAIL command on the nodes specified to find out which table spaces are not ready to be rolled forward. Use the QUERY STATUS option of the rollforward command to determine the status of the table space rollforward. If the rollforward status is ″TBS pending″, a new table space rollforward can be started. If the rollforward status is ″TBS working″, a table space

rollforward is already in progress.

If starting a new table space rollforward, put the tablespaces into rollforward pending state by restoring them.

If continuing a table space rollforward and one or more of the table spaces involved have been restored and put into rollforward pending state, the table space rollforward in progress must be canceled. Submit the rollforward command again with the CANCEL option and the same table space list. When the rollforward in progress is canceled, the table spaces will have been put into restore pending state. Restore the table spaces and submit the original rollforward command again.

SQL4909W **Roll forward recovery has completed successfully but one or more table spaces have their tables put in DRP/DRNP state. Check the administration notification log for details on node(s)** *node-list***.**

Explanation: One or more table spaces have tables in DATALINK Reconcile Pending (DRP) state or DATALINK Reconcile Not Possible (DRNP) state for one of the following reasons:

• Restore is done specifying the WITHOUT DATALINK option and rollforward to a point-in-time follows the restore. Any tables with DATALINK columns are placed in DRP state.

• Restoring from a backup image with a different database name, alias, hostname, or instance and rollforward follows the restore. Any tables with DATALINK columns are placed in DRNP state.

• Restoring from a backup image that has become unusable and rollforward follows the restore. Any tables with DATALINK columns are placed in DRNP state.

• Rollforward was to a point-in-time, not to the end of logs. Any tables with DATALINK columns in that tablespace are placed in DRP state.

- DATALINK column information does not exist at the DB2 Data Links Manager(s). Affected tables are placed in DRNP state.
- Rollforward involved DATALINK columns defined with the "RECOVERY NO" option. Affected tables are placed in DRP state.

If ",..." is displayed at the end of the node list, see the diagnostic log for the complete list of nodes.

(Note : if you are using a partitioned database server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: Look in the db2diag.log file to find out what tables are put in DRP/DRNP state. See the Administration Guide for information on reconciling tables that are in DRP/DRNP state.

SQL4910N   **Overflow log path** *log-path* **is not valid.**

Explanation: The overflow log path specified on the ROLLFORWARD command is not valid. The overflow log path must be a directory in a file system. This directory must be accessible by the instance owner id.

User Response: Resubmit the command with a valid overflow log path.

SQL4911N   **The host variable data type is not valid.**

Explanation: The data type of the host variable is not valid.

The function cannot be completed.

User Response: Correct the data type of the host variable and call the function again.

SQL4912N   **The host variable data length is out of range.**

Explanation: The length of the host variable is not valid.

The function cannot be completed.

User Response: Correct the length of the host variable and call the function again.

SQL4913N   **The host variable token ID has already been used.**

Explanation: 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.

User Response: Correct the token ID of the host variable and call the function again.

SQL4914N   **The host variable token ID is not valid.**

Explanation: The token ID of the host variable is not valid.

The function cannot be completed.

User Response: Correct the token ID of the host variable and call the function again.

SQL4915N   **The "sqlainit" function has already been called.**

Explanation: Precompiler Services has already been initialized.

User Response: No action is required. Continue processing.

SQL4916N   **The "sqlainit" function has not been called.**

Explanation: Precompiler Services must be initialized before the requested function call can be processed.

The function cannot be completed.

User Response: Issue an sqlainit function call to initialize Precompiler Services.

SQL4917N   **Element** *number* **in the option array is not valid.**

Explanation: 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.

User Response: Correct the values stored in the option array. Call the function again.

SQL4918N **The term_option parameter of function "sqlainit" is not valid.**

Explanation: The *term_option* parameter is not valid.

The function cannot be completed.

User Response: Correct the *term_option* parameter and call the function again.

SQL4919N **The task_array parameter of function "sqlacmpl" is too small.**

Explanation: The task array structure passed to Precompiler Services on the sqlacmpl function call is too short.

The function did not complete successfully.

User Response: 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.**

Explanation: The token id array structure passed to Precompiler Services on the sqlacmpl() function call is too small.

The function did not complete successfully.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Remove or add the clause to correct the statement.

SQL4941N **The SQL statement is blank or empty.**

Explanation: The text following EXEC SQL was blank or empty.

The function cannot be completed.

User Response: 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*.**

Explanation: 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. For a user-defined data type, the host variable might be defined with an associated built-in data type that is not compatible with the result type of the FROM SQL transform function defined in the transform group for the statement.

The function cannot be completed.

User Response: 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.

Explanation: The number of host variables specified in both the INTO clause and the SELECT clause must be the same.

The function is processed.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: Parameter markers can be used only in dynamic SQL statements.

The function cannot be completed.

User Response: Use host variables instead of parameter markers for static SQL statements.

SQL4946N   The cursor or statement name *name* **is not defined.**

Explanation: The cursor or statement name *name* specified in the statement is not defined.

The function cannot be completed.

User Response: Check the application program for completeness and possible spelling errors in the cursor or statement names.

SQL4947W   An INCLUDE SQLDA statement was encountered and ignored.

Explanation: The FORTRAN precompiler shipped by the database manager does not support the INCLUDE SQLDA statement.

The statement is ignored. Processing continues.

User Response: 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.

Explanation: Compound SQL statements containing user-defined SQLDAs are not supported in a 16-bit application.

User Response: 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.

Explanation: 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.

User Response: Correct the *sqlda_id* parameter in the application program.

SQL4952N The sqlvar_index parameter of function *name* is not valid.

Explanation: 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.

User Response: Correct the *sqlvar_index* parameter in the application program.

SQL4953N The call_type parameter of function *name* is not valid.

Explanation: The *call_type* parameter of the specified function in the application is not valid.

The function cannot be completed.

User Response: Correct the *call_type* parameter in the application program.

SQL4954N The section_number parameter of function *name* is not valid.

Explanation: The *section_number* parameter of the specified function in the application program is not valid. Note that for the following SQL statements, the section_number parameter of the function sqlacall() is used to pass in the statement type:

- CONNECT
- SET CONNECTION
- RELEASE
- DISCONNECT

The function cannot be completed.

User Response: Correct the *section_number* parameter in the application program.

SQL4970N Roll-forward recovery on database *name* cannot reach the specified stop point (end-of-log or point-in-time) because of missing log file(s) on node(s) *node-list*.

Explanation: This message is returned in the following situations:

- a request was made to bring the specified database out of roll-forward pending state by specifying caller action SQLUM_ROLLFWD_STOP, SQLUM_STOP, SQLUM_ROLLFWD_COMPLETE, or SQLUM_COMPLETE but the Rollforward Database utility cannot find the necessary archive log file(s) in the database log directory or the overflow log directory on the specified node(s) to reach the stopping point from the previous ROLLFORWARD DATABASE command.

- in a multi-node environment, if the Rollforward Database utility cannot find the necessary archive log file(s) to bring a node into synchronization with the catalog node.

If ",..." is displayed at the end of the node list, see the administration notification log for the complete list of nodes.

Roll-forward recovery has stopped. The database is left in roll-forward pending state.

Note : if you are using a partitioned database server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.

User Response: Use the ROLLFORWARD DATABASE command with the QUERY STATUS option to determine which log files are missing.

Do one of the following:

- Make sure that all archive log files are available in the database log directory or the overflow log path, and issue the ROLLFORWARD DATABASE command again.

- If you cannot find the missing log files, restore the database/table space on all nodes, then do point-in-time recovery using a timestamp that is earlier than that of the earliest missing log file.

SQL4971N **Roll-forward recovery on database** *name* **on node** *node-number* **failed previously while stopping. Roll-forward recovery must be stopped.**

Explanation: A request was made to continue rolling forward the specified database by specifying caller action SQLUM_ROLLFWD. The previous iteration of roll-forward recovery failed while stopping. If rolling forward at the database level, this means that it failed during log truncation. Roll-forward recovery for this database must now be stopped by specifying caller action SQLUM_ROLLFWD_STOP, SQLUM_STOP, SQLUM_ROLLFWD_COMPLETE, or SQLUM_COMPLETE.

Note : if you are using a partitioned database server, the node number indicates which node the error occurred on. Otherwise, it is not pertinent and should be ignored.

User Response: Reissue the ROLLFORWARD DATABASE command with caller action SQLUM_ROLLFWD_STOP, SQLUM_STOP, SQLUM_ROLLFWD_COMPLETE, or SQLUM_COMPLETE. Any stoptime specified will be ignored since the previous stoptime is already being processed.

SQL4972N **Log extent** *extent* **on node** *node-number* **could not be moved to the database logpath.**

Explanation: The Rollforward utility was invoked with the STOP option. As part of roll-forward processing, the log extent *extent* has to be truncated. This extent must exist in the database logpath. Currently, the extent exists in the overflow log path. An attempt was made to move the extent from the overflow log path to the database logpath. The attempt failed. Roll-forward processing has been stopped.

Note : if you are using a partitioned database server, the node number indicates which node the error occurred on. Otherwise, it is not pertinent and should be ignored.

User Response: Move the extent from the overflow log path to the database logpath and then resubmit the ROLLFORWARD DATABASE command.

SQL4973N **Forward recovery on database** *name* **cannot complete because the log information on node(s)** *node-list* **does not match the corresponding record(s) on the catalog node.**

Explanation: The Rollforward utility processed all the log files found on each node, but the stop points on the specified nodes do not match the corresponding records on the catalog node. This can be caused by missing log files on either the catalog node or on the specified nodes, or the catalog node must be included in the list of nodes to be rolled forward.

The ROLLFORWARD DATABASE processing stops.

User Response: Do one of the following:
- Check if the catalog node needs to be rolled forward. If it does, submit the ROLLFORWARD DATABASE command again and include the catalog node.
- Use the ROLLFORWARD DATABASE command with the QUERY STATUS option to determine which log files are missing. When you find the log files, put them in the log path or overflow log path and resume forward recovery.
- If you cannot find the missing log files, restore the database on all nodes, then do point-in-time recovery using a stoptime that is earlier than that of the earliest missing log file.

SQL4974W **The ROLLFORWARD DATABASE QUERY STATUS command encountered sqlcode** *sqlcode*.

Explanation: The ROLLFORWARD DATABASE QUERY STATUS command encountered an error

with the sqlcode *sqlcode*. The query might not be successful on some nodes for various reasons. The most severe error is indicated by *sqlcode*. The roll-forward status is only returned for the successful nodes.

User Response: Look up the sqlcode *sqlcode* in the *Message Reference*, or online to determine the problems on the failed nodes. Take the required corrective actions, then continue the forward recovery on these nodes.

SQL4975W   **Roll-forward operation was cancelled successfully. The database or selected table spaces have to be restored on node(s) *node-list*.**

Explanation: A roll-forward operation was cancelled before it successfully completed, and the database or selected table spaces are left in an inconsistent state. The database or selected table spaces are in restore pending state on the listed nodes.

If "*,...*" is displayed at the end of the node list, see the administration notification log for the complete list of nodes.

Note : if you are using a partitioned database server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.

User Response: Restore the database or selected table spaces on the listed nodes. The table spaces that are in restore pending state can be identified on the specified node(s) by the LIST TABLESPACES command or the db2dart utility.

SQL4976N   **The ROLLFORWARD DATABASE command cannot be submitted on a non-catalog node.**

Explanation: The ROLLFORWARD DATABASE command is only accepted on the catalog node.

User Response: Submit the command on the catalog node.

SQL4977N   **Dropped table export directory *directory* is not valid.**

Explanation: The export directory path specified on the ROLLFORWARD command is not valid. The export directory path must be a directory in a file system. This directory must be accessible by the instance owner id.

User Response: Resubmit the command with a valid export directory path.

SQL4978N   **The dropped table can not be accessed.**

Explanation: The dropped table can not be accessed. This may be due to the table being put into unavailable state because of a LOAD without copy or a NOT LOGGED INITIALLY operation.

User Response: The table can not be recovered using the DROPPED TABLE RECOVERY option.

SQL4979W   **Unable to export the dropped table data.**

Explanation: The command was unable to export the data of the dropped table whose recovery is being attempted. This can happen when the dropped table id specified on the ROLLFORWARD command is invalid, or when not all of the logs are available for roll forward. This warning will be generated when an error occurs during a dropped table recovery using a ROLLFORWARD ... AND STOP command.

User Response: Resubmit the command after ensuring that the dropped table id provided is valid, and that all of the logs are available for roll forward.

SQL4994N   **Precompilation was ended because of a user interrupt request.**

Explanation: Precompilation was terminated because of an interrupt. The user may have pressed an interrupt key sequence.

Processing is terminated. No package was created.

User Response: Resubmit the precompilation if needed.

SQL4997N **The authorization ID is not valid.**

Explanation: 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.

User Response: Retry the application with a valid authorization ID.

SQL4998C **The application state is in error; the database connection has been lost.**

Explanation: The connection to the database has been lost.

The function cannot be processed.

User Response: Reconnect to the database.

SQL4999N **A Precompiler Services or Run-Time Services error occurred.**

Explanation: 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.

User Response: Record the message number (SQLCODE) and all error information from the SQLCA if possible.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt.
- Environment: Outer Precompiler Using Precompiler Services API
- Required information:
  – Problem description
  – SQLCODE
  – SQLCA contents if possible
  – Trace file if possible.

SQL5000 - SQL5099

SQL5001N *authorization-ID* **does not have the authority to change the database manager configuration file.**

Explanation: The user attempted to Update or Reset the database manager configuration file without having SYSADM authority.

The requested change is not made.

User Response: 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.**

Explanation: A system error, probably an I/O error, was encountered while accessing a configuration file.

The command cannot be processed.

User Response: Resubmit the command.

If the error persists, check the db2diag.log file for details and ensure that the configuration file is accessible. If unable to resolve the problem, contact an IBM Service Representative for assistance.

SQL5010N **The path name to the database manager configuration file is not valid.**

Explanation: 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.

User Response: Resubmit the command. If the error continues, reinstall the database manager.

SQL5012N   **Host variable** *host-variable* **is not an exact numeric data type.**

Explanation: A host variable *host-variable* was specified, but it is not valid in the context in which it was used. Host variable *host-variable* was specified as part of ABSOLUTE or RELATIVE in a FETCH statement, or in a ROWS clause of a FETCH or INSERT statement. The host variable was not usable for one of the following reasons:

• The host variable is not an exact numeric data type. Decimal data types with a scale of zero and integer data types are exact numeric data types.

• The host variable is a decimal data type, but the scale is not zero. Decimal data types must have zero decimal digits to have a scale of zero.

The statement cannot be processed.

User Response: Change the host variable to be an exact numeric data type.

sqlcode: -5012

sqlstate: 42618

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.**

Explanation: The value for the maximum number of remote connections to the workstation must be between 1 and 255.

The requested change is not made.

User Response: 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.**

Explanation: 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.

User Response: 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).**

Explanation: 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.

User Response: 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).**

Explanation: 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.

User Response: Repeat the update request, entering one of the possible values.

SQL5025C   The database manager configuration file is not current.

Explanation: 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.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: The release number in the database manager configuration file or database configuration file is not valid.

The command cannot be processed.

User Response: 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.

Explanation: 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.

User Response: 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

SQL5040N   One of the socket addresses required by the TCP/IP server support is being used by another process.

Explanation: One of the socket addresses required by the server either is in use by another program, or has not been freed completely by the TCP/IP subsystem after the database manager has been stopped.

User Response: If you have just issued db2stop, wait for a couple of minutes so that the TCP/IP subsystem has enough time to clean up its resources. Otherwise, make sure no other programs on your workstation are using the port numbers reserved for your service name in the /etc/services file. The port number is a component of a socket address.

SQL5042N   One of the communication protocol server support processes failed to start up.

Explanation: Due to system call failure, or a communication subsystem call failure, one of the communication protocol server support processes did not start up successfully.

User Response: You can investigate the problem by either:

- examine the system log records;
- turning on trace, run db2start again, then examine the trace record.

SQL5043N **Support for one or more communications protocols failed to start successfully. However, core database manager functionality started successfully.**

Explanation: 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.

User Response: The database manager attempts to start all communication protocols specified by the DB2COMM environment variable.

Check the administration notification log for details on this error. The log will contain more information 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.**

Explanation: There is not enough memory available to execute this function.

The function cannot be completed.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Reinstall the database manager.

SQL5051N **An object qualified by** *qualifier* **cannot be created in schema** *schema-name*.

Explanation: An object created in a CREATE SCHEMA statement is qualified by *qualifier* which is different from the schema name. All objects created in a CREATE SCHEMA statement must be either qualified by the schema name *schema-name* or not qualified. Unqualified objects are implicitly qualified by the schema name.

The statement cannot be processed.

User Response: Explicitly qualify the object in the schema with *schema-name* or remove *qualifier* from the object name.

sqlcode: -5051

sqlstate: 42875

SQL5055C The content of the database configuration file is not valid.

Explanation: 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.

User Response: Recreate the database or restore it from a backup version.

sqlcode: -5055

sqlstate: 58031

SQL5060N The configuration parameter token specified is invalid.

Explanation: 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.

User Response: Choose a valid token number from those specified in the description of the Configuration Services API in the *Application Development Guide*. Recode the call to the API, and rerun the program.

SQL5061N An invalid pointer to structure sqlfupd was passed to Configuration Services.

Explanation: 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 *Application Development Guide* for more information.

User Response: 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.

Explanation: 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 *Application Development Guide* for more information.

User Response: Correct the code which invokes Configuration Services, and rerun the program.

SQL5065C The nodetype value in database manager configuration file is not valid.

Explanation: The *nodetype* parameter in the database manager configuration file is not valid.

The command cannot be processed.

User Response: Reinstall the database manager.

SQL5066W The database configuration parameter value for token *token-name* has been truncated.

Explanation: The database configuration parameter value is larger than the specified token can contain.

A new token now represents this database configuration parameter value and should be used if the value is larger than can be contained by the old token.

User Response: Use the new token for this database configuration parameter.

SQL5070N **The count parameter in the configuration command is not valid. It must be greater than 0.**

Explanation: The count value passed as a parameter to a Configuration Services API must be greater than 0.

The command cannot be processed.

User Response: Correct the code which invokes Configuration Services, and rerun the program.

SQL5075N **The configuration utility was interrupted.**

Explanation: 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.

User Response: Resubmit the command.

SQL5076W **The update completed successfully. The current value of NOTIFYLEVEL will cause some health monitor notifications not to be issued to the notification log.**

Explanation: Health Monitor issues notifications to the notification log and to specified e-mail and page contacts. The current value of NOTIFYLEVEL is set too low to have notifications be issued for alarms and warnings. NOTIFYLEVEL must be set to 2 or higher for alarm notifications and 3 or higher for warning notifications.

User Response: Increase the value of the database manager configuration parameter NOTIFYLEVEL.

SQL5077N **The parameter** *parameter* **is not supported by the Configuration Advisor for this server release. Supported parameters include** *supported-parameters***.**

Explanation: The parameter specified is not supported by this server release.

User Response: Reissue the command without the parameter or replace it with another supported parameter.

SQL5081N **The entry in the database configuration file for the size of the buffer pool (buffpage) is not in the valid range.**

Explanation: 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 (204000 for Extended Server Edition) (4KB pages). Under HP-UX, the value must be between 16 and 150000 (4KB pages).

The requested change is not made.

User Response: 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.**

Explanation: The value for the initial log file size must be between 12 and $(2**32 - 1)$.

The requested change is not made.

User Response: 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.**

Explanation: The value for the size of one log file extension must be between 4 and 256.

The requested change is not made.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The value for the size of the agent heap must be between 2 and 85.

The requested change is not made.

User Response: Resubmit the command with a valid value for the agent heap size.

SQL5099N   **The path** *path-name* **indicated by the database configuration parameter** *parameter* **is not valid, reason code** *reason-code***.**

Explanation: The path is not valid for one of the following reasons:

1	The path string is longer than 242 bytes.
2	The path does not exist.
3	The first directory in the path has a name in the form SQLNNNNN, where NNNNN is a value from 00001 to 99999.
4	A file with the correct name was found in the specified path, but is not a log file for this database.
5	The path is currently being used by another database.
6	The path is currently being used by the same database for another purpose.
7	The device specified by the path is not big enough to hold the log files.
8	A raw device cannot be specified as a mirror log path or an overflow log path.
9	The path cannot be accessed.

The requested change is not made.

User Response: Resubmit the command with a valid value for the path.

sqlcode: -5099

sqlstate: 08004

SQL5100 - SQL5199

SQL5100N   **The entry in the database manager configuration file for the number of allowed concurrent databases is too large.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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).**

Explanation: 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.

User Response: Do one or both of the following:

* Increase the size of the buffer pool.
* Decrease the maximum number of active processes allowed.

SQL5112N **The value of the configuration parameter** *parameter* **must be either 0 or 1.**

Explanation: The request is not completed because the value given for *parameter* is invalid.

User Response: Ensure that the value specified for *parameter* is either 0 or 1, then try the request again.

SQL5120N **Old and new log parameters cannot both be modified at the same time.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: Resubmit the command with a valid database options value.

SQL5122N **Access to the database was invalid because of a machine dependent check.**

Explanation: The database and database configuration file cannot be accessed because of copy protection.

The user request is denied.

User Response: 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.**

Explanation: An error occurred while accessing SQLOGCTL.LFH for the specified database.

The requested change is not made.

User Response: 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***.**

Explanation: 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:

1 Database server with local and remote clients

2 Client

3 Database server with local clients

4 Partitioned Database Server with local and remote clients

5 Satellite Database Server with local clients

The requested change is not made.

User Response: 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***.**

Explanation: The request is not done because the value of *parameter* is not within the valid range.

User Response: 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***.**

Explanation: The request is not done because the value of *parameter* is not within the valid range.

User Response: 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***.**

Explanation: The configuration parameter is not set or too long.

User Response: Change the value of the configuration parameter to within the maximum length given.

SQL5133N **The value** *value* **for configuration parameter** *parm* **is invalid. The valid set of values is** *value-list***.**

Explanation: *value* was the value specified for the configuration parameter *parm*. This value is not one of the allowable values shown in the *value_list*.

Consult the Application Development Guide (SQLFUPD entry) and the Administration Guide for the meanings of these values.

User Response: Change the value of the configuration parameter to one of the values shown in the valid list.

SQL5134N **The configuration parameter tpname contains invalid characters.**

Explanation: 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)

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: 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 and Windows) are:

1. the path must be a drive letter
2. the drive must exist

User Response: 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.**

Explanation: 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

User Response: 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, DCE, KERBEROS, SERVER_ENCRYPT, DCE_SERVER_ENCRYPT, or KRB_SERVER_ENCRYPT.**

Explanation: The allowed values for the configuration parameter "authentication" are:

- SERVER = 0
- CLIENT = 1
- DCE = 3
- SERVER_ENCRYPT = 4
- DCE_SERVER_ENCRYPT = 5
- KERBEROS = 7
- KRB_SERVER_ENCRYPT = 8

The requested change is not made.

User Response: 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.**

Explanation: The allowable range for *avg_appls* is from 1 to the value of *maxappls*.

The requested change is not made.

User Response: 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.

Explanation: 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.

User Response: Change the value of the configuration parameter to a value within the valid range.

SQL5150N The value specified for the configuration parameter *parameter* **is less than the minimum allowable value of** *minimum-value***.**

Explanation: The request is not completed because the value given for *parameter* is too low. *parameter* may not be less than *minimum value*

User Response: Ensure that the value specified for *parameter* is within the valid range, then try the request again.

SQL5151N The value specified for the configuration parameter *parameter* **is less than the minimum allowable value of** *minimum-value* **and is not -1.**

Explanation: The request is not completed because the value given for *parameter* is invalid. Except for an allowable value of -1, *parameter*

may not be less than *minimum value*.

User Response: Ensure that the value specified for *parameter* is within the valid range, then try the request again.

SQL5152N The value specified for the configuration parameter *parameter* **is greater than the maximum allowable value of** *maximum-value***.**

Explanation: The request is not completed because the value given for *parameter* is too high. *parameter* may not be greater than *maximum value*

User Response: Ensure that the value specified for *parameter* is within the valid range, then try the request again.

SQL5153N The update cannot be completed because the following relationship would be violated: *condition*

Explanation: A valid configuration file must maintain the following relationship:

condition.

The update request could not be completed because the resulting configuration would violate that relationship.

User Response: Resubmit the request, ensuring that the relationship is satisfied.

SQL5154N The requested combination of configuration values for "authentication" and *parameter* **is not allowed.**

Explanation: The database manager configuration parameter "authentication" must have the value "CLIENT" if the value of *parameter* is non-default.

The requested change is not made.

User Response: Resubmit the command with a valid combination of configuration parameter values.

SQL5155W **The update completed successfully. The current value of SORTHEAP may adversely affect performance.**

Explanation: The value of SORTHEAP is currently greater than half the value of the database manager configuration parameter SHEAPTHRES. This may cause performance to be less than optimal.

User Response: Increase the value of the database manager configuration parameter SHEAPTHRES and/or decrease the value of SORTHEAP so that SHEAPTHRES is at least twice as large as SORTHEAP.

A larger ratio is desirable in most cases. See the *Administration Guide* for recommendations on configuration parameter tuning.

SQL5156N **The value of the database manager configuration parameter "trust_allclnts" must be one of NO, YES, or DRDAONLY.**

Explanation: The allowed values for the configuration parameter "trust_allclnts" are:

- NO = 0
- YES = 1
- DRDAONLY = 2

The requested change is not made.

User Response: Resubmit the command with a valid value for "trust_allclnts".

SQL5180N **DB2 is unable to read the federation configuration file** *file-name***.**

Explanation: The federated configuration file could not be found, or it could not be opened for reading.

User Response: Specify the federation configuration file in the DB2_DJ_INI registry variable. Be sure that the file exists and is readable.

SQL5181N **The format of line** *line-number* **in the federation configuration file** *file-name* **is invalid.**

Explanation: The specified line does not have the proper format. The format should be: <variable-name> = <value>.

User Response: Put the specified line in the format described in the preceding paragraph.

SQL5182N **A required environment variable,** *variable-name***, has not been set.**

Explanation: In the federated configuration file, the environment variable *variable-name* is unlisted, or it is listed but has no value.

User Response: Consult the "Administration Guide" to find out what values can be assigned to the environment variable *variable-name*. Then use the db2set command to set this variable to the value that you want it to have.

SQL5185N **Pass-through to** *server-type* **data sources is not supported.**

Explanation: The Pass-Through facility cannot be used to access *server-type* data sources.

User Response: None required.

sqlcode: -5185

sqlstate: 428EV

SQL6000 - SQL6099

SQL6000N **DB2 Conversion of QMF Data.**

Explanation: This is the normal termination message.

User Response: No action is required.

SQL6001N   The prefix for the filename was not specified.

Explanation: The user executed a module of the SQLQMF facility directly rather than using the SQLQMF facility command.

The command cannot be processed.

User Response: Use the provided SQLQMF facility command.

SQL6002N   Communications Manager encountered a download error.

Explanation: Communications Manager encountered an error during the download of the host file.

The command cannot be processed.

User Response: Review the Communications Manager message log.

SQL6003N   The file exported from QMF contains rows that are too long. The row length is *number***.**

Explanation: The computed row size (the sum of the computed column sizes) exceeds the maximum of 7000 bytes.

The command cannot be processed.

User Response: 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   *function* returned the unexpected return code *code*.

Explanation: An unexpected error occurred during processing. The Communications Manager or DB2 may not be installed or configured correctly.

The command cannot be processed.

User Response: 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.

Explanation: 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.

User Response: 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.

Explanation: 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.

Federated system users: there isn't enough space to hold the db2djlink output. Additional space is required for the temporary files that db2djlink creates and uses.

The command cannot be processed.

User Response: Check for sufficient disk space on the C: drive. Retry the command.

Federated system users: increase the size of the file system where the federated server is installed. To estimate the amount of free space db2djlink needs on an AIX operating system, enter:

```
ls -e /
install_directory/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.**

Explanation: The indicated decimal field could not be converted.

The command cannot be processed.

User Response: 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.

Explanation: The file specified by the *filename* parameter is not in the expected QMF format.

The command cannot be processed.

User Response: 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.**

Explanation: The downloaded QMF file has a column width greater than 4000 bytes.

The command cannot be processed.

User Response: 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.

Explanation: The file being processed has more than 255 data columns.

The command cannot be processed.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: Too many parameters were specified with the command.

The command cannot be processed.

User Response: 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.

Explanation: 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.

User Response: Retry the command with a correct *host filename* syntax.

SQL6014N Invalid command syntax, colon (':') should follow a keyword.

Explanation: Keyword parameters with operands require the keyword to be immediately followed by a ":" character and then the operand.

The command cannot be processed.

User Response: Verify the command syntax and retry the command.

SQL6015N The keyword is not recognized.

Explanation: The keyword parameter indicator ("/") was followed by a value that is not a keyword.

The command cannot be processed.

User Response: Retry the command using a different keyword value.

SQL6016N Too many operands for the System/370 filename *name*.

Explanation: 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.

User Response: Retry the command using a different keyword value.

SQL6017N More information may be contained in the Import message log *name*.

Explanation: The database IMPORT operation terminated with warning or error messages.

The command has left its work files allocated.

User Response: 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.

Explanation: The S/370 filename is a required parameter.

The command cannot be processed.

User Response: Retry the command with a host filename.

SQL6019N Communications short session ID *ID* is either too long or is invalid.

Explanation: The value provided for the communications short session ID is either longer than one byte or is not alphabetic.

The command cannot be processed.

User Response: Retry the command with a valid value.

SQL6020N An import option was specified without also providing a database name.

Explanation: No database name was provided and an import option was specified.

The command has terminated.

User Response: Retry the command and include a database name.

SQL6021N   Data was successfully imported.

Explanation: This is a normal ending message when the SQLQMF facility imports data to a database.

User Response: No action is required.

SQL6022N   System database directory is not shared by all nodes.

Explanation: All nodes should be accessing one physical copy of the system database directory.

The command cannot be processed.

User Response: Ensure that all nodes are accessing the system database directory that resides in the sqllib directory, then try the request again.

SQL6023N   The user does not have the authority to run the Get Table Partitioning Information utility on table *name*.

Explanation: The user attempted to retrieve partitioning information on the specified table without the appropriate authorization (either the SYSADM or DBADM authority or the CONTROL or SELECT privilege on the table).

The utility stops processing.

User Response: Do not attempt to call the Get Table Partitioning Information utility without appropriate authorization. Contact your system administrator for assistance.

SQL6024N   Table or index *name* **is not defined on node** *node-number*.

Explanation: The application is connected to node *node-number*, and the table or index *name* is not defined.

The cause is one of the following:

- The node to which the application is connected is not a member of the nodegroup in which the table or index was created.
- The node is not in use by that nodegroup.

The utility stops processing.

User Response: Connect the application to a node on which the table or index is defined. Determine the nodegroup in which the table was created and obtain the list of nodes in the nodegroup by selecting the appropriate rows from the NODEGROUPDEF catalog view. The nodes whose IN_USE value is set to Y are the ones on which the table or index is defined.

SQL6025N   Database backed up on node *node1* **cannot be restored to node** *node2*.

Explanation: The backup image used for the restore is a backup of the database on a different node.

User Response: Ensure that you have the correct backup image for the node, then try the request again.

SQL6026N   Database with catalog node *node1* **cannot be restored to a database with catalog node** *node2*.

Explanation: The catalog node can exist on only one node and there is a discrepancy between the backup image and the node being restored to. This can occur in the following cases:

- The backup image specified catalog node *node1* and the restore was attempted on an existing database whose catalog node is node *node2*.
- The restore was attempted to a new database and the catalog node has not been restored first. (Restore the catalog node first to create the database on all nodes).

User Response: Verify that the correct backup image is being restored.

If you are restoring to an existing database and want to change the catalog node to *node2*, the existing database must be dropped first.

If you are restoring to a new database, restore the catalog node *node1* first.

SQL6027N   **The path** *path* **for the database directory is not valid.**

Explanation: The path *path* specified for either the CREATE DATABASE or the CATALOG DATABASE command starts with the character '.' or contains the character string '/.'.

The command cannot be processed.

User Response: Ensure that the specified path is a fully qualified path, does not start with the character '.', and does not contain the character string '/.'. Then try the request again.

SQL6028N   **Catalog database failed because database** *dbname* **was not found in the local database directory.**

Explanation: When cataloguing a local database into the system database directory, the command/API has to be issued from a node on the server where the database resides.

User Response: Reissue the command/API from a node where the database resides.

SQL6030N   **START or STOP DATABASE MANAGER failed. Reason code** *reason-code***.**

Explanation: The reason code indicates the error. The statement cannot be processed.

(1)   Cannot access the sqllib directory of the instance.

(2)   The full path name added to the profile filename is too long.

(3)   Cannot open the profile file.

(4)   The nodenum parameter value is not defined in the db2nodes.cfg file in the sqllib directory.

(5)   The nodenum parameter must be specified when a command option is specified.

(6)   The port parameter value is not valid.

(7)   The new couple hostname/port is not unique.

(8)   The FORCE option cannot be specified when the NODENUM option is specified.

(9)   The hostname and port parameters must be specified when using the ADDNODE option.

(10)   Cannot update the db2nodes.cfg file in the sqllib directory for the ADDNODE or RESTART option.

(11)   The hostname parameter value is not valid.

(12)   The pointer to the sqledbstrtopt or sqledbstopopt structure is not valid.

(13)   No port value is defined for your DB2 instance id (/etc/services file on UNIX-based systems).

(14)   The port value is not in the valid port range defined for your DB2 instance id (/etc/services file on UNIX-based systems).

(15)   A hostname value has no corresponding port 0 defined in the db2nodes.cfg file in the sqllib directory.

(16)   The value specified for the command or option parameter is not valid.

(17)   The DROP option cannot be specified when the NODENUM option is not specified.

(18)   The value specified for the callerac parameter is not valid.

(19)   Unable to create the UNIX socket directory /tmp/db2_<ver>_<rel> /$DB2INSTANCE.

(20)   The node number specified with the ADDNODE option already exists in the db2nodes.cfg file or the node has been already been added since the last stop database manager command was issued.

(21)   The tablespace type specified with the ADDNODE option is not valid.

(22)   The tablespace node specified with the ADDNODE option is out of range.

(23) The computer name parameter must be specified for the ADDNODE option.

(24) The user name parameter must be specified for the ADDNODE option.

(25) The computer name is not valid.

(26) The user name is not valid.

(27) The password is not valid.

(28) The password is expired.

(29) The user account specified is either disabled, expired, or restricted.

User Response: The action corresponding to the reason code is:

(1) Ensure that the $DB2INSTANCE userid has the required permissions to access the sqllib directory of the instance.

(2) Change the profile name to a shorter name so that the total length of the fully qualified path added to the profile name length is smaller than the SQL_PROFILE_SZ defined in the file sqlenv.h.

(3) Ensure that the profile file exists.

(4) Ensure that the nodenum value specified is defined in the db2nodes.cfg file in the sqllib directory and is between 0 and 999.

(5) Resubmit the command with the nodenum parameter specified.

(6) Ensure that the port value is between 0 and 999. If a value is not specified, the port value defaults to 0.

(7) Ensure that the new couple hostname/port is not already defined in the db2nodes.cfg file in the sqllib directory.

(8) Do not specify the FORCE option when you specify the NODENUM option.

(9) Ensure that the hostname and the port values are specified when you specify the ADDNODE option.

(10) Ensure that the $DB2INSTANCE

username has write access to the sqllib directory of the instance, that there is sufficient disk space, and the file exists.

(11) Ensure that the specified hostname is defined on the system.

(12) Ensure that the pointer is not NULL and points to the sqledbstrtopt for the sqlepstr() API, or to the sqledbstopopt structure for the sqlepstp() API.

(13) Ensure that the services file (/etc/services on UNIX-based systems) contains an entry for your DB2 instance id.

(14) Ensure that you only use port values that are specified in the services file (/etc/services file on UNIX-based systems) for your instance.

(15) Ensure that all the hostname values have a port 0 defined in the db2nodes.cfg file in the sqllib directory including the restart option parameters.

(16) Ensure that the value specified for the option parameter is within the valid range.

(17) Specify the NODENUM option when you specify the DROP option.

(18) Ensure that the value specified for the callerac parameter is within the valid range.

(19) Check the permission of the /tmp filesystem to make sure all intermediate directories of /tmp/db2_<ver>_<rel>/ $DB2INSTANCE can be created.

(20) Ensure that the correct node number is being specified. You must stop the database manager to update the db2nodes.cfg file with nodes that have been added to the system since the previous stop database manager command.

(21) Ensure that the value specified for the tablespace type is within the valid range.

(22) Ensure that the tablespace node value

specified is defined in the db2nodes.cfg and is between 0 and 999.

(23) Specify the computer name of the system on which the new node is created using the COMPUTER option.

(24) Specify a valid domain account user name and password for the new node using the USER and PASSWORD options.

(25) Resubmit the command with a valid computer name.

(26) Resubmit the command with a valid user name.

(27) Resubmit the command with a valid password.

(28) Change/update the account password and resubmit the command.

(29) Resubmit the command with a valid user account.

SQL6031N **Error in the db2nodes.cfg file at line number** *line*. **Reason code** *reason-code*.

Explanation: The statement cannot be processed because of a problem with the db2nodes.cfg file, as indicated by the following reason codes:

(1) Cannot access the sqllib directory of the instance.

(2) The full path name added to the db2nodes.cfg filename is too long.

(3) Cannot open the db2nodes.cfg file in the sqllib directory.

(4) A syntax error exists at line *line* of the db2nodes.cfg file in the sqllib directory.

(5) The nodenum value at line *line* of the db2nodes.cfg file in the sqllib directory is not valid.

(6) The nodenum value at line *line* of the db2nodes.cfg file in the sqllib directory is out of sequence.

(7) The nodenum value at line *line* of the db2nodes.cfg file in the sqllib directory is not unique.

(8) The port value at line *line* of the db2nodes.cfg file in the sqllib directory is not valid.

(9) The hostname/port couple at line *line* of the db2nodes.cfg file in the sqllib directory is not unique.

(10) The hostname at line *line* of the db2nodes.cfg file in the sqllib directory is not valid.

(11) The port value at line *line* of the db2nodes.cfg file in the sqllib directory is not defined for your DB2 instance id in the services file (/etc/services on UNIX-based systems).

(12) The port value at line *line* of the db2nodes.cfg file in the sqllib directory is not in the valid port range defined for your DB2 instance id in the services file (/etc/services on UNIX-based systems).

(13) The hostname value at line *line* of the db2nodes.cfg file in the sqllib directory has no corresponding port 0.

(14) A db2nodes.cfg file with more than one entry exists, but the database manager configuration is not MPP.

(15) The netname at line *line* of the db2nodes.cfg file in the sqllib directory is not valid.

User Response: The action corresponding to the reason code is:

(1) Ensure that the $DB2INSTANCE userid has the required permissions to access the sqllib directory of the instance.

(2) Make the instance home directory path name shorter.

(3) Ensure that the db2nodes.cfg file exists in the sqllib directory and is not empty.

(4) Ensure that at least 2 values are defined per line in the db2nodes.cfg file and that the file does not contain blank lines.

(5) Ensure that the nodenum value defined in the db2nodes.cfg file is between 0 and 999.

(6) Ensure that all the nodenum values defined in the db2nodes.cfg file are in ascending order.

(7) Ensure that each nodenum value defined in the db2nodes.cfg file is unique.

(8) Ensure that the port value is between 0 and 999.

(9) Ensure that the new couple hostname/port is not already defined in the db2nodes.cfg file.

(10) Ensure the hostname value defined in db2nodes.cfg at line *line* is both defined on the system and operational.

(11) Ensure that the services file (/etc/services on UNIX-based systems) contains an entry for your DB2 instance id.

(12) Ensure that you only use port values that are specified in the services file (/etc/services file on UNIX-based systems) for your instance.

(13) Ensure that the port value 0 has been defined for the corresponding hostname in the db2nodes.cfg file.

(14) Perform one of the following actions:
- Remove the db2nodes.cfg file.
- Alter the db2nodes.cfg file to contain exactly one entry.
- Install Enterprise Server Edition.

(15) Ensure the netname value defined in db2nodes.cfg at line *line* is both defined on the system and operational.

SQL6032W **Start command processing was attempted on** *total-number* **node(s).** *number-started* **node(s) were successfully started.** *number-already-started* **node(s) were already started.** *number-not-started* **node(s) could not be started.**

Explanation: The database manager did not start successfully on all the nodes. Not all of the data in the database may be accessible. Data on the nodes that started successfully or were already running is accessible.

User Response: Check the log file created in the log directory of the sqllib directory for the instance to see which nodes did not start.

SQL6033W **Stop command processing was attempted on** *total-number* **node(s).** *number-stopped* **node(s) were successfully stopped.** *number-already-stopped* **node(s) were already stopped.** *number-not-stopped* **node(s) could not be stopped.**

Explanation: The database manager did not stop successfully on all the nodes. The database manager is still active on the nodes that could not be stopped.

User Response: Check the log file created in the log directory of the sqllib directory for the instance to see which nodes did not stop.

SQL6034W **Node** *node* **is not being used by any databases.**

Explanation: During DROP NODE VERIFY processing, all databases are scanned to verify that this node does not exist in the nodegroups of any database, and that no event monitor is defined on this node.

User Response: This node can now be removed from the system by issuing the 'db2stop drop nodenum <node>' command.

SQL6035W   Node *node* **is being used by database** *database***.**

Explanation: During DROP NODE VERIFY processing, databases are scanned to verify that this node does not exist in the nodegroups of any database, and that no event monitor is defined on this node. Node *node* is being used by database *database* and cannot be dropped.

User Response: You must do the following before dropping the node:

1. Redistribute the data to remove it from the node using the REDISTRIBUTE NODEGROUP command. Use either the DROP NODE option of the REDISTRIBUTE NODEGROUP command or the ALTER NODEGROUP statement to drop the node from the nodegroup. This must be done for all nodegroups in which the node to be dropped is a member.

2. Drop any event monitors that are defined on the node.

3. Issue the 'db2stop drop nodenum <node>' command to drop the node.

SQL6036N   START or STOP DATABASE MANAGER command is already in progress.

Explanation: A START DATABASE MANAGER or a STOP DATABASE MANAGER command is already in progress on the system.

User Response: Wait for the command in progress to complete, then try the request again.

SQL6037N   START or STOP DATABASE MANAGER timeout value was reached.

Explanation: The start_stop_time value defined in the database manager configuration was reached at the node. This value specifies the time, in minutes, in which a node must respond to a Start Database Manager, Stop Database Manager or Add Node command.

User Response: Do the following:

- Check the administration notification log to see if an error message has been logged for the node which timed out. If an error has not been logged, and timeouts persist, you may need to increase the start_stop_time value specified in the database manager configuration file.
- If the timeout occurred during a Start Database Manager command, you should issue a Stop Database Manager command for all nodes which timed out.
- If the timeout occurred during a Stop Database Manager command, you should issue a Stop Database Manager command for all nodes which timed out or for all nodes. Any node which is already stopped will return with a message stating that the node is stopped.

SQL6038N   No partitioning key is defined.

Explanation: The user attempted to use the Get Row Partitioning Information utility without specifying a partitioning key.

The utility stops processing.

User Response: Ensure that you specify a partitioning key, then try the request again.

SQL6039N   Partitioning column *column-number* **is currently defined as not nullable.**

Explanation: An attempt was made to assign a null value to partitioning column *column-number*, which is not nullable.

The utility stops processing.

User Response: Assign a non-null value or change the type of the partitioning column to be nullable.

SQL6040C   No FCM buffers are available.

Explanation: No FCM buffer is available.

The statement cannot be processed.

User Response: Try the request again.

If the error persists, increase the number of FCM buffers (*fcm_num_buffers*) specified in the

database manager configuration file, then try the request again.

sqlcode: -6040

sqlstate: 57011

SQL6041C No FCM connection entries are available.

Explanation: No FCM connection entry is available. FCM is unable to increase the number of connection entries automatically because the maximum value has been reached.

The statement cannot be processed.

User Response: Try the request again after other processes have freed up some of this resource.

sqlcode: -6041

sqlstate: 57011

SQL6042C No FCM message anchors are available.

Explanation: No FCM message anchor is available. FCM is unable to increase the number of message anchors automatically because the maximum value has been reached.

The statement cannot be processed.

User Response: Try the request again after other processes have freed up some of this resource.

sqlcode: -6042

sqlstate: 57011

SQL6043C No FCM request blocks are available.

Explanation: No FCM request block is available. FCM is unable to increase the number of request blocks automatically because the maximum value has been reached.

The statement cannot be processed.

User Response: Try the request again after

other processes have freed up some of this resource.

sqlcode: -6043

sqlstate: 57011

SQL6044N The syntax of the string representation *string* with value of datatype *datatype-value* and length *length* is not correct.

Explanation: The specified string cannot be recognized as the target datatype. (The *Application Development Guide* contains information on datatypes.) Either the syntax is not valid or the value is out of range.

The utility stops processing.

User Response: Ensure that the string representation or the datatype is correct, then try the request again.

SQL6045N The datatype *datatype-value* of length *datatype-length* is not supported.

Explanation: The datatype and data length are not supported for a partitioning key.

The utility stops processing.

User Response: See the *Administration Guide* for information about datatypes. See the *API Reference* for information about the Get Row Partitioning Information API.

SQL6046N Specified DROP NODE action is not valid.

Explanation: The value specified for the action parameter of the DROP NODE command is not valid. Only the verify mode is supported for the DROP NODE command. The parameter must be set to the value SQL_DROPNODE_VERIFY.

The command cannot be processed.

User Response: Ensure that the action is set to SQL_DROPNODE_VERIFY, then try the request again.

SQL6047N Nodegroup cannot be redistributed because table *name* **does not have a partitioning key.**

Explanation: At least one table in the single-node nodegroup does not have a partitioning key. All tables in the single-node nodegroup must have a partitioning key before the nodegroup can be redistributed to a multiple-node nodegroup.

The operation was not performed.

User Response: Use the ALTER TABLE command to specify partitioning keys for tables that do not have one. Then try the request again.

SQL6048N A communication error occurred during START or STOP DATABASE MANAGER processing.

Explanation: A TCP/IP communication error occurred while the START or STOP DATABASE MANAGER command was trying to establish connection with all the nodes defined in the sqllib/db2nodes.cfg file.

User Response: Do the following:

- Ensure that the node has the proper authorization defined in the .rhosts or the host.equiv files.

- Ensure that the application is not using more than (500 + (1995 - 2 * total_number_of_nodes)) file descriptors at the same time.

- Ensure all the Enterprise Server Edition environment variables are defined in the profile file.

- Ensure the profile file is written in the Korn Shell script format.

- Ensure that all the host names defined in the db2nodes.cfg file in the sqllib directory are defined on the network and are running.

SQL6049N The log control file for database *name* **could not be found on node(s)** *node-list***.**

Explanation: The SQLOGCTL.LFH file for the database is not in the database directory on the specified node(s).

The database was not started.

If ",..." is displayed at the end of the node list, see the syslog file for the complete list of nodes.

User Response: Either restore the database from a backup on the specified node(s), or re-create the database.

SQL6050N An I/O error occurred while accessing the log control file for database *name* **on node(s)** *node-list***.**

Explanation: An error occurred while accessing the SQLOGCTL.LFH file for the database on the specified node(s).

The database cannot be used.

If ",..." is displayed at the end of the node list, see the syslog file for the complete list of nodes.

User Response: Either restore the database from a backup on the specified node(s), or re-create the database.

SQL6051N Database *name* **is not configured for roll-forward recovery on node(s)** *node-list***.**

Explanation: The specified database is not configured for roll-forward recovery on the specified node(s).

The database is not rolled forward on all nodes.

If ",..." is displayed at the end of the node list, see the syslog file for the complete list of nodes.

User Response: Ensure that recovery is required on the specified node(s), then restore the most recent backup version of the database on these nodes.

SQL6052N **Cannot roll forward database** *name* **because it is not in roll-forward pending state on node(s)** *node-list*.

Explanation: The specified database is not in the roll-forward pending state on the specified node(s). This may be because the database has not been restored, or was restored with the WITHOUT ROLLING FORWARD option, or roll-forward recovery is complete on these nodes.

The database is not rolled forward.

If "..." is displayed at the end of the node list, see the syslog file for the complete list of nodes.

User Response: Do the following:
1. Ensure that recovery is required on the specified node(s).
2. Restore a backup version of the database on these nodes.
3. Issue the ROLLFORWARD DATABASE command.

SQL6053N **Error exists in file** *file*. **Reason code =** *reason-code*.

Explanation: There is an error in the specified file as indicated by the reason code:

(1) The number of values in the partitioning map file is not 1 or 4,096.

(2) The number of values in the distribution file is not 4,096.

(3) The data in the distribution file is not in a valid format.

(4) The node number in the partitioning map is not between 0 and 999.

(5) The sum of all the values in the distribution file is greater than 4,294,967,295.

(6) The specified target partition map contains a node number that is not defined in SYSCAT.NODEGROUPDEF for the specified nodegroup.

User Response: The action corresponding to the reason code is:

(1) Ensure that the partitioning map file contains only a single value (if the resultant nodegroup is a single-node nodegroup) or exactly 4,096 values (if the resultant nodegroup is a multinode nodegroup).

(2) Ensure that the distribution file contains exactly 4,096 values, one per hash partition.

(3) Ensure that the values in the distribution file are integers greater than or equal to 0, and that the sum of all the distribution values is less than or equal to 4,294,967,295.

(4) Ensure that the node numbers are in the range of greater than or equal to 0 and less than or equal to 999.

(5) Ensure that the sum of all the distribution values for the 4,096 partitions is less than or equal to 4,294,967,295.

(6) Either issue ALTER NODEGROUP to add the missing node(s) or change the partitioning map file to exclude the node(s) not defined in sysibm.sysnodegroupdef.

SQL6054N **The archive file** *name* **is not a valid log file for the database** *name* **on node** *node-number*.

Explanation: An archive log file exists in the log directory on the specified node, but it is not valid.

The ROLLFORWARD DATABASE processing stops.

User Response: To determine the correct archive log file, issue the ROLLFORWARD DATABASE command with the QUERY STATUS option. Move the correct archive log file to 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 issue the ROLLFORWARD DATABASE command again.

SQL6055N **The archive file** *name* **does not belong to database** *name* **on node** *node-number*.

Explanation: An archive log file in the log directory on the specified node does not belong to the specified database.

The ROLLFORWARD DATABASE processing stops.

User Response: To determine the correct archive log file, issue the ROLLFORWARD DATABASE command with the QUERY STATUS option. Move the correct archive log file to 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 issue the ROLLFORWARD DATABASE command again.

SQL6056N **Nodegroup cannot be redistributed. Reason code =** *reason-code*.

Explanation: The operation cannot be executed. The reason code indicates the error.

(1) The nodegroup specification is not legal. The resulting nodegroup after redistribution would not contain any nodes.

(2) A previous redistribution operation did not complete successfully.

(3) A redistribution operation is already in progress.

(4) There is no previously aborted redistribution command to CONTINUE or ROLLBACK.

(5) No data redistribution is performed because the data in the nodegroup is already distributed as specified.

(6) The REDISTRIBUTE NODEGROUP command was not submitted from the catalog node.

(7) The REDISTRIBUTE NODEGROUP command is not available nor applicable in this non-partitioned version of the product.

(8) Redistribution is not allowed if there are user temporary table spaces in the nodegroup with existing declared temporary tables.

(9) The REDISTRIBUTE NODEGROUP command is not allowed if there are table spaces in the nodegroup with tables in the access mode No Data Movement (ACCESS_MODE='D' in SYSCAT.TABLES).

User Response: The action corresponding to the reason code is:

(1) Do not drop all the nodes of the nodegroup during redistribution.

(2) Identify why the previous redistribution failed and take the required corrective action. Submit the REDISTRIBUTE NODEGROUP command using either the CONTINUE or ROLLBACK option. CONTINUE will complete the previously aborted redistribution operation and ROLLBACK will undo the effects of the previously aborted operation.

(3) Issue the next REDISTRIBUTION NODEGROUP command after the current one completes.

(4) You cannot invoke the CONTINUE or ROLLBACK option on a nodegroup that is not associated with a failed redistribution operation.

(5) Try using a different target partitioning map or distribution file. Otherwise, redistribution is not necessary.

(6) Issue the command again from the catalog node.

(7) Do not issue the REDISTRIBUTE NODEGROUP command using this version of the product.

(8) Request redistribution again when there are no declared temporary tables using the user temporary table spaces in the nodegroup.

(9) Execute SET INTEGRITY IMMEDIATE

CHECKED statement on the dependent immediate materialized query and staging tables of the tables in No Data Movement mode, and then issue the REDISTRIBUTE NODEGROUP command again.

SQL6057N **Archive file** *name* **is not associated with the database** *name* **that was restored, or with the log file previously processed on node** *node-number***.**

Explanation: An archive log file is found in the log directory on the specified node, but does not belong to the specified database.

The ROLLFORWARD DATABASE processing stops.

User Response: To determine the correct archive log file, issue the ROLLFORWARD DATABASE command with the QUERY STATUS option. Move the correct archive log file to 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 issue the ROLLFORWARD DATABASE command again.

SQL6058N **Roll-forward recovery stopped because of error** *error* **while retrieving log file** *name* **for database** *name* **on node** *node-number***.**

Explanation: Roll-forward processing invokes the *db2uexit* to retrieve the log file for the database on the specified node. The error may have occurred in the *db2uexit*.

The ROLLFORWARD DATABASE processing stops.

User Response: See the user exit documentation in the *Administration Guide* for a description of the error, and either resume or terminate the roll-forward recovery.

SQL6059N **The point-in-time passed to the Rollforward utility must be greater than or equal to** *timestamp*, **because database** *name* **on node(s)** *node-list* **contains information later than the specified time.**

Explanation: An advanced virtual timestamp is in the database backup.

If ″,...″ is displayed at the end of the node list, see the syslog file for the complete list of nodes.

User Response: Do one of the following:

- Issue the command again with a point-in-time that is greater than or equal to *timestamp*.

- Restore an earlier backup on node(s), then issue the ROLLFORWARD DATABASE command again.

SQL6061N **Roll-forward recovery on database** *name* **cannot reach the specified stop point (end-of-log or point-in-time) because of missing log file(s) on node(s)** *node-list***.**

Explanation: The Rollforward Database utility cannot find the necessary log file(s) in the logpath.

User Response: Do one of the following:

- Use the ROLLFORWARD DATABASE command with the QUERY STATUS option to determine which log files are missing. When you find the log files, put them in the log path and resume forward recovery.

- If you cannot find the missing log files, restore the database on all nodes, then do point-in-time recovery using a timestamp that is earlier than that of the earliest missing log file.

SQL6062N **Roll-forward recovery on database** *name* **cannot complete because the log information on node(s)** *node-list* **does not match the corresponding record(s) on the catalog node.**

Explanation: The Rollforward utility processed all the log files found on each node, but the stop points on the specified nodes do not match the corresponding records on the catalog node. This can be caused by missing log files on either the catalog node or on the specified nodes, or the catalog node must be included in the list of nodes to be rolled forward.

The ROLLFORWARD DATABASE processing stops.

User Response: Do one of the following:

- Check if the catalog node needs to be rolled forward. If it does, submit the ROLLFORWARD command again and include the catalog node.

- Use the ROLLFORWARD DATABASE command with the QUERY STATUS option to determine which log files are missing. When you find the log files, put them in the log path and resume forward recovery.

- If you cannot find the missing log files, restore the database on all nodes, then do point-in-time recovery using a timestamp that is earlier than that of the earliest missing log file.

SQL6063N **Roll-forward recovery on database** *name* **has stopped on node(s)** *node-list* **due to a change in log file size.**

Explanation: The Rollforward Database utility has stopped rolling forward because the log files have changed in size. It must be restarted to establish the new log file size.

If ",..." is displayed at the end of the node list, see the diagnostic log for the complete list of nodes.

Roll-forward recovery has stopped.

(Note : if you are using a partitioned database

server, the node number(s) indicate which node(s) the error occurred on. Otherwise, it is not pertinent and should be ignored.)

User Response: Reissue the ROLLFORWARD command to continue.

SQL6064N **SQL error** *sqlcode* **occurred during data redistribution.**

Explanation: An error occurred during data redistribution.

The utility stops processing.

User Response: Look at the SQLCODE (message number) in the message for more information. Make any required changes and try the request again.

SQL6065N **An error occurred while writing to file** *file*.

Explanation: One of the following conditions was encountered:

- The file could not be opened.

- An I/O error occurred while writing to the file.

- An I/O error occurred while closing the file.

The command or the utility cannot be processed.

User Response: Ensure that the file exists and that the file permissions allow write access. Try the command or the utility again.

SQL6067W **ROLLFORWARD DATABASE QUERY STATUS command encountered sqlcode** *sqlcode*.

Explanation: The ROLLFORWARD DATABASE QUERY STATUS command encountered an error with the sqlcode *sqlcode*. The query might not be successful on some nodes for various reasons. The most severe error is indicated by *sqlcode*. The roll-forward status is only returned for the successful nodes.

User Response: Look up the sqlcode *sqlcode* in the *DB2 Messages Reference*, or online to determine the problems on the failed nodes. Take

the required corrective actions, then continue the forward recovery on these nodes.

SQL6068W **Rollforward operation was cancelled successfully. The database has to be restored on node(s)** *node-list***.**

Explanation: A Rollforward operation was cancelled before it successfully completed, and the database is left in an inconsistent state. The restore pending flag is on for the listed nodes.

If ″,...″ is displayed at the end of the node list, see the syslog file for the complete list of nodes.

User Response: Restore the database on the listed nodes.

SQL6069N **The ROLLFORWARD DATABASE command cannot be submitted on a non-catalog node.**

Explanation: The ROLLFORWARD DATABASE command is only accepted on the catalog node.

User Response: Submit the command on the catalog node.

SQL6071N **The requested operation cannot be processed because a new node has been added to the system. The system must be stopped and started again before the operation can be done.**

Explanation: One of the following:
- The request was issued from the new node, and this node cannot communicate with other nodes.
- A CREATE or DROP DATABASE operation was requested before all nodes were stopped and started again to include the new node.

User Response: Issue db2stop to stop all the nodes. When all nodes are successfully stopped, issue db2start to start all the nodes, including the new node, then try the request again.

sqlcode: -6071

sqlstate: 57019

SQL6072N **DB2START with the RESTART option cannot proceed because the specified node is already active.**

Explanation: The node specified for restart is already active on the system.

User Response: If necessary, issue a DB2STOP specifying the node to stop and issue the DB2START command again to restart the node.

SQL6073N **Add Node operation failed. SQLCODE =** *sqlcode***.**

Explanation: The Add Node operation failed with sqlcode *sqlcode*.

User Response: Check the message associated with the sqlcode that is returned in either the *DB2 Messages Reference*, or online.

Take any required corrective action, then try the request again.

SQL6074N **Add Node operation failed because a Create or Drop Database operation is currently executing.**

Explanation: An Add Node operation cannot execute concurrently with a Create or Drop Database operation.

User Response: Wait for the Create or Drop Database operation to complete, then try the request again.

SQL6075W **The Start Database Manager operation successfully added the node. The node is not active until all nodes are stopped and started again.**

Explanation: The db2nodes.cfg file is not updated to include the new node until all nodes are simultaneously stopped by the STOP DATABASE MANAGER (db2stop) command. Until the file is updated, the existing nodes cannot communicate with the new node.

User Response: Issue db2stop to stop all the nodes. When all nodes are successfully stopped,

issue db2start to start all the nodes, including the new node.

SQL6076W   **Warning! This command will remove all database files on the node for this instance. Before continuing, ensure that there is no user data on this node by running the DROP NODE VERIFY command.**

Explanation: This procedure will remove all the database partitions from the specified node.

User Response: Ensure that the DROP NODE VERIFY command was run before dropping this node. If using the API, ensure that the callerac parameter is specified correctly.

SQL6077W   **The db2stop DROP NODENUM procedure ended successfully, but could not remove all the files. See the file *file* for details.**

Explanation: The db2stop DROP NODENUM procedure ended successfully, but some user data files are still on the node.

User Response: Information in *file* file will indicate the directory structure from which files could not be deleted.

SQL6078N   **The db2stop DROP NODENUM procedure could not update the database information for database *dbname*.**

Explanation: The db2stop DROP NODENUM procedure could not access the catalog node for database *dbname*.

User Response: Try the request again. If the problem persists, contact your service representative.

SQL6079W   **The db2stop DROP NODENUM command was cancelled successfully.**

Explanation: The db2stop DROP NODENUM command was stopped before it began processing.

User Response: None.

SQL6080W   **The Start Database Manager operation successfully added in the node, but no database partitions have been created on the node. The node is not active until all nodes are stopped and started again.**

Explanation: The db2nodes.cfg file is not updated to include the new node until all nodes are simultaneously stopped by the STOP DATABASE MANAGER (DB2STOP) command. Until the file is updated, the existing nodes cannot communicate with the new node.

User Response: Issue DB2STOP to stop all the nodes. When all nodes are successfully stopped, issue DB2START to start all the nodes, including the new node. When all nodes are successfully started, the database system can be used.

SQL6081   **A communication error caused a DB2STOP FORCE command to time out on this node.**

Explanation: A communication error occurred on one or more of the database nodes, causing the DB2STOP FORCE command to time out on the current node, or, DB2STOP FORCE terminated because a severe error occurred during FORCE on one or more nodes. Any node where the communication error occurred will receive the SQL6048N message.

User Response: Do the following:

1. Correct the communication error on the node (or nodes) that received the SQL6048N message.

2. Issue a DB2START command and ensure that all nodes that received the SQL6048N message started successfully.

3. Issue the DB2STOP FORCE command again from any node.

SQL6100 - SQL6199

SQL6100N **The partitioning map in the data file and the partitioning map for the database are not the same.**

Explanation: The data to be loaded has either not been partitioned, or was partitioned with a partitioning map other than the current one for the nodegroup to which the table belongs. The data cannot be loaded.

User Response: If the data was not partitioned, use the db2split program to partition the data, then load the partitioned data.

If the data was partitioned, do one of the following:

- Use the partitioning map from the header of the data file to redistribute the nodegroup to which the table belongs. Then try the request again.
- Repartition the data with the current partitioning map for the nodegroup. Then try the request again to load the newly partitioned data.

SQL6101N **This data file contains data for node** *node-1* **but the Load utility is connected to node** *node-2***.**

Explanation: The data to be loaded is associated with a node number that differs from the node number of the node to which the application is connected. The data cannot be loaded.

User Response: Either find the data file associated with this node and try the request again with that data file, or connect to the node associated with this data file and issue the request at that node.

SQL6102W **The parameter** *name* **is reserved for future use. Its value should be set to** *default-value***.**

Explanation: A parameter reserved for future capabilities has been set to a value that is not its

default value. It should be set to the default value to ensure future compatibility.

User Response: Ensure that the parameter *name* is set to the value, *default-value*, then try the request again.

SQL6103C **An unexpected utility error occurred. Reason code =** *reason-code***.**

Explanation: An unexpected utility error occurred.

User Response: 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. See the *Administration 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

SQL6104N **The load utility does not support the creation of indexes.**

Explanation: The load utility does not support the creation of indexes, but the table to be loaded has at least one index defined on it. The index may have been explicitly created with the CREATE INDEX statement or implicitly created when a primary key was defined for the table.

User Response: Drop all indexes defined on the table with the DROP INDEX statement. Drop the primary key with the ALTER TABLE statement. Resubmit the command.

Use CREATE INDEX and ALTER TABLE as required to recreate indexes and primary key after the load completes successfully.

SQL6105W The Load utility has completed processing. A Rollforward done to a point in time after the load will not succeed. Take a database backup now if database recoverability is required.

Explanation: No logging is done by the Load utility. If you try to roll forward with a backup made before the load, the operation will fail when it encounters references to the data that was loaded.

User Response: To ensure database recoverability after the load, do a backup before the data is modified.

SQL6106N The filetype modifier "NOHEADER" was specified, but the nodegroup on which the table is defined is not a single-node nodegroup.

Explanation: The data to be loaded was specified as having no header information. However the target table specified is not a single-node table. The data cannot be loaded.

User Response: The data must be split using db2split, then loaded without the "NOHEADER" option.

SQL6107N The partitioning key information in the data file is not correct.

Explanation: Either the data was not split with db2split or the db2split operation was not successful.

User Response: Use the db2split program to partition the data and then try the request again with the partitioned data. If the insert-column option is being used, ensure that all of the partitioning columns are specified in the column list.

If the problem persists, contact your technical service representative with the following information:

- Problem description
- SQLCODE and embedded reason code
- SQLCA contents if possible
- Trace file if possible

SQL6108N The number of partitioning keys defined in the data file header (*number-1*) does not match the number of partitioning keys defined for the table (*number-2*).

Explanation: The partitioning columns specified in the db2split configuration file were not correct. The data was not correctly split.

User Response: Do the following:

1. Ensure that the correct partitioning columns are specified in the db2split configuration file.
2. Split the data.
3. Issue the Load operation with the newly partitioned data.

SQL6109N Utility expected partitioning column *column-name-1*, but found partitioning column *column-name-2*.

Explanation: In the db2split configuration file:

- One of the partitioning columns defined for the table was not specified.
- The order of the partitioning columns was incorrect
- A column was specified that is not a partitioning column for the table.

User Response: Do the following:

1. Ensure that the db2split configuration file is correct.
2. Split the data.
3. Issue the Load operation with the newly partitioned data.

SQL6110N Utility expected partitioning column type *column-type-1* for column *column-name-1*, but the data file lists it as type *column-type-2*.

Explanation: The db2split configuration file is not correct.

User Response: Do the following:

1. Ensure that the db2split configuration file is correct.
2. Split the data.
3. Issue the Load operation with the newly partitioned data.

SQL6111N   Cannot create a subdirectory under the path specified by newlogpath.

Explanation: When the newlogpath parameter is updated, the system attempts to create a subdirectory under the specified path using the node name as the name of the subdirectory. One of the following operating system errors prevented the creation of the subdirectory:

- The file system or path does not have the appropriate permissions for file creation.
- The file system does not have enough disk space.
- The file system does not have enough file blocks or inodes.

The requested change is not made.

User Response: Do one of the following, then try the request again:

- Ensure that the specified path exists and that the filesystem and path have read/write permissions.
- Specify a different newlogpath.

If the problem persists, see your system administrator.

SQL6112N   The requested change cannot be done. The resulting configuration parameter settings are not valid. Reason code *reason-code***.**

Explanation: Three rules govern the settings of some configuration parameters. The rule that would be violated is indicated by the *reason-code*:

(1)   max_coordagents + num_initagents <= maxagents

(2)   num_initagents <= num_poolagents

(3)   maxdari <= max_coordagents

User Response: Ensure that the values you specify do not violate any of the above rules, then try the request again.

SQL6500 - SQL6599

SQL6500W   RESTARTCOUNT in the load command may cause some problems.

Explanation: Since multiple loading processes for the same table are completely independent, it is almost impossible to have an identical restartcount for those multiple loading processes.

User Response: Please make sure you have the correct load command.

SQL6501N   Database name was not given in the load command.

Explanation: Database name has to be specified in the load command.

User Response: Please specify the database

name and rerun the command.

SQL6502N   The path name (parameter: data_path) for data file was not specified.

Explanation: If the input data file is remote, the file will be transferred locally. The path to the file on the remote machine must be supplied.

User Response: Please specify the path name to the remote data file and rerun the command.

SQL6504N **There are errors in the output node list specification (parameter: outputnodes) in the configuration file.**

Explanation: The output node list specification is not valid.

User Response: Check the sample configuration file, correct your output node list specification, and rerun the command.

SQL6505N **There are errors in the partitioning database partition list specification (parameter: PARTITIONING_DBPARTNUMS) in the load command.**

Explanation: The partitioning database partition list specification is not valid.

User Response: Correct your partitioning database partition list specification, and rerun the command.

SQL6506N **The program failed to extract partitioning key information for table** *table-name* **from the system catalog table.**

Explanation: Either the table is not defined, or it is not defined in an MPP environment.

User Response: Make sure you have the table properly defined.

SQL6507N **Checking level (parameter: check_level) in the configuration file is not valid.**

Explanation: Checking level (parameter: check_level) can be either CHECK or NOCHECK. The default is CHECK.

User Response: Correct the parameter in the configuration file, and rerun the command.

SQL6508N **The program failed to create the output pipe for the ftp process.**

Explanation: If input data files are remote, they will be transferred to a local pipe. If this local

pipe already exists, the process will fail.

User Response: Please ensure your working space is clean.

SQL6509N **The program failed to create input pipes for the partitioning agents.**

Explanation: The program can not create temporary input pipes for the splitter processes.

User Response: Please ensure your working space is clean.

SQL6510N **The program failed to create the temporary directory at local non-NFS space of partition** *partition-num***.**

Explanation: The program needs a temporary working directory at local non-NFS space of all partitioning and loading partitions.

User Response: Please ensure your working space is clean.

SQL6511N **Load failed to create output for the partitioning agent at partition** *partition-num***.**

Explanation: The program can not create temporary output pipes for the partitioning agent at partition *partition-num*.

User Response: Please ensure your working space is clean.

SQL6512N **Load failed to create input pipes for the merging agent at partition** *partition-num***.**

Explanation: The program can not create temporary input pipes for the merging agent at partition *partition-num*.

User Response: Please ensure your working space is clean.

SQL6513N **Load failed to create the input pipe for the loading agent at partition** *partition-num***.**

Explanation: The program can not create temporary input pipe for the loading agent at partition *partition-num*.

User Response: Please ensure your working space is clean.

SQL6514N **The program can not read the node configuration file:** *node-cfg-file***.**

Explanation: Either the file does not exist, or it is not readable.

User Response: Please check the existence and permissions of the node configuration file.

SQL6515N **The program failed to find a load command in the configuration file.**

Explanation: A CLP load command must be supplied in the configuration file.

User Response: Please specify a CLP load command in the configuration file.

SQL6516N **The program failed to connect to database** *db-name***.**

Explanation: Either the database manager has not been started yet, or it encountered problems.

User Response: Check your database manager status.

SQL6517N **Load failed to extract the partition list, where the table** *tbl-name* **is defined, from the system catalog table.**

Explanation: Either the table is not defined, or it is not defined in an MPP environment.

User Response: Check how the table was defined in the database.

SQL6518N **The record length (reclen in the load command) is not valid.**

Explanation: Valid record is between 1 to 32768.

User Response: Please correct the record length, and rerun the command.

SQL6519N **The mode (parameter: mode)** *mode* **in the configuration file is not valid.**

Explanation: The running mode of this program can be: SPLIT_ONLY, LOAD_ONLY, SPLIT_AND_LOAD (default), or ANALYZE.

User Response: Please correct the mode in the configuration file.

SQL6520N **The program failed to create output pipes for the process, which generates the header information for split file.**

Explanation: The program can not create output pipes for the process, which generates the header information for split file.

User Response: Please ensure your working space is clean.

SQL6521N **The configuration file** *cfg-file* **for this program doesn't exist.**

Explanation: The program needs a configuration file.

User Response: Please create a configuration file.

SQL6522N **The program found a path name for the input data files in the load command.**

Explanation: Path name for input data files in the load command is not allowed. There is a separate parameter (data_path) for this purpose.

User Response: Please correct the configuration file.

SQL6523N **The element** *partition-num* **in the partitioning database partition list (parameter: PARTITIONING_DBPARTNUMS) is not defined in the node configuration (db2nodes.cfg) file.**

Explanation: All partitions in the partitioning database partition list must have an entry in the node configuration file.

User Response: Please correct your partitioning database partition list.

SQL6524N **The element** *partition-num* **in the output partition list (parameter: OUTPUT_DBPARTNUMS) is not a member of the partition list where the table is defined.**

Explanation: All partition in the output partition list must be members of the partition list where the table is defined.

User Response: Please correct your output partition list.

SQL6525N **The program can not read the input data file** *file-name***.**

Explanation: Either the input data file was not found, or it is not readable.

User Response: Please check the existence and permissions of the input data file.

SQL6526N **The program can not write in current working directory** *cwd***.**

Explanation: The current working directory is not writable.

User Response: Please check the permission of current working directory.

SQL6527N **The partition where statistic data will be collected, (parameter: RUN_STAT_DBPARTNUM), is not a member of the output partition list.**

Explanation: The partition where statistic data will be collected must be a member of the output partition list.

User Response: Please correct the RUN_STAT_DBPARTNUM parameter.

SQL6528N **Record length was not specified in the load command.**

Explanation: If the BINARYNUMERICS or PACKEDDECIMAL modifier is specified in the load command, record length (reclen) must be specified in the load command as well.

User Response: Please correct the load command.

SQL6529N **The No Header option (NOHEADER) was not specified in the load command.**

Explanation: If a table is defined in a single node nodegroup, the NOHEADER modifier must be specified in the load command.

User Response: Please correct the load command.

SQL6530N **The data type of one partitioning key is float or double.**

Explanation: If the input file is a non-binary data file, float or double columns can not be defined as partitioning key.

User Response: Supply a binary data file, or change the definition of your table.

SQL6531N **The program failed to reset tablespace quiesce.**

Explanation: There are probably on-going loading processes. Make sure all previous loading processes are completed before another AutoLoader session can be started.

User Response: Check the processes status on your machine.

SQL6532N **A savecount in the load command can not be set to non-zero.**

Explanation: The savecount in the load command can not be set to non-zero if there are multiple partitioning nodes, the mode is PARTITION_AND_LOAD, and the command is either a REPLACE INTO or INSERT INTO load command.

User Response: Please correct the load command.

SQL6533N **A restartcount in the load command can not be set to non-zero.**

Explanation: Because multiple partitioning nodes produce random order of records for loading process(es), using RESTART INTO with a restartcount can not guarantee a successful recovery.

User Response: Please correct the load command.

SQL6534N **There is an error in the .netrc file** *netrc-file***.**

Explanation: Either the .netrc file could not be found, or there is no entry for remote host *machine*, or the permissions of the file are not correct.

User Response: Please check the existence and permission of the .netrc file.

SQL6535N **The mode PARTITION_ONLY or ANALYZE is not valid.**

Explanation: If a table is defined in a single node nodegroup, either partitioning or analyzing is not necessary.

User Response: Please change the mode to LOAD_ONLY or PARTITION_AND_LOAD.

SQL6536N **The program** *progname* **failed to open file** *filename* **for reading.**

Explanation: An AutoLoader process can not successfully open a file or pipe for reading.

User Response: Please ensure everything is correct in the configuration file.

SQL6537N **The program** *progname* **failed to open file** *filename* **for writing.**

Explanation: An AutoLoader process can not successfully open a file or pipe for writing.

User Response: Please ensure everything is correct in the configuration file.

SQL6538N **Load failed to read the partitioned file** *partitioned-file***.**

Explanation: When Load is invoked in LOAD_ONLY mode, the input data file must have already been partitioned, and all partitioned files must be readable by the Load.

User Response: Check whether or not the input data file has been partitioned, and the access permissions of the resulting partitioned files.

SQL6539N **There is at least one command in** *cmd-list* **not found in the working environment.**

Explanation: The execution of this program depends on a number of common Unix commands. The process fails if any one of them is not available in the working environment.

User Response: Make sure you have all necessary commands properly installed on your system.

SQL6540N **File type** *file-type* **specified in the load command is not valid.**

Explanation: Valid file types are ASC (positional ASCII) or DEL (delimited ASCII).

User Response: Correct the load command in the configuration file.

SQL6550N **The partitioning map file** *map-file-name* **could not be opened for writing.**

Explanation: The file name and path for the partitioning map could not be opened. An error occurred.

User Response: Confirm that the partitioning map file name and file path were specified correctly and that the file can be opened for writing.

SQL6551N **An error occurred while attempting to write to the partition map file.**

Explanation: A filesystem error occurred when writing to the partitioning map file.

User Response: Check that the file path is correct, and that the target device has sufficient space to hold the partitioning map output.

SQL6552N **An error occurred while attempting to open a temporary configuration file** *filename* **for writing.**

Explanation: The file name and path for a temporary file could not be opened. An error occurred.

User Response: Confirm that the utility temporary file storage path was specified correctly and that the path allows for files to be opened for writing.

SQL6553N **An error occurred while attempting to write to a temporary configuration file** *filename*.

Explanation: A filesystem error occurred when writing to a temporary file.

User Response: Check that the file path is correct, and that the target device has sufficient space for the file data.

SQL6554N **An error occurred when attempting to remotely execute a process.**

Explanation: The utility attempted to start a child process on a different database partition, but an error occurred.

User Response:

- If no user ID or password was provided to the utility for remote access, ensure that the user ID invoking the utility is authorized to execute programs on the target node(s).

- If a user ID and password were supplied to the utility confirm that they were supplied correctly.

- If running on NT make sure the DB2 installation has correctly defined an NT Service for the splitter operation on all nodes.

- If you are unable to resolve this problem, please contact DB2 service.

SQL6555N **The load utility encountered an unexpected communications error.**

Explanation: The utility encountered an error attempting one of the following operations:

- Attempting to connect to a TCP/IP socket.
- Attempting to read or write a TCP/IP msg.
- Attempting to initialize TCP/IP communications
- Attempting to retrieve a complete host name.
- Attempting to select an active TCP/IP socket.
- Attempting to close an active socket.
- Attempting to retrieve a port number.

User Response:

- If service name setup was required for the version of the load utility you are using, ensure that the service names have been defined correctly.

- If you are running concurrent load utility jobs, ensure that you have followed the documented setup requirements to avoid service name conflicts between the concurrent utility jobs.

- If the problem persists, contact your DB2 Service representative.

SQL6556W **An incomplete record was detected at the end of file** *filename*.

Explanation: An incomplete data record was found at the end of a data file provided to the utility by the user.

User Response: Examine the source data for syntactical correctness.

SQL6557N **Failed to retrieve the default node number.**

Explanation: The utility attempted to determine a default node number, but was unable to.

User Response: State source and target node numbers explicitly in the utility configuration file, or contact DB2 Service for assistance.

SQL6558N **The utility failed to determine the current working directory and/or drive.**

Explanation: The utility attempted to determine the current working directory and/or drive but encountered an error.

User Response: Contact DB2 Service for assistance.

SQL6559N **An invalid command line option was provided to the AutoLoader utility.**

Explanation: The user specified a command line option for the AutoLoader utility which is either not supported, or obsolete.

User Response: Refer to the AutoLoader documentation, or online help, for the supported options and features.

SQL6560N **Node** *node-number* **which is an execution node for partitioning does not appear in the db2nodes.cfg file.**

Explanation: A node specified as an execution node for partitioning does not appear to be a member in the db2nodes.cfg file. The work intended to complete on this node can not be started.

User Response: Add the node to the node list definition in the db2nodes.cfg file, or specify an alternate node for the partition operation which is a member of the node configuration.

SQL6561N **Target node** *node-number* **for load does not appear in the nodegroup.**

Explanation: A node was specified as a target node for loading, but this node is apparently not a member of the node group being loaded.

User Response: Check the nodegroup definition and confirm that the target node specified for loading is part of this nodegroup. If the node is not part of the nodegroup correct the utility target node specification to include the correct list of nodes. If the node is part of the nodegroup, please contact DB2 Service for assistance.

SQL6562N **The utility is unable to retrieve the instance name.**

Explanation: The utility attempted to retrieve the instance name but encountered an error.

User Response: Confirm that the utility is running on a node with DB2 installed, and a valid instance is running. For additional help, please contact DB2 Service.

SQL6563N **Failed to retrieve the current user ID.**

Explanation: The utility attempted to retrieve the current user ID for the ID, but an error was encountered.

User Response: Contact DB2 Service.

SQL6564N   The password supplied is not valid.

Explanation: The user supplied an explicit password to the utility, but the password is not valid.

User Response: Supply a valid password.

SQL6565I   Usage: db2atld [-config config-file] [-restart] [-terminate] [-help]

Explanation:
- The '-config' option will run this program using a user specified configuration file; the default is autoload.cfg.
- The '-restart' option will run this program in restart mode; the configuration file should not be modified since the last incomplete AutoLoader job.
- The '-terminate' option will run this program in terminate mode; the configuration file should not be modified since the last incomplete AutoLoader job.
- The '-help' option will generate this help message.

The AutoLoader configuration file is a user supplied file which contains the LOAD command to be executed, the target database, and several optional parameters that the user may specify. The sample configuration file, 'AutoLoader.cfg', supplied in the samples directory contains inline comments which describe the available options and their defaults. When running this program with '-restart' and '-terminate' option, user should not modify the configuration file from their last incomplete job.

User Response: Refer to the DB2 documentation for further details about the AutoLoader utility.

SQL6566N   The LOAD command is missing from the AutoLoader configuration file.

Explanation: The LOAD command is missing from the AutoLoader configuration file. The

parameter must be specified.

User Response: Ensure that you have specified the correct configuration file for the AutoLoader, and that the LOAD command is specified inside.

SQL6567N   The *option-name* option appears multiple times in the AutoLoader configuration file.

Explanation: An option parameter was specified multiple times inside the AutoLoader configuration file.

User Response: Correct the configuration file so that each option appears at most only once.

SQL6568I   The Load utility is now issuing all *request-type* requests.

Explanation: The Load utility is now dispatching the *request-type* operation on each of the partitions.

User Response: This is an informational message.

SQL6569I   The AutoLoader is now issuing all split requests.

Explanation: The AutoLoader is now issuing the split operation on each of the target split partitions.

User Response: This is an informational message.

SQL6570I   The AutoLoader is waiting for all splitters to complete.

Explanation: The AutoLoader is waiting for all splitters to complete.

User Response: This is an informational message.

SQL6571I   The Load utility is waiting for all operations to complete.

Explanation: The Load utility is waiting for operations to complete.

User Response: This is an informational message.

SQL6572I   The LOAD operation has begun on partition *node-number*.

Explanation: The LOAD operation has begun on the specified partition.

User Response: This is an informational message.

SQL6573I   The remote execution of the splitter utility on partition *node-number* **finished with remote execution code** *code*.

Explanation: The remote execution of the splitter utility on the specified partition has completed.

User Response: This is an informational message.

SQL6574I   The utility has read *MB-count* **megabytes from the source data.**

Explanation: This information is generated periodically to provide status to the user on the progress of large Load jobs.

User Response: This is an informational message.

SQL6575I   The utility has completed reading *MB-count* **megabytes from the user data.**

Explanation: This message is written at the completion of the Load to indicate the total volume of user data that has been processed.

User Response: This is an informational message.

SQL6576N   The AutoLoader utility encountered a threading error. Reason code *reason-code*, **return code** *ret-code*.

Explanation: The following is an explanation of the reason code *reason-code*:

- 1 - The AutoLoader utility tried to create a thread but failed with return code *ret-code*.
- 2 - The AutoLoader utility tried to wait for a thread to finish, but failed with return code *ret-code*.

User Response: Ensure that you are running on an operating system which supports threaded applications, and that your thread limit per process is sufficient. The thread requirements are as follows:

- One thread for every load process which must be started,
- One thread for every splitter process,
- One thread for data feeding to the splitter processes.

SQL6577N   The AutoLoader utility does not support the ROWCOUNT option in the load command.

Explanation: The ROWCOUNT option in the load command is not supported in the AutoLoader utility.

User Response: Correct your load command in the AutoLoader configuration file, and resubmit your command.

SQL6578N   Invalid AutoLoader option. RESTART/TERMINATE option can only be specified with SPLIT_AND_LOAD or LOAD_ONLY mode.

Explanation: RESTART/TERMINATE option in AutoLoader can only work with SPLIT_AND_LOAD or LOAD_ONLY mode.

User Response: Please check the AutoLoader configuration file or the autloader option flags.

SQL6579N   **Invalid LOAD command in the AutoLoader configuration file. RESTART and TERMINATE options of the AutoLoader are used to perform LOAD RESTART and LOAD TERMINATE operation respectively.**

Explanation: Users should not specify RESTART nor TERMINATE in the LOAD command. Instead, the RESTART and TERMINATE options of the AutoLoader should be used.

User Response: Without modifying the AutoLoader configuration file, users should launch db2atld with option RESTART or TERMINATE.

SQL6580I   **LOAD is restarting on node** *node-num* **in phase** *restarting-phase***.**

Explanation: AutoLoader has acknowledged that LOAD is restarting in either LOAD/BUILD/DELETE phase.

User Response: This is an informational message.

SQL6581I   **Load cannot restart on node** *node-num***.**

Explanation: AutoLoader has acknowledged that LOAD cannot be restarted on a given node.

User Response: This is an informational message.

SQL6582I   **Restarting LOAD on node** *node-num* **is not required.**

Explanation: AutoLoader has acknowledged that LOAD does not need to be restarted on a given node.

User Response: This is an informational message.

SQL6583N   **The partitioning key definition is incompatible with partitioned database load mode** *load-mode***.**

Explanation: An identity column was specified as part of the partitioning key definition, but the load mode specified was not PARTITION_AND_LOAD and the identityoverride modifier was not specified.

User Response: Either change the load mode to PARTITION_AND_LOAD, specify the identityoverride modifier, or remove the identity column from the partitioning key definition.

SQL7000 - SQL7099

SQL7001N   **Unknown command** *command* **was requested.**

Explanation: The command submitted to REXX could not be recognized.

The command cannot be processed.

User Response: 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.**

Explanation: The user specified an incorrect cursor name.

The command cannot be processed.

User Response: Verify that the cursor name is one of the form "c1" through "c100".

SQL7003N   **The statement name is invalid.**

Explanation: The user has specified an incorrect statement name.

The command cannot be processed.

User Response: Verify that the statement name is of the form "s1" through "s100".

SQL7004N   The syntax of the request is invalid.

Explanation: REXX was unable to parse the command string submitted.

The command cannot be processed.

User Response: Use proper command syntax.

SQL7005W   The cursor used by this OPEN statement has not been declared.

Explanation: An OPEN statement was attempted without a declaring a cursor.

The command cannot be processed.

User Response: Insert a DECLARE statement before the OPEN statement and rerun the procedure.

SQL7006N   Invalid keyword *keyword* **given on** *request***.**

Explanation: The statement contains a not valid keyword *keyword*.

The command cannot be processed.

User Response: Specify the correct keyword format.

SQL7007N   REXX variable *variable* **does not exist.**

Explanation: A REXX variable that did not exist in the REXX variable pool was passed.

The command cannot be processed.

User Response: 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.**

Explanation: A variable that contained inconsistent data was passed to REXX.

The command cannot be processed.

User Response: 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***.**

Explanation: The variable *variable* passed to REXX contained inconsistent data. The data string from *variable* was truncated.

The command cannot be processed.

User Response: 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***.**

Explanation: The scan ID *variable* passed to REXX did not exist or contained inconsistent or missing data.

The command cannot be processed.

User Response: Verify that the data contained in the scan ID has been assigned correctly, and run the procedure again.

SQL7011N   Required parameter *parameter* **not given.**

Explanation: The parameter *parameter* is required by REXX command syntax but was not specified.

The command cannot be processed.

User Response: Specify the required parameter value, and run the procedure again.

SQL7012N   Attempted ISL change while connected to a database.

Explanation: Isolation level (ISL) may not be changed while connected to a database.

The command is ignored.

User Response: If a change in isolation level is necessary, disconnect from the current database,

then set the isolation level and connect to the database.

SQL7013N Mismatched cursor and statement name or hold attribute.

Explanation: 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.

User Response: Verify that the cursor and statement numbers match and rerun the procedure.

SQL7014N Incorrect number of host variable components.

Explanation: 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.

User Response: 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.

Explanation: The variable name specified is not valid in REXX. The name must conform to the language requirements.

The command cannot be processed.

User Response: Change the name to one that conforms to the REXX requirements and resubmit the command.

SQL7016N Invalid syntax provided to the SQLDB2 interface. Related Error: *db2-error*. The command cannot be processed.

Explanation: Invalid syntax was provided to the SQLDB2 interface, e.g. both an input file and a command were specified.

User Response: Look up the related error code for more information.

SQL7032N SQL procedure *procedure-name* not created. Diagnostic file is *file-name*.

Explanation: SQL Procedure *procedure-name* was not created. One of the following has occurred:

- Support for SQL stored procedures is not installed and configured on this server. You need to have the DB2 Application Development Client and a C compiler installed on the server to create SQL procedures. You may need to set the DB2 registry variable DB2_SQLROUTINE_COMPILER_PATH to point to a script or batch file that contains environment settings for the C compiler on your platform.
- DB2 failed to precompile or compile the SQL stored procedure. DB2 creates an SQL procedure as a C program that contains embedded SQL. Errors not found during the initial parsing of the CREATE PROCEDURE statement can be found during the precompile or compile stage.

For UNIX platforms, the full path of the file that contains diagnostic information is:

$DB2PATH/function/routine/sqlproc/ \
$DATABASE/$SCHEMA/tmp/*file-name*

where $DATABASE represents the name of the database, and $SCHEMA represents the schema name of the SQL procedure.

For OS/2 and Windows operating systems, the full path of the file that contains diagnostic information is:

%DB2PATH%\function\routine\sqlproc\ \
%DATABASE%\%SCHEMA%\tmp*file-name*

where %DATABASE% represents the name of the database, and %SCHEMA% represents the schema name of the SQL procedure.

User Response: Ensure that both a compatible C compiler and a DB2 Application Development Client are installed on the server. If a precompile or compile error occurred, refer to the messages from the precompiler or compiler in the diagnostic file *file-name*.

Ensure that the DB2 registry variable DB2_SQLROUTINE_COMPILER_PATH is set to point to a script or batch file that sets up the C compiler environment. On a UNIX operating system, for example, you may create a script called "sr_cpath" in the /home/DB2INSTANCE/sqllib/function/routine directory. To set the DB2 registry variable DB2_SQL_ROUTINE_COMPILER_PATH accordingly, issue the following command:

```
db2set DB2_SQLROUTINE_COMPILER_PATH = \
  " \
  /home/DB2INSTANCE/sqllib/function/ \
  routine/sr_cpath"
```

sqlcode: -7032

sqlstate: 42904

SQL7035W   The executable of the SQL procedure *procedure-name* **is not saved in the database catalog.**

Explanation: The executable of the SQL procedure exceeds the limit of 2 megabytes and

therefore cannot be stored in the database catalog. It will not be recovered automatically at database restore time or on ROLLBACK of a DROP PROCEDURE statement.

User Response: If the warning resulted from a CREATE PROCEDURE statement, ensure that you keep a backup of the executable associated with SQL procedure *procedure-name*. If the warning was issued during a restore operation or a ROLLBACK of a DROP PROCEDURE statement, then the executable associated with *procedure-name* must be manually synchronized with the SQL procedure that is defined in the catalog.

sqlcode: +7035

sqlstate: 01645

SQL7099N   Invalid error *error* **has occurred.**

Explanation: A REXX internal error has occurred.

The command cannot be processed.

User Response: Verify that REXX is installed properly and rerun the procedure. If the error continues, note the error number and call the dealer.

SQL8000 - SQL8099

SQL8000N   DB2START processing failed; a valid product license was not found.

Explanation: A valid license key could not be found and the evaluation period has expired.

User Response: Install a license key for the fully entitled version of the product. You can

obtain a license key for the product by contacting your IBM representative or authorized dealer.

SQL8001N   DB2 connect processing failed; a valid product license was not found.

Explanation: A valid license key could not be found and the evaluation period has expired.

User Response: Install a license key for the fully entitled version of the product. You can obtain a license key for the product by contacting your IBM representative or authorized dealer.

sqlcode: -8001

sqlstate: 42968

SQL8002N **DRDA connect processing failed; a valid product license was not found.**

Explanation: A valid license key could not be found and the evaluation period has expired.

User Response: Install a license key for the fully entitled version of the product. You can obtain a license key for the product by contacting your IBM representative or authorized dealer.

sqlcode: -8002

sqlstate: 42968

SQL8006W **The product** *product-name* **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 FOLLOWING DIRECTORY:** *directory-name***.**

Explanation: A valid license key has not been installed for this product. 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.

User Response: 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. 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 evaluation period for the product** *text***. For evaluation license terms and conditions, refer to the IBM Evaluation Agreement in the EVALUATE.AGR file, located in the following directory:** *text***.**

Explanation: A valid license key has not been installed for this product. The evaluation period will expire after the specified number of days.

User Response: 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.

SQL8008N **The product** *text* **does not have a valid license key installed and the evaluation period has expired. Functions specific to this product are not enabled.**

Explanation: A valid license key could not be found and the evaluation period has expired.

User Response: Install a license key for the fully entitled version of the product. You can

obtain a license key for the product by contacting your IBM representative or authorized dealer.

SQL8009W   **The number of concurrent users of the DB2 Workgroup product has exceeded the defined entitlement of** *number*. **Concurrent user count is** *number*.

Explanation: The number of concurrent users has exceeded the number of defined DB2 concurrent user entitlements.

User Response: Contact your IBM representative or authorized dealer to obtain additional DB2 user entitlements and update the DB2 license information in the License Center.

sqlcode: +8009

sqlstate: 01632

SQL8010W   **The number of concurrent users of the DB2 Connect product has exceeded the defined entitlement of** *number*. **Concurrent user count is** *number*.

Explanation: The number of concurrent users has exceeded the number of defined DB2 concurrent user entitlements.

User Response: Contact your IBM representative or authorized dealer to obtain additional DB2 user entitlements and update the DB2 license information in the License Center.

sqlcode: +8010

sqlstate: 01632

SQL8011W   **One or more database partitions does not have a valid DB2 license key installed for the** *product-name* **product.**

Explanation: A valid license key has not been installed for this product on all database partitions. 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.

User Response: 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.ARG FILE. 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.

SQL8012W   **The number of concurrent users of the DB2 Enterprise product has exceeded the defined entitlement of** *number*. **Concurrent user count is** *number*.

Explanation: The number of concurrent users has exceeded the number of defined DB2 concurrent user entitlements.

User Response: Contact your IBM representative or authorized dealer to obtain additional DB2 user entitlements and update the DB2 license information in the License Center.

sqlcode: +8012

sqlstate: 01632

SQL8013W   **The number of concurrent database connections to the DB2 Connect product has exceeded the defined entitlement of** *number*. **Database connection count is** *number*.

Explanation: The license for the DB2 Connect product you are using supports a limited number of concurrent database connections. You have

requested a number of connections that exceeds this limit.

User Response: Upgrade to a version of the DB2 Connect product with a higher limit of concurrent database connections.

DB2 Connect Enterprise Edition users: obtain licenses for additional users by purchasing Additional User Packs.

sqlcode: +8013

sqlstate: 01632

SQL8014N **The version of the DB2 Connect product you are using is not licensed for use with TCP/IP protocol. Upgrade to a full function DB2 Connect product to use TCP/IP.**

Explanation: This version of DB2 Connect is limited to SNA connectivity. TCP/IP connections are not supported.

User Response: Upgrade to a full function DB2 Connect product such as DB2 Connect Personal Edition or DB2 Connect Enterprise Edition to use TCP/IP.

sqlcode: -8014

sqlstate: 42968

SQL8015N **The version of the DB2 Connect product you are using is not licensed for updating multiple databases in the same transaction.**

Explanation: This version of DB2 Connect is limited to working with a single database in a transaction. It does not support 2-phase commit protocol.

User Response: Upgrade to a full function DB2 Connect product such as DB2 Connect Personal Edition or DB2 Connect Enterprise Edition to be able to update multiple databases in a single transaction.

sqlcode: -8015

sqlstate: 42968

SQL8016N **The user** *user-name* **is not defined as a registered user for the product** *product-name***.**

Explanation: The user has not been registered to use this product.

User Response: Contact your IBM representative or authorized dealer to obtain additional DB2 registered user entitlements and update the registered user list for this product in the License Center.

SQL8017W **The number of processors on this machine exceeds the defined entitlement of** *licensed-quantity* **for the product** *product-name***. The number of processors on this machine is** *processor-quantity***. You should purchase additional processor based entitlements from your IBM representative or authorized dealer and update your license using the License Center or the db2licm command line utility. For more information on updating processor based licenses, refer to the Quick Beginnings manual. For more information on the db2licm utility, refer to the Command Reference.**

User Response: **sqlcode**: +8017

SQL8018W **The number of concurrent users of this product has exceeded the defined entitlement of** *number***. Concurrent user count is** *number***.**

Explanation: The number of concurrent users has exceeded the number of defined concurrent user entitlements.

User Response: Contact your IBM representative or authorized dealer to obtain additional user entitlements and update the DB2 license information in the License Center.

sqlcode: +8018

sqlstate: 01632

SQL8019N There was an error updating the license for the OLAP Starter Kit. RC = *reason-code***.**

Explanation: The license utility could not update the OLAP Starter Kit license because of an unknown error.

User Response: Try the command again. If the problem persists, please contact an IBM service representative.

sqlcode: -8019

SQL8020W The number of concurrent non-DB2 data sources for one server has exceeded the defined entitlement of *number-sources***. Current number of non-DB2 data sources is** *number-entitled***.**

Explanation: The number of concurrent non-DB2 data sources has exceeded the number of defined entitlements.

User Response: Contact your IBM representative or authorized dealer to obtain additional entitlements and then update the DB2 license information in the License Center.

sqlcode: +8020

SQL8021W The data source *source-name* **is not defined as a registered data source for** *product-name***.**

Explanation: The data source has not been configured as a registered data source. You must purchase entitlements for all non-DB2 data sources.

User Response: Contact your IBM representative or authorized dealer to obtain additional data source entitlements and update the registered data source list for this product in the License Center.

sqlcode: +8021

SQL8100 - SQL8199

SQL8100N The table is full.

Explanation: 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.

Federated system users: this situation can also be detected by the data source.

User Response:

- 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.

Federated system users: isolate the problem to the data source failing the request (refer to the Troubleshooting 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 the federated server, 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.**

Explanation: 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.

User Response:

- Check to ensure that the file systems are mounted correctly
- Restore the database from a backup
- Call your IBM service representative

SQL9300 - SQL9399

SQL9301N **An invalid option is specified or an option parameter is missing.**

Explanation: Either the option specified is invalid or an option parameter is not specified.

The command cannot be processed.

User Response: Correct the option and resubmit the command.

SQL9302N **Invalid option parameter:** *option-parameter*.

Explanation: The option parameter specified is invalid.

The command cannot be processed.

User Response: Correct the option parameter and resubmit the command.

SQL9303N **No** *option* **is specified.**

Explanation: The required option *option* is not specified.

The command cannot be processed.

User Response: Resubmit the command with the required option.

SQL9304N **Unable to open file** *filename* **for writing.**

Explanation: The command cannot be processed.

User Response: Ensure that the system can access the file.

SQL9305N **The** *name* **is too long. The maximum length is** *max-length*.

Explanation: The <name> is longer than the maximum length of <max-length>.

The command cannot be processed.

User Response: Ensure that the <name> does not exceed the maximum length.

SQL9306N **One or more of the field names are too long. The maximum length is** *max-length*.

Explanation: The total length of a field name includes the specified prefix and/or a column suffix, which can either be its name or its number. This total length must not exceed the maximum length.

The command cannot be processed.

User Response: Ensure that all the field names do not exceed the maximum length.

SQL9307N **Remarks cannot be retrieved from the database. Error code =** *sqlcode*.

Explanation: An error has occurred when attempting to retrieve the remarks of a column from the database.

The command cannot be processed.

User Response: Check the error code for more information.

SQL9308W **SQL data type** *sqltype* **of the column** *colname* **is not supported.**

Explanation: The SQL data type *sqltype* is not supported in the specified host language.

No declaration is generated for this column.

User Response: Ensure that this is the desired table.

SQL9320I **Connecting to the database** *database*...

Explanation: The utility is attempting to connect to the database *database*.

User Response: See the error message for details if it fails.

SQL9321I **Binding the utility automatically...**

Explanation: The utility is attempting to bind itself to the database.

User Response: See the error message for details if it fails.

SQL9322I **Retrieving column information from the table** *table*...

Explanation: The utility is attempting to retrieve column information from the table *table*.

User Response: See the error message for details if it fails.

SQL9323I **Generating declarations to the file** *filename*...

Explanation: The utility is attempting to generate declarations to the file *filename*.

User Response: See the error message for details if it fails.

SQL9324I **Disconnecting from the database** *database*...

Explanation: The utility is attempting to disconnect from the database *database*.

User Response: See the error message for details if it fails.

SQL10000 - SQL10099

SQL10002N **The password supplied is too long.**

Explanation: 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.

User Response: 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.**

Explanation: 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.

User Response: 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.

Explanation: 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.

This error is returned if a connection is attempted between 32-bit and 64-bit platforms. Connections between 32-bit and 64-bit platforms are not supported.

The command cannot be processed.

User Response: 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, EXCLUSIVE for exclusive use or EXCLUSIVE MODE ON SINGLE NODE for exclusive use on single node. For DB2 Connect connections, only SHARE mode is supported. EXCLUSIVE MODE ON SINGLE NODE is only supported in MPP configuration.**

Explanation: The *mode* parameter of the CONNECT TO statement must be SHARE for shared, EXCLUSIVE for exclusive use or EXCLUSIVE MODE ON SINGLE NODE for exclusive use on single node. If connecting to a

database using DB2 Connect, only shared access is allowed. EXCLUSIVE MODE ON SINGLE NODE is only supported in MPP configuration.

The command cannot be processed.

User Response: Resubmit the command with a valid *mode* parameter.

SQL10007N   Message *msgno* **could not be retrieved. Reason code:** *code*.

Explanation: 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. Either DB2CODEPAGE is set to a code page which the database does not support, or the client's locale is not supported by the database.

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.

User Response: 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.**

Explanation: 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.

User Response: 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.**

Explanation: The function routine within the library cannot be found.

The command cannot be processed.

User Response:

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*.

Explanation: When loading the DARI library, the command was interrupted, perhaps by the interrupt key (usually Ctrl+Break or Ctrl+C).

Processing is stopped.

User Response: Resubmit the command.

sqlcode: -10011

sqlstate: 42724

SQL10012N An unexpected operating system error was received while loading the specified library *name*.

Explanation: An unexpected error occurred when trying to load the library module specified in the program name field.

User Response: 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. Then contact IBM as specified in this guide.

sqlcode: -10012

sqlstate: 42724

SQL10013N The specified library *name* **could not be loaded.**

Explanation: The library module could not be found.

User Response: 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.

Federated system users: if the error message occurs as a result of issuing db2start on the federated system and the *name* is "from DB2_DJ_COMM" then the federated server experienced a problem loading one of the wrapper modules identified by the DB2_DJ_COMM environment variable. The DB2_DJ_COMM environment variable must be updated to contain only valid wrapper modules.

If the error message occurs as a result of issuing db2start on the federated system and the *name* is not "from DB2_DJ_COMM", the user must re-install the federated instance.

If the library identifies a wrapper module, that module must be installed, link-edited (as required), and available in the correct directory. Refer to the *Installation and Configuration Supplement* for more information about configuring wrapper modules.

sqlcode: -10013

sqlstate: 42724

SQL10014N **The invoking DARI program name** *name* **specified is not valid.**

Explanation: The DARI library module or the DARI program name syntax is not correct.

User Response: Ensure that the DARI library or program name is specified correctly.

sqlcode: -10014

sqlstate: 42724

SQL10015N **There are not enough system resources to load the specified library** *name***.**

Explanation: There is not enough memory to load the library module.

User Response: Stop the application. Possible solutions are:

- Remove background processes.
- Decrease the values of the configuration parameters that define the allocation of memory.
- Install more memory.

sqlcode: -10015

sqlstate: 42724

SQL10017N **The SQL CONNECT RESET statement is not allowed in a Database Application Remote Interface (DARI) procedure.**

Explanation: The remote procedure contains an SQL CONNECT RESET statement.

User Response: Remove the SQL CONNECT RESET statement and retry the remote procedure.

sqlcode: -10017

sqlstate: 38003

SQL10018N **The disk is full. Processing was ended.**

Explanation: The disk 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 file system, or the PC/IXF data file and the message file are on the same file system.

The EXPORT utility stops processing. The exported data is not complete.

User Response: Provide more space on the disk or specify that the data file be located on a different file system than the database or the message file and resubmit the command.

SQL10019N The database cannot be accessed in the specified path.

Explanation: 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.

User Response: 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.

Explanation: A write operation was attempted to a database residing on a file system which does not allow write access.

The statement cannot be processed.

User Response: Verify that the file permissions for the file system on which the database resides allow write access.

SQL20000 - SQL20099

SQL20005N The internal identifier limit of *limit* **has been exceeded for object type** *object-type***.**

Explanation: An internal identifier uniquely identifies objects of type *object-type*. The limit for internal identifiers for this type of object has been exceeded. This can happen during a CREATE DISTINCT TYPE, CREATE FUNCTION, CREATE PROCEDURE or CREATE SEQUENCE statement, or an ALTER TABLE or CREATE TABLE statement that defines an identity column.

The statement cannot be processed.

User Response: Drop objects of type *object-type* that are no longer being used.

sqlcode: -20005

sqlstate: 54035

SQL20010N Mutation method *method-ID* **is not allowed when the instance of the structured type is NULL.**

Explanation: The method *method-ID* is a mutator method that is specified with an instance of a structured type that is null. Mutator methods cannot be processed on a null instance. In some cases, the method name is not available.

The statement cannot be processed.

User Response: Determine the null instance that

is used with a mutator method. Use a constructor function to create a non-null instance of the structured type before using any mutator methods on the instance.

To determine the method name associated with the *method-ID*, use the following query:

```
SELECT FUNCSHEMA, FUNCNAME,
       SPECIFICNAME
 FROM SYSCAT.FUNCTIONS
 WHERE
   FUNCID = INTEGER(
           method-ID
           )
```

sqlcode: -20010

sqlstate: 2202D

SQL20011N Transform group *group-name* **is already defined for a subtype or supertype of data type** *type-name***.**

Explanation: A transform group called *group-name* already exists for a type in the same hierarchy as *type-name*. It may be defined for a supertype or subtype of *type-name*. A transform group name can only be used once within a structured type hierarchy.

The statement cannot be processed.

User Response: Change the name of the transform group.

SQL20012N **Type** *type-name* **does not have any associated transform groups to drop.**

Explanation: There are no transforms defined for *type-name*. There is nothing to drop.

The statement did not drop any transform groups.

User Response: Ensure the name of the type (including any required qualifiers) is correctly specified in the SQL statement and that the type exists.

sqlcode: -20012

sqlstate: 42740

SQL20013N **The object** *super-object-name* **is not valid as a supertype, supertable, or superview of object** *sub-object-name*.

Explanation: If the statement resulting in the error is creating a type, then *super-object-name* is a type that cannot be a supertype of *sub-object-name* because it is not a user-defined structured type.

If the statement resulting in the error is creating a table, then *super-object-name* is a table that cannot be a supertable of table *sub-object-name* because it is not defined as a typed table or the type of table *super-object-name* is not the direct supertype of the type used in defining table *sub-object-name*.

If the statement resulting in the error is creating a view, then *super-object-name* is a view that cannot be a superview of view *sub-object-name* because it is not defined as a typed view or the type of view *super-object-name* is not the direct supertype of the type used in defining view *sub-object-name*.

The statement cannot be processed.

User Response: Specify a valid type, table, or view in the UNDER clause of the CREATE statement.

sqlcode: -20013

sqlstate: 428DB

SQL20014N **The transform group** *group-name* *transform-type* **transform function for type** *type-name* **is not valid. Reason code =** *reason-code*.

Explanation: The *transform-type* transform function in transform group *group-name* is not valid. The reason is based on the *reason-code* as follows.

1	There can only be one parameter for a FROM SQL transform function.
2	The parameter of a FROM SQL transform function must be of type *type-name*.
3	The RETURNS data type of a TO SQL transform function must be of type *type-name*.
4	The RETURNS type of a FROM SQL transform function that returns a scalar must be a built-in data type other than DECIMAL.
5	All of the RETURNS types of a FROM SQL transform function that returns a scalar must be built-in data types other than DECIMAL.
6	There must be at least one parameter for a TO SQL transform function.
7	The parameter types for a TO SQL transform function must all be built-in data types other than DECIMAL.
8	The TO SQL transform function must be a scalar function.
9	The FROM SQL transform function must be written using LANGUAGE SQL, or must use another FROM SQL transform function that is written using LANGUAGE SQL.
10	The TO SQL transform function must be written using LANGUAGE SQL or use a TO SQL transform function that is written using LANGUAGE SQL.

The statement cannot be processed.

User Response:

1   Specify a FROM SQL transform function with a signature that has only one parameter.

2   Specify a FROM SQL transform function where the type of the parameter is the same as *type-name*.

3   Specify a TO SQL transform function where the RETURNS type is the same as *type-name*.

4   Specify a FROM SQL transform function with a RETURNS type that is a built-in data type other than DECIMAL.

5   Specify a FROM SQL transform function where each element of the row has a RETURNS type that is a built-in data type other than DECIMAL.

6   Specify a TO SQL transform function with a signature that has at least one parameter.

7   Specify a TO SQL transform function where all the parameter types are built-in data types other than DECIMAL.

8   Specify a TO SQL transform function that is a scalar function.

9   Specify a FROM SQL transform function that is written using LANGUAGE SQL, or that uses another FROM SQL transform function that is written using LANGUAGE SQL.

10   Specify a TO SQL transform function that is written using LANGUAGE SQL or that uses a TO SQL transform function that is written using LANGUAGE SQL.

sqlcode: -20014

sqlstate: 428DC

SQL20015N   A transform group *group-name* **is not defined for data type** *type-name*.

Explanation: A specified transform group *group-name* is not defined for the data type *type-name*. The data type *type-name* may have been explicitly specified in the statement or may be implicit based on the use of a structured type that requires the existence of the transform group for the data type.

If *group-name* is empty, either the TRANSFORM GROUP bind option or the CURRENT DEFAULT TRANSFORM GROUP special register was not specified and therefore there was no transform for *type-name*.

The statement cannot be processed.

User Response: Define transform group *transform-type* for data type *type-name* using the CREATE TRANSFORM statement. If the error occurred when dropping the transform, no action is required because the transform group did not exist for the data type.

If *group-name* is empty, specify the TRANSFORM GROUP bind option on the CURRENT DEFAULT TRANSFORM GROUP special register.

sqlcode: -20015

sqlstate: 42741

SQL20016N   The value of the inline length associated with type or column *type-or-column-name* **is too small.**

Explanation: For the definition of structured type *type-or-column-name*, it has an INLINE LENGTH value specified that is smaller than the size returned by the constructor function (32 + 10 * number_of_attributes) for the type and is less than 292. For the altering of column *type-or-column-name*, the INLINE LENGTH specified is smaller than the current inline length.

The statement cannot be processed.

User Response: Specify an INLINE LENGTH value that is large enough. For a structured type, that is at least the size returned by the constructor function for the type or is at least

292. For a column, it must be larger than the current inline length. If this error occurs when altering the type (or some supertype of this type) to add an attribute, either the attribute cannot be added or the type must be dropped and re-created with a larger INLINE LENGTH value.

sqlcode: -20016

sqlstate: 429B2

SQL20017N Addition of this subtype exceeds the maximum number of levels of a type hierarchy.

Explanation: The maximum number of levels in a type hierarchy is 99. Adding this type would exceed the maximum.

The statement cannot be processed.

User Response: Do not add any more subtypes to this type hierarchy.

sqlcode: -20017

sqlstate: 54045

SQL20018N Row function *function-name* **must return at most one row.**

Explanation: The function is a defined to return a single row. The result of the processing the function is more than one row.

User Response: Ensure that the function is defined in such a way that at most one row is returned.

sqlcode: -20018

sqlstate: 21505

SQL20019N The result type returned from the function body cannot be assigned to the data type defined in the RETURNS clause.

Explanation: The data type of each column returned by the function body must be assignable to the corresponding column specified in the RETURNS clause. If the function is a scalar function, there is only one column.

User Response: Change the RETURNS type or the type returned from the function body so that the data types of corresponding columns are assignable.

sqlcode: -20019

sqlstate: 42866

SQL20020N Operation *operation-type* **is not valid for typed tables.**

Explanation: The operation identified by *operation-type* cannot be performed on a typed tables.

The statement cannot be processed.

User Response: Remove the ADD COLUMN clause or SET DATATYPE clause from the ALTER statement. Columns can only be added by re-defining the table with a structured type that includes the new column as an attribute. Similarly, the data type of a column can only be changed by re-defining the table with a type that includes the column with a different data type.

sqlcode: -20020

sqlstate: 428DH

SQL20021N Inherited column or attribute *name* **cannot be changed or dropped.**

Explanation: The value of *name* identifies either a column name or an attribute name (depending on the statement context) that is inherited from a typed table, typed view, or structured type above it in the table, view, or type hierarchy.

- In the CREATE TABLE statement, the WITH OPTIONS clause cannot be specified for column *name* in the CREATE TABLE statement because it is inherited from a supertable in the table hierarchy.
- In the ALTER TABLE statement, the SET SCOPE clause or COMPRESS clause cannot be specified for column *name* because it is inherited from a supertable in the table hierarchy.
- In the CREATE VIEW statement, the WITH OPTIONS clause cannot be specified for

column *name* in the CREATE VIEW statement because it is inherited from a superview in the view hierarchy.

- In the ALTER TYPE statement, the DROP ATTRIBUTE clause cannot be specified for attribute *name* because it is inherited from a supertype in the type hierarchy.

The statement cannot be processed.

User Response: The options for a column may only be set or altered for the table or view in the typed table hierarchy or typed view hierarchy where the column is introduced. An attribute can only be dropped from the data type in the type hierarchy where the attribute was introduced.

sqlcode: -20021

sqlstate: 428DJ

SQL20022N **The scope for the reference column** *column-name* **is already defined.**

Explanation: The scope for the reference column *column-name* cannot be added because it is already defined.

The statement cannot be processed.

User Response: Remove the ADD SCOPE clause from the ALTER TABLE statement.

sqlcode: -20022

sqlstate: 428DK

SQL20023N **Parameter** *parm-number* **of the external or sourced function has a scope defined.**

Explanation: A reference type parameter should not have a scope defined when used with external or sourced user defined functions.

The statement cannot be processed.

User Response: Remove the SCOPE specification from the definition of the parameter.

sqlcode: -20023

sqlstate: 428DL

SQL20024N **The scope table or view** *target-name* **is not defined with structured type** *type-name***.**

Explanation: The scope table or view *target-name* is not valid for use as the scope of this reference because it is:

- not a typed table or
- not a typed view or
- the type of the table or view is not the same as the target type of the REF type.

The statement cannot be processed.

User Response: Specify the scope of the reference using a typed table or typed view with the same type as the target type of the REF type.

sqlcode: -20024

sqlstate: 428DM

SQL20025N **SCOPE is not specified in the RETURNS clause of an external function or is specified in the RETURNS clause of a sourced function.**

Explanation: There are two possible causes:

- A reference type must have a scope defined when used as the result of a user-defined external function.
- A reference type cannot have a scope defined when used as the result of an user-defined sourced function. The function will use the scope of the source function.

The statement cannot be processed.

User Response: When defining an external function with a reference type as a returns type, ensure that the SCOPE clause is specified. When defining SOURCED function with a reference type as a returns type, ensure that the SCOPE clause is not specified.

sqlcode: -20025

sqlstate: 428DN

SQL20026N   The type *type-name* **is not a structured type or is not an instantiable structured type.**

Explanation: The statement requires a structured type that can be instantiated. The type *type-name* is either:

- not a structured type; or
- a structured type that is defined as not instantiable.

The statement cannot be processed.

User Response: Verify that the correct type name is used in the statement.

sqlcode: -20026

sqlstate: 428DP

SQL20027N   The subtable or subview *sub-object-name* **was not created because subtable or subview** *object-name* **with type** *type-name* **already exists.**

Explanation: Within a typed table or view hierarchy, only one subtable or subview may exist of a particular subtype. The table or view *sub-object-name* cannot be created since there is already a table or view of type *type-name* defined. The table or view that already exists is *object-name*.

The statement cannot be processed.

User Response: Verify that the subtable or subview is being created with the correct type and that the subtable is being created under the correct supertable or the subview is being created under the correct superview.

sqlcode: -20027

sqlstate: 42742

SQL20028N   The table or view *table-name* **cannot have a different schema name from the other tables or views in the same hierarchy.**

Explanation: All tables in a typed table hierarchy must have the same schema name and

all views in a typed view hierarchy must have the same schema name.

The statement cannot be processed.

User Response: Verify that the schema name of the table or view is correct. If a hierarchy name is specified, verify that its schema name matches the schema name of the root table or view. If a subtable is being created, verify that it is being created under the correct supertable. If a subview is being created, verify that it is being created under the correct superview.

sqlcode: -20028

sqlstate: 428DQ

SQL20029N   *operation* cannot be applied to a subtable.

Explanation: The operation *operation* was applied to a table that is not the root of a table hierarchy.

The statement cannot be processed.

User Response: Specify the root table of the table hierarchy in the operation.

sqlcode: -20029

sqlstate: 428DR

SQL20030N   Attributes of a structured type *type-name* **cannot be added or dropped when a typed table, typed view, or index extension** *object-name* **is dependent on the type.**

Explanation: The attributes of a structured type cannot be added or dropped if a typed table or typed view of the structured type or any of its subtypes currently exists. Attributes of a structured type also cannot be added or dropped when a column exists in the table that directly or indirectly uses *type-name*. Also, attributes of a structured type cannot be added or dropped when the type *type-name* or one of its subtypes is used in an index extension. The table, view, or index extension *object-name* is one table, view, or index extension that is dependent on the structured type *type-name*. There may be other

tables, views, or index extensions that are dependent on the type or one of its proper subtypes.

The statement cannot be processed.

User Response: Verify that the right type is being altered or drop any tables, views, and index extensions that are dependent on the structured type *type-name*.

sqlcode: -20030

sqlstate: 55043

SQL20031N *Object* **may not be defined on a subtable.**

Explanation: Primary key and unique constraints can only be defined on the root table of a typed table hierarchy. Similarly, unique indexes can only be defined on the root table of a typed table hierarchy.

The statement cannot be processed.

User Response: Unique indexes, Primary key or unique constraints can only be defined on the root table of a table hierarchy.

sqlcode: -20031

sqlstate: 429B3

SQL20032N **Index on the specified columns cannot be defined on subtable** *table-name***.**

Explanation: The columns specified for the index were all introduced at a higher level in the typed table hierarchy than subtable *table-name*. Therefore, the index cannot be created on this subtable.

The statement cannot be processed.

User Response: Determine the table in the table hierarchy where all of the columns are included for the first time. Use this as the table name when creating the index.

sqlcode: -20032

sqlstate: 428DS

SQL20033N **An expression including** *partial-expression* **does not include a valid scoped reference.**

Explanation: The expression that includes *partial-expression* requires an operand that is reference type with a scope defined. If the expression includes the DEREF function, the argument of the function must be a reference type with a scope defined.

For the dereference operator (->), the left operand needs to be a reference type with a scope defined.

The statement cannot be processed.

User Response: Correct the SQL statement syntax so the operand or argument is a reference type with a scope defined.

sqlcode: -20033

sqlstate: 428DT

SQL20034N **The data type** *list-type-name* **is not included in a structured data type hierarchy that includes the data type** *left-type-name* **of the left operand of the TYPE predicate.**

Explanation: All the data types listed in the TYPE predicate must be included in a data type hierarchy that includes the data type of the left operand of the TYPE predicate. The data type *left-type-name* is not a structured data type (not part of any type hierarchy) or the data type *list-type-name* is not included in the data type hierarchy that includes *left-type-name*.

The statement cannot be processed.

User Response: Ensure that the data type of the expression and all listed data types in the TYPE predicate are structured data types within the same data type hierarchy. If *left-type-name* is SYSIBM.REFERENCE, use DEREF to make the result data type of the expression a structured data type.

sqlcode: -20034

sqlstate: 428DU

SQL20035 **Invalid left operand of a dereference operator. Path expression starts with** *expression-string***.**

Explanation: The left operand of the dereference operator in a path expression is not valid. Possible causes are:

- The left operand includes a column function that uses a column function as an argument.
- The left operand expression includes a column function and a reference to a column that is not in the GROUP BY clause.

User Response: Correct the left operand of the dereference operator for the path expression that starts with *expression-string*.

sqlcode: -20035

sqlstate: 428DV

SQL20036N **The object identifier column** *column-name* **cannot be referenced using the dereference operator.**

Explanation: The dereference operator is used with *column-name* as the right operand. This column is the object identifier column of the target table of the dereference and is not valid for this operator.

The statement cannot be processed.

User Response: Correct the name of the column in the dereference operation.

sqlcode: -20036

sqlstate: 428DW

SQL20037N **Object identifier column is required to create the root table or root view** *object-name* **of a typed table or view hierarchy.**

Explanation: When creating the root table of a typed table hierarchy, an object identifier (OID) column must be defined (using the REF IS clause) on the CREATE TABLE statement.

When creating the root view of a typed view hierarchy, an object identifier (OID) column must

be defined (using the REF IS clause) on the CREATE VIEW statement.

The statement cannot be processed.

User Response: Add the required OID column (REF IS clause) to the CREATE TABLE or CREATE VIEW statement.

sqlcode: -20037

sqlstate: 428DX

SQL20038N **The** *keywords* **clause cannot be specified with the EXTEND USING clause.**

Explanation: The EXTEND USING clause in the CREATE INDEX statement cannot be specified with the *keywords* clause.

The statement cannot be processed.

User Response: Remove the specification of *keywords* clause or the EXTEND USING clause from the CREATE INDEX statement.

sqlcode: -20038

sqlstate: 42613

SQL20039N **The definition of index** *index-name* **does not match the definition of index extension** *index-ext-name***.**

Explanation: The index definition and the index extension definition do not match. The following lists the possibilities for where the definitions do not match.

- The number of arguments following the index extension name in the EXTEND USING clause is not the same as the number of instance parameters of the index extension.
- The data types of the arguments following the index extension name in the EXTEND USING clause do not match exactly (including length or precision and scale) the data types of the corresponding instance parameters of the index extension.
- The number of columns specified for the index is not same as the number of source key parameters of the index extension.

- The data types of the index columns do not match exactly (including length or precision and scale) the data types of the corresponding source key parameters of the index extension. There is an exception to exact matching of data types for the case of subtypes. The column of the index may be a subtype of the data type specified as the corresponding source key parameter.

The statement cannot be processed.

User Response: Change the index definition so that the index extension matches.

sqlcode: -20039

sqlstate: 428E0

SQL20040N **Number or type of the result of the range-producing table function** *range-function-name*, **is inconsistent with that of key transformation table function** *transform-function-name* **for index extension** *index-ext-name*.

Explanation: The range-producing function must:
- return up to twice as many columns as returned by the key transformation function
- have an even number of columns (the first half of the return columns are the start key values and the second half of the return columns are the stop key values)
- have the each start key column with the same type as the corresponding stop key column
- have the type of each start key column be the same as the corresponding transformation function column.

More precisely, let $a_1{:}t_1,\dots\ a_n{:}t_n$ be the function result columns and data types of the key transformation function. The function result columns of the range-producing function must be $b_1{:}t_1,\dots,b_m{:}t_m,c_1{:}t_1,\dots,c_m{:}t_m$, where m <= n and the "b" columns are the start key columns and the "c" columns are the stop key columns.

The statement cannot be processed.

User Response: Specify a range-producing table function that is consistent with the key transformation table function.

sqlcode: -20040

sqlstate: 428E1

SQL20041N **Number or the type of target key parameters does not match with the number or type of key transform function** *function-name* **for index extension** *index-ext-name*.

Explanation: The number of target key parameters has to match with the number of results returned by the key transform function. In addition, the type of the target key parameters must exactly match the corresponding function result types.

The statement cannot be processed.

User Response: Specify the correct number and type of parameters as the target key parameters.

sqlcode: -20041

sqlstate: 428E2

SQL20042N **The maximum allowable** *parm-type* **parameters is exceeded in index extension** *index-ext-name*. **The maximum is** *max-value*.

Explanation: Too many parameters were specified. If *parm-type* is INDEX EXTENSION, then up to *max-value* instance parameters may be specified. If *parm-type* is INDEX KEYS, then up to *max-value* key source parameters may be specified.

The statement cannot be processed.

User Response: Specify no more than the maximum number of parameters.

sqlcode: -20042

sqlstate: 54046

SQL20043N Argument for routine *routine-name* **is not valid. Reason code =** *reason-code***.**

Explanation: The routine *routine-name* could be the transformation function, the range-producing function, or a routine (function or method) referenced in the FILTER USING clause. The reason code indicates why an argument is not valid:

1   For a key transformation function, the argument is not an observer method or an index extension instance parameter.

2   The expression used as the argument uses a routine that specifies LANGUAGE SQL.

3   The expression used as the argument is a subquery.

4   The data type of an expression used as the argument cannot be a structured type.

5   The argument of a key transformation function cannot have a data type of structured data type, LOB, DATALINK, LONG VARCHAR, or LONG VARGRAPHIC.

The statement cannot be processed.

User Response: Specify a valid argument for the function.

sqlcode: -20043

sqlstate: 428E3

SQL20044N The routine *routine-name* **or the CASE expression is not valid in a CREATE INDEX EXTENSION or CREATE FUNCTION statement. Reason code =** *reason-code***.**

Explanation: A routine (function or method) *routine-name* used in the CREATE INDEX EXTENSION or CREATE FUNCTION statement is not valid where it is used. If *routine-name* is empty, then the CASE expression used for filtering is not valid. The reason code indicates the reason.

1   The key transformation function is not a table function.

2   The key transformation function is not an external function.

3   The key transformation function is a variant function.

4   The key transformation function is an external action function.

5   The range-producing function is not a table function.

6   The range-producing function is not an external function.

7   The range-producing function is a variant function.

8   The range-producing function is an external action function.

9   The index-filter function is not an external function.

10   The index-filter function is a variant function.

11   The index-filter function is an external action function.

12   The result type of a filter function or CASE expression is not an integer data type.

13   A subquery is used in a CASE expression or as an argument of a filter function.

The statement cannot be processed.

User Response: If *routine-name* is not empty, specify a routine that conforms to the rules for a function or method specified in the specific clause of the CREATE INDEX EXTENSION or CREATE FUNCTION statement. Otherwise, specify a CASE expression that conforms to the rules for a CASE expression in the FILTER USING clause.

sqlcode: -20044

sqlstate: 428E4

SQL20045N The data type of instance parameter *parameter-name* **is invalid in index extension** *index-ext-name*.

Explanation: An instance parameter must be one of the following data types: VARCHAR, VARGRAPHIC, INTEGER, DECIMAL, or DOUBLE.

The statement cannot be processed.

User Response: Specify a valid data type for the instance parameter *parameter-name*.

sqlcode: -20045

sqlstate: 429B5

SQL20046N SELECTIVITY clause following *predicate-string* **can only be specified for a valid user-defined predicate.**

Explanation: The SELECTIVITY clause is specified with a predicate that does not include a valid user-defined function. A valid user-defined function includes a PREDICATES clause with a WHEN clause that matches the predicate. The SELECTIVITY clause cannot be specified except with a user-defined predicate.

The statement cannot be processed.

User Response: Remove the SELECTIVITY clause following the predicate.

sqlcode: -20046

sqlstate: 428E5

SQL20047N The search method *method-name* **is not found in the index extension** *index-ext-name*.

Explanation: The method *method-name* referenced in an exploitation rule of the user-defined predicate has to match one of the search methods specified in the index extension *index-ext-name*.

The statement cannot be processed.

User Response: Specify a method defined in the index extension.

sqlcode: -20047

sqlstate: 42743

SQL20048N The search argument of method *method-name* **does not match the one in the corresponding search method in the index extension** *index-ext-name*.

Explanation: The search argument provided for method *method-name* does not match the argument of the corresponding search method in the index extension *index-ext-name*. Either the number of arguments or the type of the arguments do not match the defined number or type of the parameters.

The statement cannot be processed.

User Response: Specify a search argument that matches the parameters defined in the index extension.

sqlcode: -20048

sqlstate: 428E6

SQL20049N Type of an operand following the comparison operator in the AS PREDICATE WHEN clause does not exactly match the RETURNS type.

Explanation: The definition of the user-defined predicate is not valid. In the AS PREDICATE WHEN clause, the type of the operand following the the comparison operator is not an exact match with the RETURNS type of the function.

The statement cannot be processed.

User Response: Specify an the operand with the correct data type.

sqlcode: -20049

sqlstate: 428E7

SQL20050N The search target or search argument *parameter-name* **does not match a name in the function being created.**

Explanation: Each search target in an index exploitation rule has to match some parameter name of the function that is being created. Each search argument in an index exploitation rule must match either an expression name in the EXPRESSION AS clause or a parameter name of the function being created. Parameter names must be specified in the parameter list for the function.

The statement cannot be processed.

User Response: Specify only valid names of the function in the search target or search argument.

sqlcode: -20050

sqlstate: 428E8

SQL20051N The argument *parameter-name* **cannot appear as both a search target and a search argument in the same exploitation rule.**

Explanation: In the exploitation clause, a function parameter cannot be specified as an argument following KEY and as an argument of the method specified following the USE keyword.

The statement cannot be processed.

User Response: Specify a parameter name of the function in no more than on of the search target or the search argument.

sqlcode: -20051

sqlstate: 428E9

SQL20052N Column *column-name* **is an object identifier column which cannot be updated.**

Explanation: The UPDATE statement includes setting a column that is an object identifier (OID) column. An OID column cannot be updated.

The statement cannot be processed.

User Response: Remove the SET of *column-name* from the UPDATE statement.

sqlcode: -20052

sqlstate: 428DZ

SQL20053N Fullselect in typed view *view-name* **is not valid. Reason code =** *reason-code***.**

Explanation: The fullselect specified in the definition of typed view *view-name* does not conform to the rules for the fullselect of a typed view. Possible reason codes follow.

1   A subview definition contains a common table expression, OR a branch in a subview definition does not range over a single table, view, nickname or an alias.

2   The row-set of a branch over a table-hierarchy cannot be proven by the database manager to be distinct from the row-set of all branches over the same table-hierarchy in the rest of the typed view hierarchy.

3   The first expression of a branch over a hierarchy in the root view is:
   - not the object identifier column of the typed table or typed view referenced in the FROM clause and the UNCHECKED option in the REF IS clause is not used OR
   - if the table in the FROM clause is not typed, the column is nullable or does not have a unique index defined on only that column and the UNCHECKED option in the REF IS clause is not used OR
   - not identical to the expression in the branch over the same hierarchy in the subview.

4   A table or view over which a branch of a subview ranges is not a subtable or subview of the table or view referenced in any branch of the superview and either the subview uses EXTEND AS or

the root view does not have the UNCHECKED option in the REF IS clause turned on.

5 The fullselect includes references to the NODENUMBER or PARTITION functions, non-deterministic functions, or functions defined to have external action.

6 A branch in a subview cannot range over an OUTER table or view if any branch in its superview ranges over a table or view in the same hierarchy without using OUTER.

7 A subview ranges over a view in its own view hierachy.

8 A subview uses a set operation other than UNION ALL in its definition OR UNION ALL is used in the definition without specifying the UNCHECKED option in the REF IS clause of the root view.

9 A subview contains two branches of a UNION ALL that range over the same table hierarchy or view hierarchy.

10 The subview definition contains a GROUP BY or HAVING clause.

User Response: Change the fullselect of the view definition based on the *reason-code*.

1 Use only one table or view in the FROM clause. Encapsulate complex selects into views that the typed view can range over using the UNCHECKED option of the root view's REF IS clause.

2 Specify a different table or view in the FROM clause of each branch that is not the same as one already used in the view hierarchy or use a predicates that clearly define the row-set for each branch as distinct compared to the row-sets for other branches in the typed view hierarchy.

3 Ensure that the first column of the root view conforms to the rules to be a valid object identifier column for the typed

view. Consider using the UNCHECKED option in the ref is clause.

4 Specify a subtable or subview of the table or view specified in the FROM clause of a branch of the superview. Or use the UNCHECKED option in the root view definition in conjunction with the AS (without EXTEND) clause in the subview definition.

5 Remove the reference to the function from the fullselect.

6 If this is the first subview to use OUTER in a branch over this hierarchy, change the FROM clause so that OUTER is not used. If the superview uses OUTER, include OUTER in the FROM clause of the subview.

7 Do not source a subview on other views in the same hierarchy.

8 If UNION ALL was used, use the UNCHECKED option in the REF IS clause of the rootview to allow multiple branches in the subview definition. For other set operations encapsulate the set operation into a view and use the UNCHECKED option in the subview to allow sourcing on common views.

9 Unify the branches to a select over their common supertable or superview and use predicates (e.g. the type predicate) to filter for the desired rows.

10 Encapsulate the GROUP BY and HAVING clause into a view and use the UNCHECKED option in the root view to allow sourcing on common views.

sqlcode: -20053

sqlstate: 428EA

SQL20054N **The table** *table-name* **is in an invalid state for the operation. Reason code =** *reason-code***.**

Explanation: The table is in a state that does not allow the operation. The reason code

indicates the state of the table that prevents the operation.

21 The table is in Datalink Reconcile Pending (DRP) state or Datalink Reconcile Not Possible (DRNP) state.

22 The function cannot be used in a generated column.

The statement cannot be processed.

User Response: The action is based on the reason code as follows.

21 Refer to the Administration Guide for information on Datalink Reconcile Pending (DRP) and Datalink Reconcile Not Possible (DRNP) states to take appropriate action.

22 Use SET INTEGRITY FOR <table-name> OFF before altering the table. Then alter the table and use SET INTEGRITY FOR <table-name> IMMEDIATE CHECKED FORCE GENERATED to generate the values for the new or altered column.

sqlcode: -20054

sqlstate: 55019

SQL20055N **A result column data type in the select list is not compatible with the defined data type for column** *column-name***.**

Explanation: The data type of the select list expression corresponding to *column-name* is incompatible with the data type for the attribute of the structured type. Both must be:

- Numeric
- Character
- Graphic
- Dates or character
- Times or character
- Timestamps or character
- Datalinks
- the same distinct types

- reference types where the target type of the select list expression is a subtype of the target type of the attribute.

The statement cannot be processed.

User Response: Examine the current definition for the table and the associated structured type. Ensure the select list expression data type for the specified column is a compatible type.

sqlcode: -20055

sqlstate: 42854

SQL20056N **Processing on DB2 Data Links Manager** *name* **encountered an error. Reason code =** *reason-code***.**

Explanation: The processing on the DB2 Data Links Manager for the statement encountered an error as indicated by the following reason codes.

01 An inconsistency was detected between the data on the DB2 Data Links Manager and a DATALINK value in a table.

02 The DB2 Data Links Manager reached a resource limit during the processing.

03 The DB2 Data Links Manager does not support file pathnames longer than 128 characters.

99 The DB2 Data Links Manager encountered an internal processing error.

The statement cannot be processed.

User Response: The action is based on the reason code as follows.

01 Run the reconcile utility on the table.

02 The DB2 Data Links Manager administrator should identify the resource from the diagnostic logs and take corrective action.

03 You should ensure that the file pathname (excluding the file system prefix) to be stored in the DATALINK column does not exceed 128 characters.

For example, in the URL "http://server.com/dlfiles/dir1/.../file1" -- assuming the DLFS file system prefix is "/dlfiles" -- the file pathname "/dir1/.../file1" must not exceed 128 characters.

99   Save the diagnostic logs from the DB2 Data Links Manager and the database manager and contact IBM service.

sqlcode: -20056

sqlstate: 58004

SQL20057N   Column *column-name* in subview *view-name* cannot be defined as read only when the corresponding column is updatable in the superview.

Explanation: The column identified by *column-name* in the subview *view-name* is defined (implicitly) as read only. The superview of *view-name* includes the corresponding column that is updatable. A column cannot be changed from updatable to read only in a typed view hierarchy.

The statement cannot be processed.

User Response: Change the CREATE VIEW statement so that the column of the subview *view-name* is updatable or drop the superview(s) and recreate them using the READ ONLY clause to force the column to be read only.

sqlcode: -20057

sqlstate: 428EB

SQL20058N   The fullselect specified for the materialized query table *table-name* is not valid.

Explanation: The materialized query table definition has specific rules regarding the contents of the fullselect. Some rules are based on the materialized query table options (REFRESH DEFERRED or REFRESH IMMEDIATE) while others are based on whether or not the table is replicated. The fullselect in the

CREATE TABLE statement that returned this condition violates at least one of the rules as described in the SQL Reference.

The statement cannot be processed.

User Response: Change the fullselect in the CREATE TABLE statement so that it conforms to the rules based on the materialized query table options and whether or not the materialized query table is replicated.

sqlcode: -20058

sqlstate: 428EC

SQL20059W   The materialized query table *table-name* may not be used to optimize the processing of queries.

Explanation: The materialized query table is defined with REFRESH DEFERRED and a fullselect that is currently not supported by the database manager when optimizing the processing of queries. The rules are based on the materialized query table options (REFRESH DEFERRED or REFRESH IMMEDIATE). The fullselect in the CREATE TABLE statement that returned this condition violates at least one of the rules as described in the SQL Reference.

The materialized query table is successfully created.

User Response: No action is required. If the materialized query table was intended for use as a summary table to optimize the processing of queries, redefine the fullselect to ensure that it is a subselect that includes a GROUP BY clause.

sqlcode: +20059

sqlstate: 01633

SQL20060N   The key transform table function used by the index extension of index *index-id* of table *table-id* in *tbspace-id* generated duplicate rows.

Explanation: The key transform table function specified by the GENERATE USING clause of the index extension used by index *index-id* generated

duplicate rows. For a given invocation of the key transform table function, no duplicate rows should be produced. This error is occurred when inserting or updating the key value for the index *index-id* of table *table-id* in table space *tbspace-id*.

The statement cannot be processed.

User Response: The code for the key transform table function used by the index extension of index *index-id* must be modified to avoid the creation of duplicate rows.

To determine the index name, use the following query:

```
SELECT IID, INDSCHEMA, INDNAME
  FROM SYSCAT.INDEXES AS I,
       SYSCAT.TABLES AS T
  WHERE IID = <index-id>
      AND TABLEID = <table-id>
      AND TBSPACEID = <tbspace-id>
      AND T.TBASCHEMA = I.TABSCHEMA
      AND T.TABNAME = I.TABNAME
```

sqlcode: -20060

sqlstate: 22526

SQL20062N Transform function *transform-type* in transform group *group-name* for type *type-name* cannot be used for a function or method.

Explanation: The transform function defined in the transform group *group-name* for type *type-name* cannot be used for a function or method because the transform function is not written in SQL (defined with LANGUAGE SQL). The transform group cannot be used for this function or method.

The statement cannot be processed.

User Response: Specify a transform group for type *type-name* that has the transform functions defined with LANGUAGE SQL.

sqlcode: -20062

sqlstate: 428EL

SQL20063N TRANSFORM GROUP clause must be specified for type *type-name*.

Explanation: The function or method includes a parameter or returns data type of *type-name* that does not have a transform group specified.

The statement cannot be processed.

User Response: Specify a TRANSFORM GROUP clause with a transform group name that is defined for *type-name*.

sqlcode: -20063

sqlstate: 428EM

SQL20064N Transform group *group-name* does not support any data type specified as a parameter or returned data type.

Explanation: The transform group *group-name* specified in the TRANSFORM GROUP clause is not defined for any data type that is included in the parameter list or the RETURNS clause of a function or method.

The statement cannot be processed.

User Response: Remove the transform group from the function or method definition.

sqlcode: -20064

sqlstate: 428EN

SQL20065N Transform group *group-name* for data type *type-name* cannot be used to transform a structured type for use with a client application.

Explanation: The transform group *group-name* for data type *type-name* defines a transform function that cannot be used when performing transforms for client applications. The possible reasons for this are based on the definition of the transform function that is not supported for client applications. The unsupported transform function may be:

• a FROM SQL function that is a ROW function

- a TO SQL function has more than one parameter

The statement cannot be processed.

User Response: For static embedded SQL, specify a different transform group using the TRANSFORM GROUP bind option. For dynamic SQL, specify a different transform group using the SET DEFAULT TRANSFORM GROUP statement.

sqlcode: -20065

SQL20066N **The** *transform-type* **transform function is not defined in the transform group** *group-name* **for data type** *type-name*.

Explanation: The *transform-type* transform function of transform group *group-name* for data type *type-name* is required for a transform group used in a function or method definition.

The statement cannot be processed.

User Response: If creating a function or method, specify a different transform group in the function or method definition. If referencing a structured type in a dynamic SQL statement, specify a different transform group for the CURRENT DEFAULT TRANSFORM GROUP special register. Alternatively, add a *transform-type* transform function to the transform group *group-name* for data type *type-name*.

sqlcode: -20066

sqlstate: 42744

SQL20067N **The** *transform-type* **transform function is defined more than once in the transform group** *group-name* **for data type** *type-name*.

Explanation: A TO SQL or FROM SQL transform function can only be specified once in a transform group. The transform group *group-name* for data type *type-name* has at least two FROM SQL or TO SQL (or both) transform functions defined.

The statement cannot be processed.

User Response: Delete the TO SQL or FROM SQL definitions from *group-name* in the transform definition so that there is only one of each.

sqlcode: -20067

sqlstate: 42628

SQL20068N **The structured type** *type-name* **may not be defined so that one of its attribute types directly or indirectly uses itself. The attribute** *attribute-name* **causes the direct or indirect use.**

Explanation: Direct use: Type A is said to directly use type B if one of the following is true:
- Type A has an attribute of type B
- Type B is a subtype of A, or a supertype of A.

Indirect use: Type A is said to indirectly use type B if the following is true:
- Type A uses type C, and type C uses type B.

You cannot define a type so that one of its attribute types directly or indirectly uses itself. The type for attribute *attribute-name* is the cause of the direct or indirect use.

User Response: Evaluate the type and remove the attribute type that causes the direct or indirect use.

sqlcode: -20068

sqlstate: 428EP

SQL20069N **The RETURNS type of the** *routine-type routine-name* **is not the same as the subject type.**

Explanation: The method *method-name* specifies SELF AS RESULT. The RETURNS data type of the method must be the same as the subject data type of the method.

User Response: Change the RETURNS type of the method *method-name* to match the subject type.

sqlcode: -20069

sqlstate: 428EQ

SQL20075N The index or index extension
index-name **cannot be created or altered because the length of** *column-name* **is more than 255 bytes.**

Explanation: The index could not be created or altered because the key column length is greater than 255.

- *index-name* is the index name.
- *column-name* is the name of the key column. If this error was returned from an ALTER TABLE operation, then the value of *column-name* is the column number.

The index extension could not be created because a column returned by the GENERATE KEY function is greater than 255 bytes.

- *index-name* is the index extension name.
- *column-name* is the name of a column returned by the GENERATE KEY function.

The statement could not be processed. The specified index or index extension was not created, or the table could not be altered.

User Response: If creating an index, eliminate the column from the index definition. If altering a table, reduce the new column length to the permitted maximum. If creating an index extension, specify a different GENERATE KEY function, or redefine the function to eliminate the column.

sqlcode: -20075

sqlstate: 54008

SQL20076N The instance for the database is not enabled for the specified action or operation. Reason code = *reason-code***.**

Explanation: An error has been detected at the instance level. The requested operation cannot be completed because a specified functional area was not installed or because a specified functional area was not enabled to the instance.

The following is the list of reason codes and

associated functional areas you can enable at the instance level:

1. The ability to execute distributed request operations against multiple data sources in a single statement.

User Response: Enable the instance for the requested action or operation. First, install the specified functional area if it is missing. Then, enable the specified functional area. Enablement steps vary by *reason-code*:

1. Set the DBM variable <DREQ> to ON and then restart the database manager. For a federated server, set the DBM variable <FEDERATED> to YES and then restart the database manager.

sqlcode: -20076

sqlstate: 0A502

SQL20077N Cannot construct structured type objects that have Datalink type attributes.

Explanation: An attempt was made to invoke the constructor of a structured type which has a Datalink and/or a Reference type attribute. This functionality is currently not supported. In Version 6.1 or earlier, this error may also be issued for a structured type object with a Reference type attribute.

The statement cannot be processed.

User Response: The error can be corrected by doing one of the following:

1. Removing the invocation of the constructor of the type from the program.
2. Removing any Datalink (or Reference) type attributes from the definition of the structured type (this may not be possible if there are any tables that depend on this type).

sqlcode: -20077

sqlstate: 428ED

SQL20078N **Hierarchy object** *object-name* **of type** *object-type* **can not be processed using operation** *operation-type***.**

Explanation: The operation *operation-type* was attempted using a hierarchy object named *object-name* of type *object-type*. This operation does not support processing of a hierarchy object.

The statement could not be processed.

User Response: Verify that the correct object name was used. For objects types TABLE or VIEW, the object must be the name of a subtable in the table or view hierarchy. In some cases, the object must specifically name the root table. For objects of type index, the name must be a name of an index created on a subtable.

sqlcode: -20078

sqlstate: 42858

SQL20080N **Method specification for** *method-name* **cannot be dropped because a method body exists.**

Explanation: The method specification *method-name* still has an existing method body that must be dropped before the method specification can be dropped.

The statement cannot be processed.

User Response: Use the DROP METHOD statement with the same method specification to drop the method body and then issue the ALTER TYPE statement again to drop the method specification.

sqlcode: -20080

sqlstate: 428ER

SQL20081N **The method body cannot be defined for a LANGUAGE** *language-type* **method specification** *method-name***.**

Explanation: The method specification*method-name* is defined with LANGUAGE *language-type*. If the LANGUAGE is SQL, the method body

must be an SQL control statement. For other languages, the EXTERNAL clause must be specified.

The statement cannot be processed.

User Response: Change the method body to match the LANGUAGE specified in the method specification.

sqlcode: -20081

sqlstate: 428ES

SQL20082N **The dynamic type** *expression-type-id* **of the expression is not a subtype of the target data type** *target-type-id* **in a TREAT specification.**

Explanation: The dynamic data type of the result of the expression specified in the TREAT specification is *expression-type-id*. The specified target data type *target-type-id* is a proper subtype of *expression-type-id* which is not allowed.

The statement cannot be processed.

User Response: Change the *target-type-id* of the TREAT specification to a supertype of *expression-type-id* or change the expression so that the dynamic data type of the result is a subtype of *target-type-id*.

To determine the data type names for *expression-type-id* and *target-type-id*, use the following query:

```
SELECT TYPEID, TYPESCHEMA, TYPENAME
FROM SYSCAT.DATATYPES
WHERE TYPEID IN INTEGER(
  expression-type-id),
  INTEGER(
    target-type-id
  )
)
```

sqlcode: -20082

sqlstate: 0D000

SQL20083N **The data type of the value returned by** *routine-type routine-id* **does not match the data type specified as RESULT.**

Explanation: The method *routine-id* specifies SELF AS RESULT and therefore requires that the data type of the value returned must be the same as the subject data type used to invoke the method. The RETURN statement in either the SQL method body or in the TO SQL transform function for the type of an external method resulted in the incorrect data type.

The statement cannot be processed.

User Response: Change the RETURN statement of the method or transform function to ensure that the data type of the returned value is always the subject type used to invoke the method.

To determine the routine name associated with the *routine-id*, use the following query:

```
SELECT FUNCSCHEMA, FUNCNAME,
       SPECIFICNAME
FROM   SYSCAT.FUNCTIONS
WHERE  FUNCID = INTEGER(
          routine-id
       )
```

sqlcode: -20083

sqlstate: 2200G

SQL20084N **The** *routine-type routine-name* **would define an overriding relationship with an existing method.**

Explanation: A method MT, with subject type T, is defined to override another method MS, with subject type S, if all of the following conditions are true:

• MT and MS have the same unqualified name and the same number of parameters.

• T is a proper subtype of S.

• The non-subject parameter-types of MT are the same as the corresponding non-subject parameter-types of MS. (Here, "same" applies to the basic type, such as VARCHAR, disregarding length and precision).

A function and a method may not be in an overriding relationship. This means that if the function were a method with its first parameter as subject S, it must not override another method of any supertype of S and it must not be overridden by another method of any subtype of S.

Furthermore, overriding relationships are not supported for:

• table and row methods

• external methods with PARAMETER STYLE JAVA

• system generated mutator and observer methods

The statement cannot be processed.

User Response: Change the routine that is being defined to use a different routine name than *routine-name* or change the parameters of the routine.

sqlcode: -20084

sqlstate: 42745

SQL20085N **A routine defined with PARAMETER STYLE JAVA cannot have structured type** *type-name* **as a parameter type or as the returns type.**

Explanation: A routine is defined with PARAMETER STYLE JAVA and one of the parameter types or a returns type is defined using the structured type *type-name*. This is not supported by this version of DB2.

The statement cannot be processed.

User Response: Change to a different parameter style for the routine or remove the structured type from the routine definition.

sqlcode: -20085

sqlstate: 429B8

SQL20086N **Length of the structured type value for a column exceeds the system limit.**

Explanation: The value for a structured type column exceeds 1 gigabyte in overall size, including descriptor data for the instance. The column could be one that is being inserted or updated directly or it may be a column that is generated.

The statement cannot be processed.

User Response: Reduce the size of the structured type value that is being assigned to the column.

sqlcode: -20086

sqlstate: 54049

SQL20087N **DEFAULT or NULL cannot be used in an attribute assignment.**

Explanation: The UPDATE statement is using an attribute assignment to set the value of an attribute in a structured type column. This form of assignment statement does not allow the use of the keyword DEFAULT or the keyword NULL as the right hand side of the assignment.

The statement cannot be processed.

User Response: Specify an expression for the right hand side of the attribute assignment or change the assignment so it is not using the attribute assignment syntax.

sqlcode: -20087

sqlstate: 428B9

SQL20089N **A method name cannot be the same as a structured type name within the same type hierarchy.**

Explanation: A specified method name is the same as the structured type which is defined for one of the supertypes or subtypes of the structured type.

The statement cannot be processed.

User Response: Specify a different name for the method.

sqlcode: -20089

sqlstate: 42746

SQL20090W **The use of the structured type having attribute *attribute-name* of type DATALINK is effectively limited to the type of a typed table or type view.**

Explanation: The attribute *attribute-name* is defined with a type of DATALINK or a distinct type based on a DATALINK. A structured type that includes such an attribute can only be used as the type of a table or view. If used as the type of a column of a table or view it can only be assigned the null value.

The statement continued processing.

User Response: Consider the intended use of the structured type. If the type will be used as a column data type, then remove the attribute *attribute-name* from the structured type or use a data type other than DATALINK for the attribute.

sqlcode: +20090

sqlstate: 01641

SQL20093N **The table *table-name* cannot be converted to or from a materialized query table. Reason code = *reason-code*.**

Explanation: The ALTER TABLE statement is being used to change a table from a materialized query table to DEFINITION ONLY or to convert a regular table to a materialized query table. The ALTER TABLE statement failed as indicated by the following reason codes.

1   The table is a typed table or hierarchy table.

2   The table is not a materialized query table and DEFINITION ONLY was specified.

3   The table is a replicated materialized query table and DEFINITION ONLY was specified.

4	The table has at least one trigger defined.
5	The table has at least one check constraint defined.
6	The table has at least one unique constraint or unique index defined.
7	The table has at least one referential constraint defined.
8	The table is referenced in the definition of an existing materialized query table.
9	The table is referenced directly or indirectly (through a view, for example) in the fullselect.
10	The table is already a materialized query table.
11	The number of columns of the existing table does not match the number of columns defined in the select list of the fullselect.
12	The data types of the columns of the existing table do not exactly match the corresponding columns in the select list of the fullselect.
13	The column names of the columns of the existing table do not exactly match the corresponding column names in the select list of the fullselect.
14	The nullability characteristics of the columns of the existing table do not exactly match the nullability characteristics of the corresponding columns in the select list of the fullselect.
15	The conversion cannot be performed if there are any other table alterations in the same ALTER TABLE statement.

User Response: The action is based on the reason code as follows.

1	The table cannot be converted to a materialized query table. Create a new materialized query table instead.
2	There is no need to convert this table. No action required.

3	A replicated table can only be a materialized query table. Create a new table instead.
4	Drop any triggers and try the ALTER TABLE statement again.
5	Drop any check constraints and try the ALTER TABLE statement again.
6	Drop any unique constraints and unique indexes. Try the ALTER TABLE statement again.
7	Drop any referential constraints and try the ALTER TABLE statement again.
8	Drop the materialized query table that references the table and try the ALTER TABLE statement again.
9	A materialized query table cannot reference itself. Modify the fullselect to remove the direct or indirect reference to the table being altered.
10	The operation is not allowed since the table is already a materialized query table.
11	Modify the fullselect to include the correct number of columns in the select list.
12	Modify the fullselect so that the result column data types exactly match the data types of the corresponding existing columns.
13	Modify the fullselect so that the result column names exactly match the column names of the corresponding existing columns.
14	The table cannot be converted to a materialized query table unless the nullability characteristics can be matched. Create a new materialized table instead.
15	Perform the other table alterations in an ALTER TABLE statement that does not include the SET MATERIALIZED QUERY AS clause.

sqlcode: -20093

sqlstate: 428EW

SQL20094N **The column** *column-name* **is GENERATED using an expression and cannot be used in the BEFORE trigger** *trigger-name***.**

Explanation: The value for column *column-name* is generated using an expression and cannot be

named in the column name list of a BEFORE UPDATE trigger or referenced as a new transition variable in a BEFORE trigger.

User Response: Remove the reference to *column-name* in the trigger *trigger-name*.

sqlcode: -20094

sqlstate: 42989

SQL20100 - SQL20199

SQL20102N **The CREATE or ALTER statement for the routine** *routine-name* **specified the** *option-name* **option which is not allowed for the routine.**

Explanation: The option *option-name* was specified when creating or altering the routine *routine-name*. The option does not apply to the routine because of other characteristics of the routine.

User Response: For an ALTER statement, ensure that the correct routine is being specified. Otherwise, remove the failing option and reissue the statement.

sqlcode: -20102

sqlstate: 42849

SQL20108N **A result set contains an unsupported data type in position** *position-number* **for the cursor** *cursor-name* **opened by the stored procedure** *procedure-name***.**

Explanation: The stored procedure named by procedure-name cannot return at least one of its query result sets named by cursor-name because at least one of the columns, column position-number contains a data type which is not supported by either the DRDA Application Requestor (client) or the DRDA Application Server (server). As such the call to the stored procedure fails.

User Response: Modify the OPEN statement (and subsequent FETCH statements) for cursor cursor-name in the stored procedure

procedure-name on the server to not select the unsupported data type in column position-number. The client application which invoked the stored procedure may need to be modified to reflect the change in the stored procedure.

sqlcode: -20108

sqlstate: 56084

SQL20109W **Error in DB2 debugger support. Reason code:** *reason-code***.**

Explanation: The debugger support had an error situation which will disable debugging, but not affect normal execution. The following is a list of reason codes:

1. The debugger support is not installed.

2. There has been a syntax error in IP address of debugger client in the debugger table.

3. A timeout in the communication between the debugger backend and the debugger client occurred.

4. A problem occurred accessing the debugger table DB2DBG.ROUTINE_DEBUG.

User Response:

1. Verify that you install the debugger option on the DB2 server machine.

2. Make sure that the IP address in the debugger table has the correct syntax.

3. Check that the debugger demon on the client is started and that the ports on the client and server match.

4. Verify that you have created the debugger tables with the correct layout.

sqlcode: +20109

sqlstate: 01637

SQL20111N   **A SAVEPOINT, RELEASE SAVEPOINT, or ROLLBACK TO SAVEPOINT statement cannot be issued in this context. Reason code =** *reason-code***.**

Explanation: The statement cannot be processed because it violates a restriction as indicated by the following reason code:

1. Savepoint cannot be issued inside a trigger.

2. Savepoint cannot be issued inside a global transaction.

The statement cannot be processed.

User Response: Remove the SAVEPOINT, RELEASE SAVEPOINT, or ROLLBACK TO SAVEPOINT statement in the trigger or global transaction.

sqlcode: -20111

sqlstate: 3B503

SQL20112N   **A SAVEPOINT cannot be set because a SAVEPOINT already exists and nested SAVEPOINTS are not supported.**

Explanation: An error occurred in a SAVEPOINT or atomic compound SQL statement. A savepoint already exists and nested savepoints are not supported in this environment.

User Response: If re-establishing an existing savepoint is necessary, issue the RELEASE SAVEPOINT statement to free the existing savepoint, and then reissue the SAVEPOINT statement. For atomic compound SQL, a SAVEPOINT cannot be set until after the end of the compound statement.

sqlcode: -20112

sqlstate: 3B002

SQL20113N   **Null cannot be returned from method** *method-id* **defined with SELF AS RESULT.**

Explanation: The method with method identifier *method-id* is defined with SELF AS RESULT. The invocation of the method used a non-null instance of a structured type so the method cannot return a null instance.

User Response: Change the method implementation to ensure that a null value is not returned as the return value for the method. One possibility is to set all the attributes of the returned structured type to null value. To determine the name of the method that failed, use the following query:

```
SELECT FUNCSCHEMA, FUNCNAME,
      SPECIFICNAME
  FROM SYSCAT.FUNCTIONS
  WHERE FUNCID = method-id
```

sqlcode: -20113

sqlstate: 22004

SQL20114W   **Column** *column-name* **in table** *table-name* **is not long enough for the defined length of the USER default value.**

Explanation: The column *column-name* is defined with a length that is less than 128 bytes. The clause DEFAULT USER has been specified for this column. Since the USER special register is defined as VARCHAR(128), any attempt to assign the default value for *table-name* by a user with a user ID longer than the column length results in an error. A user with a user ID longer than the column length would never be able to insert or update this column to the default value.

User Response: If your system standards would not allow a user ID to exceed the length of the column, then this warning may be ignored. To prevent this warning from occurring, the length of the column must be at least 128 bytes. You can change the column length by dropping and creating the table again, or, if the data type is VARCHAR, by using ALTER TABLE to increase the length of the column.

sqlcode: +20114

sqlstate: 01642

SQL20115N **The** *routine-type routine-name* **cannot be used as a** *transform-type* **transform function in transform group** *group-name***.**

Explanation: If *routine-type* is FUNCTION, then the function defined by *routine-name* cannot be used as a transform function because it is a built-in function. If *routine-type* is METHOD, then the method defined by *routine-name* cannot be used as a transform function because it is a method.

The statement cannot be processed.

User Response: Specify a different function for the *transform-type* transform function of transform group *group-name*.

sqlcode: -20115

sqlstate: 428EX

SQL20116N **The data type of the search target** *parameter-name* **does not match that of the source key specified in the index extension** *index-extension-name***.**

Explanation: If the search target is of a built-in or distinct data type, its type must exactly match that of the source key specified in the index extension. If the data type of the search target is a structured type, it must be in the same structured type hierarchy as the data type of the source key in the index extension.

The statement cannot be processed.

User Response: Specify a search target with a data type that matches that of the source key in the index extension.

sqlcode: -20116

sqlstate: 428EY

SQL20117N **A window specification for an OLAP function is not valid. Reason code =** *reason-code***.**

Explanation: The window specification (OVER clause) of an OLAP function invocation is not correctly specified. The incorrect specification is indicated by the *reason-code*.

1 RANGE or ROWS is specified without an ORDER BY in the window specification.

2 RANGE is specified and the window ORDER BY clause includes more than one sort-key-expression.

3 RANGE is specified and the data type of the range value cannot be used in a subtraction expression with the data type of the sort-key-expression in the window ORDER BY clause.

4 UNBOUNDED PRECEDING is specified after CURRENT ROW or UNBOUNDED FOLLOWING is specified before CURRENT ROW.

The statement cannot be processed.

User Response: Change the window specification to correct the invalid specification indicated by the *reason-code*.

1 Add a window ORDER BY clause to each window specification that specifies RANGE or ROWS.

2 Ensure that each window specification that includes RANGE has exactly one sort-key-expression in the window ORDER BY clause.

3 For each window specification that includes RANGE, ensure that the range value can be subtracted from the sort-key-expression in the window ORDER BY clause, which must be a numeric type or a datetime type. For datetime sort-key-expressions, the range value must be the specific datetime duration DECIMAL type with the correct precision and scale.

4   Ensure that any window specification using "BETWEEN" and "CURRENT ROW" has the "UNBOUNDED PRECEDING" before "AND CURRENT ROW" or "UNBOUNDED FOLLOWING" after "CURRENT ROW AND".

sqlcode: -20117

sqlstate: 428EZ

SQL20118N   Structured type *type-name* **has more than the maximum number of allowable attributes. The maximum is** *max-value***.**

Explanation: The maximum number of attributes, including inherited attributes, allowed for each structured type has been exceeded with the definition of structured type *type-name*. The maximum number of attributes, including inherited attributes, is *max-value*.

The statement cannot be processed.

User Response: Ensure that the number of attributes for the structured type does not exceed the limit.

sqlcode: -20118

sqlstate: 54050

SQL20119N   A ROW function must define at least two columns.

Explanation: A function that specifies ROW in the RETURNS clause must include a column list with at least two columns.

User Response: Either remove the ROW keyword from the RETURNS clause to make it a scalar function or specify multiple columns in the column list of the RETURNS clause.

sqlcode: -20119

sqlstate: 428F0

SQL20120N   An SQL TABLE function must return a table result.

Explanation: An SQL function that specifies TABLE in the RETURNS clause must return a result that is a table. With the exception of a scalar fullselect, a scalar expression cannot be returned as the result of an SQL TABLE function.

User Response: Either remove the TABLE keyword from the RETURNS clause to make it a scalar function or specify a fullselect in the RETURN statement of the TABLE function body.

sqlcode: -20120

sqlstate: 428F1

SQL20121N   Only one of WITH RETURN or SCROLL can be specified for cursor *cursor-name***.**

Explanation: Both WITH RETURN and SCROLL were specified for cursor *cursor-name*, but this is not allowed.

The statement cannot be processed.

User Response: Change the DECLARE CURSOR statement to either specify NO SCROLL (or remove the SCROLL keyword), or remove the WITH RETURN clause.

sqlcode: -20121

sqlstate: 428F3

SQL20123   CALL to stored procedure *procedure* **failed because the result set returned for a scrollable cursor is not positioned before the first row.**

Explanation: A scrollable result set has been returned by a CALL to stored procedure *procedure*, and one or more of these cursors is not positioned before the first row.

The CALL to the stored procedure was not successful. All result set cursors defined in the stored procedure were closed before returning to the caller. The scrollable cursor cannot be used to FETCH from the result set. Actions done by the stored procedure are not rolled back, and any

external actions initiated by the stored procedure have completed because the error was detected at the end of the execution of the stored procedure.

User Response: Modify the content of the stored procedure to make sure that the result set cursors are positioned before the first row before returning to the caller.

sqlcode: -20123

sqlstate: 560B1

SQL20128 **Cursor** *cursor-name* **is scrollable, but the result table cannot include output from a table function.**

Explanation: The cursor *cursor-name* is defined to be scrollable, but the result table includes output from a table function. This combination is not supported.

The statement cannot be processed.

User Response: Either modify the definition of the cursor so it is not scrollable, or ensure that the result table does not include output from a table function.

sqlcode: -20128

sqlstate: 428F6

SQL20131N **Object number** *object-number* **of type** *object-type* **was specified more than once in a list of objects.**

Explanation: In a list of object names of type *object-type*, the object numbered *object-number* was specified more than once. The operation of the statement cannot be performed on the object more than once.

User Response: Correct the duplicated object in the list removing duplicate occurrences. (In the MDC case, the object type will be "dimension".)

sqlcode: -20131

sqlstate: 42713

SQL20133N **Operation** *operation-name* **cannot be performed on external routine** *routine-name*. **The operation can only be performed on SQL routines.**

Explanation: You attempted to perform operation *operation-name* on external routine *routine-name*. However, you can only perform that operation on SQL routines. The operation did not complete successfully.

User Response: Ensure the name you provide identifies an SQL routine.

sqlcode: -20133

sqlstate: 428F7

SQL20134N **The SQL Archive (SAR) file for routine** *routine-name* **could not be created on the server.**

Explanation: The creation of the SQL archive (SAR) for routine *routine-name* failed because DB2 could not find either the library or the bind file for the specified routine. Bind files are only available for SQL routines created with DB2 Version 7.1, FixPak 2 or later.

User Response: Recreate the procedure on a server with DB2 Version 7.1, FixPak 2 or later, and try the operation again.

sqlcode: -20134

sqlstate: 55045

SQL20135N **The specified SQL archive does not match the target environment. Reason code =** *reason-code*.

Explanation: The specified SQL archive does not match the target environment for one of the following reasons:

1   The operating system of the target environment is not the same as the operating system on which the SQL archive was created.

2   The database type and level of the target environment is not the same as

the database type and level on which the SQL archive was created.

User Response: Ensure that the environment on which the SQL archive was created matches the target environment and reissue the command. If the environments do not match, you must manually create the SQL routine using the target environment.

sqlcode: -20135

sqlstate: 55046

SQL20136N Routine *routine-name* **(specific name** *specific-name***) is declared NOT FEDERATED but the SQL statement attempted to access a federated object.**

Explanation: This SQL statement attempted to access one or more federated objects. This statement is executed from routine *routine-name* (specific name *specific-name*) which is defined as NOT FEDERATED. It is not permitted to access a federated object from a NOT FEDERATED routine.

User Response: Either remove the references to federated objects from the statement, or ALTER the routine to define it as FEDERATED.

sqlcode: -20136

sqlstate: 55047

SQL20138N The statement may not be explained because the routine *routine-name* **(specific name** *specific-name***) is not defined as MODIFIES SQL DATA.**

Explanation: The routine *routine-name* (specific name *specific-name*) is defined as either CONTAINS SQL or READS SQL DATA and attempted to explain an SQL statement. Explaining an SQL statement requires writing to the explain tables, which is only permitted in a MODIFIES SQL DATA routine.

User Response: Do not attempt to explain SQL statements from a routine defined as either CONTAINS SQL or READS SQL DATA.

sqlcode: -20138

sqlstate: 42985

SQL20139N SQL statements may not be issued in routine *routine-name* **(specific name** *specific-name***) because of a previous statement failed or was interrupted.**

Explanation: During the execution of routine *routine-name* (specific name *specific-name*), or a nested routine, either a statement failed and requires a roll back or an interrupt occurred. It is necessary for all routines invoked from the outer-most statement to finish executing and for control to be passed back to the outer-most statement so that the database manager may perform the necessary recovery. Until this recovery has been completed, SQL statements may not be issued.

User Response: The routine continues to execute. The routine should not execute any further SQL statements and should return control to the invoking statement as quickly as possible.

The recovery for the initial failure of interrupt will be automatically performed by the database manager once all routines have completed.

sqlcode: -20139

sqlstate: 51038

SQL20140W COMPRESS column attribute ignored because VALUE COMPRESSION is deactivated for the table.

Explanation: One of the following situations occurred.

1. The COMPRESS SYSTEM DEFAULT specified for the column was ignored because VALUE COMPRESS is deactivated for the table.

2. DEACTIVATED VALUE COMPRESSION was specified and columns are defined with COMPRESS SYSTEM DEFAULT.

User Response: To allow COMPRESS to be specified for the column, ALTER the table and activate VALUE COMPRESSION for the table.

sqlcode: 20140

sqlstate: 01648

SQL20142N **Sequence** *sequence-name* **cannot be used as specified.**

Explanation: *sequence-name* was referenced in a context in which it cannot be used. *sequence-name* is a sequence generated by the system for an identity column. These sequences cannot be referenced in a COMMENT ON SEQUENCE, DROP SEQUENCE, GRANT or REVOKE statement, or in a NEXTVAL or PREVVAL expression.

User Response: Specify the name of a user-defined sequence object in this context.

sqlcode: -20142

sqlstate: 428FB

SQL20143N **The encryption or decryption function failed, because the ENCRYPTION PASSWORD value is not set.**

Explanation: The ENCRYPTION PASSWORD value is not set.

User Response: Issue the SET ENCRYPTION PASSWORD statement to set the ENCRYPTION PASSWORD value. The length of the password must be a minimum of 6 bytes and a maximum of 127 bytes.

sqlcode: -20143

sqlstate: 51039

SQL20144N **The encryption password is invalid because the length of the specified password was less than 6 bytes or greater than 127 bytes.**

Explanation: The data must be encrypted with a password length between 6 and 127.

User Response: Ensure that the password length is between 6 and 127.

sqlcode: -20144

sqlstate: 428FC

SQL20145N **The decryption function failed. The password used for decryption does not match the password used to encrypt the data.**

Explanation: The data must be decrypted using the same password that was used to encrypt the data.

User Response: Ensure that the same password is used to encrypt and decrypt the data.

sqlcode: -20145

sqlstate: 428FD

SQL20146N **The decryption function failed. The data is not encrypted.**

Explanation: The data must be a result of the ENCRYPT function.

User Response: Ensure that the data type is a result of the ENCRYPT function.

sqlcode: -20146

sqlstate: 428FE

SQL20147N **The ENCRYPT function failed. Multiple pass encryption is not supported.**

Explanation: Data that has already been encrypted cannot be encrypted again.

User Response: Ensure that the data has not already been encrypted.

sqlcode: -20147

sqlstate: 55048

SQL20148N **Routine** *routine-name* **with specific name** *specific-name* **must have the RETURN statement as the last SQL statement of the compound body.**

Explanation: The RETURN statement must be the last SQL statement of the compound body in an SQL ROW or TABLE function. No other

RETURN statement is allowed within the routine body.

User Response: Ensure there is only one RETURN statement and it is the last SQL statement of the compound body.

sqlcode: -20148

sqlstate: 429BD

SQL20149W **The buffer pool configuration has completed but will not take effect until the next database restart.**

Explanation: The buffer pool has been successfully configured, but the changes will not take effect immediately. The changes will take effect when the database is restarted.

User Response: To have the changes take effect at next database restart time, no further action is required.

sqlcode: 20149

sqlstate: 01649

SQL20150N **The number of block pages for a buffer pool is too large for the size of the buffer pool.**

Explanation: The number of block pages, as specified by NUMBLOCKPAGES, must not be greater than 98 percent of the number of pages for the buffer pool, as specified by SIZE.

User Response: Either disable block-based I/O for this buffer pool by setting NUMBLOCKPAGES to zero or ensure the value of NUMBLOCKPAGES is not greater than 98 percent of the value for SIZE.

sqlcode: -20150

sqlstate: 54052

SQL20151N **The value specified for BLOCKSIZE is not in the valid range.**

Explanation: The valid range for the value of BLOCKSIZE is 2 to 256.

User Response: Change the value for BLOCKSIZE to be greater than or equal to 2 and less than or equal to 256. An optimal value would be the extent size.

sqlcode: -20151

sqlstate: 54053

SQL20152N **The specified buffer pool is not currently block-based.**

Explanation: The BLOCKSIZE option was used without specifying the number of pages to use in the block area of the buffer pool.

User Response: Specify a value for NUMBLOCKPAGES along with the BLOCKSIZE option.

sqlcode: -20152

sqlstate: 428FF

SQL20153N **The database's split image is in the suspended state.**

Explanation: The database split image cannot be used while it is in the suspended state.

User Response: Issue the db2inidb command with one of the following three options to resume I/O for this database split image:

- db2inidb <db-name> as mirror
- db2inidb <db-name> as snapshot
- db2inidb <db-name> as standby

In a multi-node environment, the db2inidb tool must be run on every node before the database can be used. The db2inidb tool can run concurrently on each node of a multi-node database.

sqlcode: -20153

sqlstate: 55040

SQL20154 **The requested insert operation into view** *view-name* **is not allowed because no target table can be determined for a row. Reason code =** *reason-code***.**

Explanation: The specified view contains a UNION ALL query. The *reason-code* indicates that a given row either:

1. does not satisfy the check constraint of any underlying base table, or

2. satisfies all the check constraints for more than one underlying base table.

Federated system users: Some other data source specific limitation may be preventing the row from being inserted.

User Response: Ensure that the check constraints used by the underlying base tables to partition their rowsets cover the set of rows to be inserted. For example, given the check constraints (T1.c1 in (1,2)) on T1, and (T2.c1 in (2,3)) on T2, and view V1 as a union of T1 and T2,

1. the row c1 = 4 does not satisfy the check constraints of either underlying base table and

2. the row c1 = 2 satisfies the check constraints of both underlying base tables.

Federated system users: If the reason is unknown, isolate the problem to the data source failing the request (see the problem determination guide) and examine the object definition and the update restrictions for that data source.

sqlcode: -20154

sqlstate: 23513

SQL20155N **The specified event monitor target tables are invalid. Reason code =** *reason-code***.**

Explanation: While processing a CREATE EVENT MONITOR statement, or while activating an event monitor, it was determined that a target table is not valid *reason-code* is one of the following:

1. At least on column name does not match an event monitor data element identifier.

2. At least one column has a data type that is incompatible with the data type for an event monitor data element identifier.

3. At least one column name matches an event monitor data element identifier, but that element is not allowed in the target table.

4. The row size for the table is too large for the page size of the table space.

5. A mandatory column is missing.

User Response: Check the administration notification log for detailed information and correct the table definition.

sqlcode: -20155

sqlstate: 55049

SQL20156W **The event monitor was activated successfully, however some monitoring information may be lost.**

Explanation: An event monitor activated successfully, however it was determined that the event monitor may lose some information because of one of the following:

• The data type for a column is smaller than the data type required to hold an event monitor element identifier. This data will be truncated.

• A target table was found in the SYSCAT.EVENTTABLES, but that table does not exist in the database. All monitor information for the corresponding table will not be recorded.

User Response: Check the administration notification log for detailed information. If necessary, recreate the event monitor to create all target tables.

sqlcode: 20156

sqlstate: 01651

SQL20157N **User** *authorization-ID* **does not have the QUIESCE_CONNECT privilege.**

Explanation: The specified authorization ID does not have the QUIESCE_CONNECT privilege to access a database or instance that is currently in a quiesced mode. The QUIESCE_CONNECT privilege must be granted before the user can connect to a quiesced database or instance.

User Response: Wait for the unquiesce to complete or contact the system administrator or database administrator for the database and request a GRANT QUIESCE_CONNECT for the authorization ID. Resubmit the command.

sqlcode:-20157

sqlstate:08004

SQL20158N **Function not supported for the level of the connected DB2 Data Links Manager.**

Explanation: The underlying DATALINK feature is not supported for the given level of the connected DB2 Data Links Manager.

User Response: The error can be corrected by removing the use of any unsupported DATALINK feature or by upgrading the DB2 Data Links Manager to the same release level as the database server.

sqlcode: -20158

sqlstate: 42997

SQL20159W **The isolation clause is ignored because of the statement context.**

Explanation: The isolation clause is specified on a statement that is included in a compound statement that is processed as inline SQL. The isolation clause is ignored and the default isolation level is used for all statements in the compound statement.

User Response: No response required. Remove the isolation clause to prevent this warning.

sqlcode: 20159

sqlstate: 01652

SQL20160W **The authorizations were granted to USER** *userid***. Groups were not considered because the authorization name is more than 8 bytes.**

Explanation: The authorization name has a length of more than 8 bytes. The privilege is granted to the user with authorization name *userid* without considering groups defined in the system that might have a matching name. Processing continues.

User Response: If the grant was intended for a user, no action is required. If the grant was intended for a group, consider choosing an alternate group because group names greater than 8 bytes are not supported. To avoid this warning message, specify the USER keyword before the authorization name.

sqlcode: 20160

sqlstate: 01653

SQL20161W **The column name** *column-name* **is not a valid column for an event monitor table.**

Explanation: The column name specified in the INCLUDES or EXCLUDES clause does not match any valid column for the event monitor table(s) being created.

User Response: Examine and correct the specified column name.

sqlcode: -20161

sqlstate: 428AA

SQL20162N **Buffer pools cannot use both block-based I/O and extended storage.**

Explanation: An attempt was made to create or alter a buffer pool to use both block-based I/O and extended storage.

User Response: Ensure that the buffer pool is

not using both block-based I/O and extended storage.

sqlcode: -20162

sqlstate: 428FF

SQL20167N   The shared memory set is not large enough to process the request.

Explanation: There is insufficient memory in a shared memory set for one of the following resource requests:

- FCM connection entries
- FCM message anchors
- FCM request blocks

User Response: Try the request again. If the error persists, stop and restart the instance to allow the larger shared memory set size (automatically updated by DB2) to take effect and try the request again.

sqlcode: -20167

sqlstate: 57011

SQL20168N   The ALTER BUFFERPOOL statement is currently in progress.

Explanation: A buffer pool cannot be dropped or altered if an ALTER operation is already in progress.

User Response: Wait until the ALTER operation has completed. Use the snapshot monitor to check the progress of the ongoing ALTER operation.

sqlcode: -20168

sqlstate: 55051

SQL20169W   The buffer pool is not started.

Explanation: The statement was successful, but it will only take effect when the buffer pool is started.

User Response: The statement has completed successfully, and will take effect when the database is activated. This happens after all applications disconnect. For the statement to take effect immediately, re-submit it when the buffer pool is started. To start the buffer pool, it is possible in some cases to drop and re-create the buffer pool using the IMMEDIATE option.

sqlcode: +20169

sqlstate: 01654

SQL20170N   There is not enough space in the table space *tablespace-name* **for the specified action.**

Explanation: Space is being removed from a tablespace using one or more of the DROP, REDUCE, and RESIZE container actions. However, the amount of space being removed is greater than the amount of space above the high-water mark.

User Response: Use the snapshot monitor to find the number of useable pages in the tablespace and the tablespace high-water mark. The difference between these values is the maximum number of pages that can be removed.

sqlcode: -20170

sqlstate: 57059

SQL20173W   The event monitor was created successfully but at least one event monitor target table already exists.

Explanation: While an event monitor was being created, the database manager tried to create one or more target tables but determined that tables with those names already exist. The event monitor was successfully created and target tables were not created to replace tables of the same name that already exist. When the event monitor is activated, it will attempt to use these tables. Event monitor activation may fail if activation processing determines these tables are not suitable.

User Response: Check the administration notification log for detailed information. If necessary, drop all target tables and recreate the event monitor to create all target tables.

sqlcode: 20173

SQL20178N **View** *view-name* **already has an INSTEAD OF** *operation* **trigger defined.**

Explanation: The view *view-name* already has an INSTEAD OF trigger defined for the indicated operation (UPDATE, DELETE, INSERT). Only one INSTEAD OF trigger can be defined for each operation on the view.

The statement cannot be processed. The INSTEAD OF trigger was not created.

User Response: If a new INSTEAD OF trigger is required, drop the existing trigger and then create the new trigger.

sqlcode: -20178

sqlstate: 428FP

SQL20179N **The INSTEAD OF trigger cannot be created because the view** *view-name* **is defined using the WITH CHECK OPTION.**

Explanation: An INSTEAD OF trigger cannot be defined on a symmetric view (a view defined using WITH CHECK OPTION) or a view on which a symmetric view has been defined directly or indirectly. The view *view-name* may be the target view of the INSTEAD OF trigger or it may be a view that depends directly or indirectly on the target view of the trigger.

The statement cannot be processed. The INSTEAD OF trigger was not created.

User Response: INSTEAD OF triggers cannot be defined on the view unless the symmetric view is replaced by a view that is not symmetric. The view would have to be dropped and re-created to allow an INSTEAD OF trigger.

sqlcode: -20179

sqlstate: 428FQ

SQL20188N **The primary key or a unique key identified by** *name* **is a subset of the columns in the ORGANIZE BY clause.**

Explanation: All the columns in a primary or unique key are included in the table's ORGANIZE BY clause. This is not allowed, since each block of pages in the table would contain just one row, wasting all the rest of the space in the block.

name is the constraint name, if specified, of the primary key or unique constraint. If a constraint name was not specified, *name* is the first column name specified in the primary key or unique constraint clause followed by three periods. If a unique index is being created, *name* is the name of the unique index.

The statement cannot be processed.

User Response: Modify the primary or unique key definition, the unique index definition or the ORGANIZE BY clause, if possible, so that the key is not a subset of the dimensions.

sqlcode: -20188

sqlstate: 429BE

SQL20189W **The buffer pool operation (CREATE/ALTER) will not take effect until the next database startup due to insufficient memory.**

Explanation: The CREATE or ALTER BUFFERPOOL statement was issued and completed successfully, however due to insufficient memory the create/alter was done DEFERRED. The changes will take effect on the next database startup.

User Response: If you do not want to wait until the next startup to activate or resize the buffer pool, you can free memory resources and try again (with the same or different size). Memory resources that can be reduced include other buffer pools, the database heap, the catalog cache, the package cache, and the utility heap. These can be reduced using the ALTER/DROP BUFFERPOOL or UPDATE DATABASE

CONFIGURATION command depending on the resource. In the future, to reserve extra memory for the dynamic allocation of buffer pool memory, you can increase the DATABASE_MEMORY database configuration parameter.

If you do not try again:

1. If it is an ALTER BUFFERPOOL that failed, you will continue to run with the current runtime size of the buffer pool. You can use the database monitor to see the current runtime size of the buffer pool.

2. If it is a CREATE BUFFERPOOL that failed, any table spaces that are created in the buffer pool will temporarily (until next startup) be put in a hidden buffer pool with the matching pagesize. Since the hidden buffer pools are small, this can result in lower than desired performance.

To try again:

1. For ALTER BUFFERPOOL, resubmit the command.

2. For CREATE BUFFERPOOL drop the buffer pool and then resubmit the command.

sqlcode: 20189

sqlstate: 01657

SQL20190N **Federated insert, update, or delete operation cannot be compiled because of potential data inconsistency.**

Explanation: When the data source does not provide application savepoint support and the server option 'iud_app_svpt_enforce' is set to 'Y', the federated insert, update, or delete operation is blocked during precompile. This will prevent potential data inconsistency, which may occur at runtime execution if an error is encountered in the middle of federated insert, update, or delete processing.

User Response: Possible solutions include:

- Change server option 'iud_app_svpt_enforce' to 'N'.

- Apply the insert, update, or delete directly on the data source.

sqlcode: -20190

sqlstate: 0A503

SQL20191N **For an INOUT parameter in a dynamically prepared CALL statement, the same host variable must be used in both the USING and INTO clauses.**

Explanation: In a CALL statement, an INOUT parameter must correspond to a single host variable. When executing a dynamically prepared CALL statement, it is necessary to specify the same host variable in both the USING and INTO clauses of the EXECUTE statement.

User Response: When specifying a host variable that corresponds to a parameter marker for an INOUT parameter, use the same host variable in both the USING and INTO clauses of the EXECUTE statement. When using SQLDAs to specify a host variable that corresponds to a parameter marker for an INOUT parameter, then the SQLDATA pointers of the corresponding SQLVARs must point to the same host variable.

sqlcode: -20191

sqlstate: 560BB

SQL20192N **The specified modes are only supported in a partitioned database environment.**

Explanation: Either RECOMMEND PARTITIONINGS or EVALUATE PARTITIONINGS was not invoked in a partitioned database environment. These two modes are only supported in a partitioned database environment.

User Response: Invoke RECOMMEND PARTITIONINGS or EVALUATE PARTITIONINGS in a partitioned database environment.

sqlcode: -20192

sqlstate: 56038

SQL20193N An error has occurred when accessing the file *file-name*. **Reason code:** *reason-code*.

Explanation: If the file name is db2nodes.cfg, some possible reason codes include:

1. Indicates that the db2nodes.cfg file contains format errors.
2. The command could not be completed because of communication errors.
3. The command could not be completed because of memory allocation errors.

User Response: Fix any format errors in the db2nodes.cfg file. Check the db2diag.log file for more information to help resolve the communication or memory allocation errors. Rerun the application.

sqlcode: -20193

sqlstate: 560BC

SQL20194N Buffer pool *bufferpool-name* **does not exist on database partition** *dbpartitionnum*.

Explanation: The ALTER BUFFERPOOL statement is specifying a buffer pool, *bufferpool-name*, that does not exist on the database partition *dbpartitionnum*.

User Response: Using the ALTER DATABASE PARTITION GROUP statement, add the database partition *dbpartitionnum* to a database partition group that has the buffer pool *bufferpool-name* already defined for it. If the buffer pool is not associated with any specific database partition group, then add the database partition to any database partition group or create a new database partition group for this database partition. Issue the ALTER BUFFERPOOL statement again.

sqlcode: -20194

sqlstate: 53040

SQL20195N An error was encountered while processing line number *line-number* **in the path rename configuration file** *config-file*. **Reason code =** *reason-code*.

Explanation: The path rename configuration file *config-file* is being used to rename containers during the current crash or rollforward recovery process. However, while processing the statements in this file, an error was encountered at line *line-number* that is preventing the recovery process from proceeding. An explanation of the error is indicated by the following reason code:

1. Invalid syntax.
2. The length of the container path exceeds the maximum allowed length for a container path.
3. The container path specified has already been listed on a previous line.
4. The length of the database path exceeds the maximum allowed length for a database path.
5. The database path specified has already been listed on a previous line.
6. The database path must be an absolute path.
7. The database path is specified incorrectly.

User Response: Make the corresponding change to the configuration file as described below and reissue the command again.

1. Correct the syntax error.
2. Specify a shorter container path.
3. Remove the duplicate container path from the file.
4. Specify a shorter database path.
5. Remove the duplicate database path from the file.
6. Use an absolute database path instead of a relative one.
7. Specify the database path in the form of a drive letter followed by a colon (for example C:).

sqlcode: -20195

sqlstate: 08504

SQL20196N **The one or more built-in types which are returned from the FROM SQL function or method does not match the corresponding built-in types which are parameters of the TO SQL function or method.**

Explanation: The built-in types which are returned by a FROM SQL transform function or method have to match the types in the parameter list of the corresponding TO SQL transform function or method.

User Response: Choose a different FROM SQL transform function or method or TO SQL transform function or method or alter either the FROM SQL transform function or method or TO SQL transform function or method to make sure that each built-in type returned from the FROM SQL function or method matches the corresponding built-in type that is a parameter of the TO SQL transform function or method.

sqlcode: -20196

sqlstate: 428FU

SQL20197N **Cannot define** *method-name* **as an overriding method. Reason code** *reason-code***.**

Explanation: An attempt was made to define *method-name* as an overriding method. An overriding relationship between this method and an original method cannot be created because of one of the following reasons, as specified by *reason-code*:

1   Cannot find an original method with the same name.

2   The original method and the overriding method do not have the same number of parameters.

3   The data type of the parameter of the overriding method does not match the data type of the corresponding parameter of the original method.

4   The parameter name of the parameter of the overriding method does not match

the parameter name of the corresponding parameter of the original method.

5   The locator indication of the parameter of the overriding method does not match the locator indication of the corresponding parameter of the original method.

6   The FOR BIT DATA indication of the parameter of the overriding method does not match the FOR BIT DATA indication of the corresponding parameter of the original method.

7   The RETURNS clause of the overriding method contains as AS LOCATOR clause, a CAST FROM clause or a FOR BIT DATA clause.

8   One of the following inherited method attributes is specified: SELF AS RESULT, SQL routine characteristics, or external routine characteristics.

9   Return type of the overriding method does not match the return type of the original method.

10   The return type of the overriding method is a structured type that is not a subtype of either:

- the return type of the original method or
- every return type for all methods that already override the original method.

User Response: Modify the method definition based on the reason specified by the *reason-code* as follows:

1   Use a method name that identifies an existing method for the supertype of the structured type.

2   Define the same number of parameters for the overriding method as exists for the original method.

3   Change the data type to match the data type of the corresponding parameter of the original method.

4   Change the parameter name to match

the name of the corresponding parameter of the original method.

5   Change the locator indication to match the locator indication of the corresponding parameter of the original method.

6   Change the FOR BIT DATA indication to match the FOR BIT DATA indication of the corresponding parameter of the original method.

7   Remove the AS LOCATOR clause, CAST FROM clause and FOR BIT DATA clause for the overriding method.

8   Remove the SELF AS RESULT method attribute and any SQL routine characteristic or external routine characteristic.

9   Change the return type to match the return type of the original method.

10   Change the return type so that it is a subtype of the original method and all methods that already override the original method.

sqlcode: -20197

sqlstate: 428FV

SQL20198N   **Method** *method-name* **calls itself recursively.**

Explanation: A method attempted to call itself recursively. The recursion might be direct or indirect in nature.

User Response: Modify the method definition.

sqlcode: -20198

sqlstate: 55054

SQL20200 - SQL20299

SQL20200N   **The install or replace of** *jar-id* **failed as** *url* **could not be located.**

Explanation: The URL specified on the install or replace jar procedure did not identify a valid jar file.

User Response: Reissue the install or replace jar procedure with a URL that identifies a valid jar file.

sqlcode: -20200

sqlstate: 46001

SQL20201N   **The install, replace or remove of** *jar-id* **failed as the jar name is invalid.**

Explanation: The jar name specified on the install, replace or remove jar procedure was invalid. For example, the jar id may be of the improper format, may not exist to be replaced or removed, or can not be installed as it already exists.

User Response: Ensure the jar id is of the

correct format. If the jar id exists, it may need to be removed before it can be installed. For the remove or replace procedures, ensure the jar id exists.

sqlcode: -20201

sqlstate: 46002

SQL20202N   **The replace or remove of** *jar-id* **failed as** *class* **is in use.**

Explanation: The specified class in the jar file is currently in use by a defined procedure, or the replacement jar file does not contain the specified class for which a procedure is defined.

User Response: Ensure all procedures referencing the classes being removed are dropped and resubmit the replace or remove procedure.

sqlcode: -20202

sqlstate: 46003

SQL20203N   User defined function or procedure *function-name* **has a Java method with an invalid signature.**

Explanation: The signature of the java method used to implement the function or procedure was invalid. For example, the method may have parameters that are not mappable to the parameters on the corresponding create statement or the method for a procedure may specify a return value.

User Response: Reissue the corresponding CREATE statement specifying parameters that match the Java method, or correct the parameters or return type of the Java method and rebuild the class.

sqlcode: -20203

sqlstate: 46007

SQL20204N   The user defined function or procedure *function-name* **was unable to map to a single Java method.**

Explanation: The identified function or procedure either failed to find a matching Java method, or found more than 1 matching Java method.

User Response: Correct either the Java method or corresponding create statement so that the function or procedure call resolves to a single Java method.

sqlcode: -20204

sqlstate: 46008

SQL20205N   The user defined function or procedure *function-name* **has an input argument with a null value that could not be passed to the method.**

Explanation: A function created with "CALLED ON NULL INPUT" or procedure has an input parameter with a null value but the Java datatype of this argument does not support null values. Examples of Java datatypes that do not support null values are BOOLEAN, BYTE,

SHORT, INT, LONG or DOUBLE.

User Response: If the method is to be called with null values, ensure the input Java types are capable of accepting a null value. For functions, the function could also be created with "RETURNS NULL ON NULL INPUT".

sqlcode: -20205

sqlstate: 39004

SQL20206W   The procedure *function-name* **returned too many result sets.**

Explanation: The specified procedure returned more results sets than were specified on the CREATE PROCEDURE statement.

User Response: Modify the procedure to return fewer result sets, or drop and recreate the procedure specifying the correct number of result sets.

sqlcode: +20206

sqlstate: 0100E

SQL20207N   The install or remove jar procedure for *jar-id* **specified the use of a deployment descriptor.**

Explanation: The DEPLOY or UNDEPLOY parameter of the install or replace jar procedure was non-zero; this parameter is not supported and must be zero.

User Response: Reissue the procedure with the DEPLOY or UNDEPLOY parameter set to zero.

sqlcode: -20207

sqlstate: 46501

SQL20208N   Table *table-name* **cannot be created. Reason code =** *reason-code*

Explanation: The table cannot be created because it violates a restriction as indicated by the following reason code:

1. The table used to define a staging table is not a materialized query table with the REFRESH DEFERRED option.

2. The table used to define the staging table already has a staging table associated with it.

The statement cannot be processed.

User Response: The action corresponding to the reason code is:

1. Specify a materialized query table with the REFRESHED DEFERRED option to define the staging table.
2. Specify a materialized query table that is not associated with a staging table.

sqlcode:-20208

sqlstate:428FG

SQL20209N **The** *option-type* **option is not valid for table** *table-name* **with reason code** *reason-code*

Explanation: The specified option is not valid as indicated by the following reason codes:

1. The READ ACCESS option cannot be specified on a table that is in the check pending no access state.
2. The FULL ACCESS option is not valid if the table still has integrity types that are not checked.
3. The FULL ACCESS option is not valid on a table that is not in the no data movement mode.
4. The FULL ACCESS option cannot be specified with the IMMEDIATE UNCHECKED clause if the table is in the no data movement mode.
5. The PRUNE option is not valid on a table that is not a staging table.
6. The PRUNE and INCREMENTAL options cannot be specified simultaneously.

User Response:

1. Do not specify the READ ACCESS option.
2. Use the FULL ACCESS option with the SET INTEGRITY ... IMMEDIATE CHECKED statement, or specify ALL as the integrity option for the SET INTEGRITY ... IMMEDIATE UNCHECKED statement.

3. Use the FULL ACCESS option with the IMMEDIATE CHECKED option.
4. Use the FULL ACCESS option without the IMMEDIATE UNCHECKED clause.
5. Remove the table that is not a staging table from the statement.
6. Specify only one of PRUNE or INCREMENTAL.

sqlcode:-20209;

sqlstate:428FH;

SQL20210N **ORDER OF** *table-designator* **was specified. But this table-designator does not contain an ORDER BY clause.**

Explanation: It was specified to order based on the order of *table-designator* but *table-designator* is not ordered since it does not contain an ORDER BY clause.

User Response: Add an ORDER BY clause to the specification of *table-designator* or use a sort-key specification instead of ORDER BY.

sqlcode: -20210

sqlstate: 428FI

SQL20211N **The specification ORDER BY or FETCH FIRST n ROWS ONLY is invalid.**

Explanation: An ORDER BY or FETCH FIRST n ROWS ONLY is not allowed in:

- the outer fullselect of a view
- the outer fullselect in the RETURN statement of an SQL Table function
- a materialized query table definition
- a subselect which is not enclosed in parenthesis

User Response: In case of:

subselect
Enclose the subselect including the ORDER BY or FETCH FIRST n ROWS ONLY in parenthesis.

FETCH FIRST n ROWS ONLY
Use the ROW_NUMBER() OVER()
clause in conjunction with a predicate in
the where clause. Example:

```
SELECT name FROM
  (SELECT
     ROW_NUMBER() OVER() AS m, name
     FROM emp
  ) AS e
WHERE m < 10
```

ORDER BY
Use ORDER BY in the query using the
view, the materialized query table, or
the SQL table function instead.

sqlcode: -20211

sqlstate: 428FJ

SQL20212N User defined routine *function-name*
encountered an exception
attempting to load Java class
class-name **from JAR** *JAR-name***.**
Original exception
*underlying-exception***.**

Explanation: A ClassNotFoundException has
occurred. The ClassNotFoundException may
reference an original Java exception which
further describes why the class wasn't found. For
example, the underlying exception might be an
I/O error that occurred trying to read a JAR file,
or an SQL error occurred reading the JAR from
the data dictionary. As much of the following
tokens that will fit within the SQLCA limits will
be returned, as appropriate.

The *function-name* identifies the specific name of
the external Java function or procedure whose
execution encountered the
ClassNotFoundException.

The *class-name* identifies the Java class whose
definition could not be found.

The *JAR-name* optionally identifies the installed
JAR expected to contain the identified class.
'(none)' will be present unless the external Java
function or procedure is defined to be in an
installed JAR.

The *underlying-exception* optionally contains the
toString() of the underlying exception that
resulted in this ClassNotFoundException, if any.
'(none)' will be present if there is no underlying
exception.

User Response: Execute an ALTER
PROCEDURE to specify where the class is to be
found, make sure the class is in the specified JAR
or the system classpath, or correction the
condition reported by the *underlying-exception*.

sqlcode: -20212

sqlstate: 46103

SQL20214N ORDER OF *table-designator* **was**
specified but this table designator
is not ordered.

Explanation: The ORDER OF clause specifies
ordering of the result table based on the order
of*table-designaotr*, which is not ordered. There is
no ORDER BY clause on the query associated
with *table-designator*, or the sort key of the
ORDER BY clause is a constant expression.

User Response: Add a valid ORDER BY clause
to the query associated with *table-designator*, or
use a sort-key specification instead of using the
ORDER OF clause.

sqlcode: -20214

sqlstate: 428FI

SQL20223N The ENCRYPT or DECRYPT
function failed. Encryption facility
not available.

Explanation: The encryption facility is not
available to service the encryption or decryption
request.

User Response: If the encryption facility is not
installed, then install it before using the
ENCRYPT or DECRYPT functions. If the
encryption facility is installed, then verify that it
is working correctly.

sqlcode: -20223

sqlstate: 560BF

SQL20225W The buffer pool operation (DROP) will not take effect until the next database startup because the buffer pool is in use.

Explanation: The DROP BUFFERPOOL statement was issued and completed successfully, however a table space is still using this buffer pool. Even though a table space may be reassigned to a different buffer pool, the table space reassignment may only take effect on the next database restart. If the table space in question is dropped by the time this operation commits, then this buffer pool will be removed from memory at that time. Until the next database restart, this buffer pool may remain in memory and a CREATE or ALTER of any other buffer pool that would reuse this buffer pool ID may be deferred.

User Response: Restart the database to delete the buffer pool from memory.

sqlcode: 20225

sqlstate: 01657

SQL22000 - SQL22099

SQL22000W Cannot find the requested configuration for object *object-name*. **Returning default configuration for** *object-type*.

Explanation: The object does not have a specific configuration of its own so the default configuration for that object type will be returned.

User Response: If the behavior of the default configuration is correct, no action has to be taken.

SQL22001W Cannot find the default configuration for object *object-name*. **Returning install configuration for** *object-type*.

Explanation: The object does not have a specific configuration of its own so the install configuration for that object type will be returned.

User Response: If the behavior of the install configuration is correct, no action has to be taken.

SQL22004W Cannot find the requested configuration for the given object. Returning default configuration for *object-name*.

Explanation: The object does not have a specific configuration of its own so the default configuration for that object type will be returned.

User Response: If the behavior of the default configuration is correct, no action has to be taken.

SQL22005W Cannot find the default configuration for the given object, returning install configuration for *object-name*.

Explanation: The object does not have a specific configuration of its own so the install configuration for that object type will be returned.

User Response: If the behavior of the install configuration is correct, no action has to be taken.

SQL22006N Cannot update or delete *object-action-or-contact-name* **because it does not exist.**

Explanation: The API was requested to update the configuration of an element which does not exist.

User Response: Create the element and then re-issue the API call.

SQL22007N The specified Health Indicator identifier or name, *Health-Indicator-name*, is invalid.

Explanation: The requested action can only be performed against a valid Health Indicator. The supplied Health Indicator does not exist.

User Response: Determine the correct the identifier or name, and resubmit the request.

SQL22008N The value *parameter-value* of parameter *parameter-name* is not of the correct format, type, or value.

Explanation: The requested action can not be taken because the supplied value for this parameter is not of an acceptable format, type, or value.

User Response: Determine the correct format and type of, or value for, this parameter, and resubmit the request.

SQL22009W There is no health contact information for this instance.

Explanation: There is no health contact information for this instance.

User Response: If the current contact information is correct, no action has to be taken.

SQL22010N The value *parameter-value* of parameter *parameter-name* is not within the acceptable range of values for this parameter, which is greater than *parameter-max-value* and/or less than *parameter-min-value*.

Explanation: The requested action can not be taken because the supplied value for this parameter is not within the acceptable range of values.

User Response: Resubmit the request with an acceptable value for this parameter.

SQL22011W Cannot find the configuration for *object-name-or-type*.

Explanation: The object or object type does not have a specific or default configuration of its own.

User Response: No action required.

SQL22012W Implicit instance attach failed.

Explanation: An explicit instance attachment does not exist and an implicit attachment to the default instance failed. The command cannot be executed.

User Response: Make sure that DB2 is started and that environment variables are set properly.

SQL22013N Cannot add *obj-act-contact* because it already exists.

Explanation: The API was requested to add an action or notification which already exists.

User Response: Modify or delete the existing actions and notifications.

SQL22014W The health monitor returned no health related data.

Explanation: There is no health data for this instance or the health monitor is off.

User Response: Verify that the health monitor is running on this instance.

SQL22015N The specified Health Indicator identifier or name, *Health-Indicator-name*, is invalid for this object type.

Explanation: The supplied Health Indicator does not exist for this object type.

User Response: Verify the object type and identifier or name, and resubmit the request.

SQL22200 - SQL22299

SQL22200N The DB2 Administration Server cannot be stopped.

Explanation: The DB2 Administration Server cannot be stopped because it is currently handling at least one request from a client.

User Response: Issue the DB2ADMIN STOP command with the /FORCE option to cancel all current requests being handled by the DB2 Administration Server and force the DB2 Administration Server to stop.

SQL22201N The DB2 Administration Server failed to authenticate the user *authorization-ID* **on the host** *hostname*. **Reason code** *reason-code*.

Explanation: The DB2 Administration Server was unable to authenticate the user *authorization-ID* for the following reason:

1. Invalid user ID or password.

2. Password expired.

3. User account is disabled.

4. User account is restricted.

5. The DB2 Administration Server is unable to process requests submitted as the root user.

6. Authorization failed.

User Response: Depending on the reason code, try the following:

1. Verify that a valid user ID and password were specified for the host *hostname*.

2. Change the password on host *hostname* for user *authorization-ID*. Contact your system administrator for assistance. Attempt the request again once the password has been changed.

3. Contact your system administrator to unlock the account.

4. Contact your system administrator to find out the restrictions placed on the account.

5. Resubmit the request as a user other than root.

6. An internal authentication error has occurred.

If you continue to receive this message after attempting the suggested solutions, contact DB2 Customer Support.

SQL22202W The DB2 Administration Server cannot be quiesced. Reason code *reason-code*.

Explanation: The DB2 Administration Server quiesce operation failed for one of the following reasons:

1. The DB2 Administration Server is already quiesced.

2. The DB2 Administration Server is currently processing requests on behalf of DB2 clients.

User Response: Depending on the reason for the failure, attempt one of the following:

1. No action required.

2. Wait until the DB2 Administration Server is not processing any requests, or resubmit the quiesce request and specify the force option. Forcing a quiesce will terminate any requests that are currently being processed.

SQL22203W The DB2 Administration Server cannot be unquiesced. Reason code *reason-code*.

Explanation: The DB2 Administration Server unquiesce operation failed for one of the following reasons:

1. The DB2 Administration Server is not quiesced.

2. There is at least one administration request in progress.

User Response: Depending on the reason for the failure, attempt one of the following:

1. No action required.

2. Wait until the DB2 Administration Server has completed processing all administration requests, or resubmit the unquiesce request and specify the force option. Forcing the unquiesce operation will allow normal requests to be handled concurrently with the administration requests, which may prevent the administration requests from completing successfully.

SQL22204N **The DB2 Administration Server encountered a non-severe error while executing a request.**

Explanation: A non-severe error occurred in the DB2 Administration Server while it was processing a request.

User Response: Refer to the DB2 Administration Server's First Failure Data Capture Log for additional information.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Contact DB2 Customer Support with the following required information:

- Problem description
- SQLCODE or message number
- SQLCA contents if possible
- Trace file if possible

SQL22205C **The DB2 Administration Server encountered an unexpected error while executing a request.**

Explanation: An unexpected error occurred in the DB2 Administration Server while it was processing a request.

User Response: Refer to the DB2 Administration Server's First Failure Data Capture Log for additional information.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Contact DB2 Customer Support with the following required information:

- Problem description
- SQLCODE or message number
- SQLCA contents if possible
- Trace file if possible

SQL22206N **An error occurred while accessing a message queue. Reason code** *reason-code*.

Explanation: An unexpected error or bad message was received on a message queue. 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.

User Response: Ensure the number of message queues allowed is not reached. Reduce the number of message queues in use if necessary and resubmit the request.

If you continue to receive this error message after attempting the suggested solutions, please contact DB2 Customer Support.

SQL22207N **The DB2 Administration server cannot execute the script at host** *hostname*. **Reason code** *reason-code*.

Explanation: The DB2 Administration Server failed to execute the script for one of the following reasons:

1. The user specified an existing script and the script does not exist.

2. The script working directory is invalid.

3. The last line of the script failed to run because a statement termination character could not be found.

4. A system error occurred while attempting to execute the script.

User Response: Depending on the reason for the failure, attempt one of the following:

1. Verify that the script specified exists on host *hostname* at the path provided.

2. Verify that the working directory is valid on host *hostname*.

3. Verify the script contents and resubmit the request.

4. Refer to the DB2 Administration Server's First Failure Data Capture Log for additional information.

If you continue to receive this error message after attempting the suggested response, refer to the DB2 Administration Server's First Failure Data Capture Log for additional information or contact IBM Customer Support.

SQL22208N   The DB2 Administration Server failed to execute function/method *function-name* **in library/class** *library-name* **at host** *hostname***. Reason code** *reason-code***.**

Explanation: The DB2 Administration server encountered an error while trying to execute a function/method in a library/class for one of the following reasons:

1. The library/class *library-name* could not be found.

2. The function/method *function-name* could not be found in the library/class *library-name*.

3. The version of the function/method invoked at the DB2 Administration Server is not supported.

User Response: Depending on the reason for the failure, attempt one of the following:

1. Verify that the library *library-name* exists on host *hostname*.

2. Refer to the DB2 Administration Server's First Failure Data Capture Log for additional information. If trace was active, invoke the Independent Trace Facility at the operating system command prompt. Contact DB2 Customer Support with the following required information:

   • Problem description
   • SQLCODE or message number
   • SQLCA contents if possible
   • Trace file if possible

3. See response for reason code 2.

SQL22209N   The DB2 Administration Server encountered an unexpected Java error on host *hostname***.**

Explanation: An error occurred while the DB2 Administration Server was attempting to start or communicate with a Java Interpreter. The following could be the cause of the problem:

1. Java has been improperly installed or configured on host *hostname*.

2. The DB2 Administration Server jdk_path configuration parameter has been set incorrectly.

User Response: Attempt the following:

1. Contact the system administrator for host *hostname* to verify that Java is properly installed and configured.

2. Verify that the DB2 Administration Server jdk_path configuration parameter has been set correctly. The jdk_path configuration parameter can be viewed using CLP. The parameter should point to the location where Java is installed on the host *hostname*.

If you continue to receive this error message after attempting the suggested response, please contact DB2 Customer Support.

SQL22210N   The DB2 Administration server cannot execute the specified request while quiesced.

Explanation: The DB2 Administration Server failed to execute the request because the DB2 Administration Server is currently quiesced. Only administrative requests can be executed while the DB2 Administration Server is quiesced.

User Response: Try the request again when the DB2 Administration Server is no longer quiesced. Contact your database administrator to determine when the DB2 Administration Server will be unquiesced.

SQL22211N An error occurred setting the DB2 Administration Server configuration parameter *parameter-token*. **Reason code** *reason-code*.

Explanation: The DB2 Administration Server configuration was not updated due to the following error:

1. Configuration parameter is unknown.
2. Configuration parameter value is not in the correct range.
3. A system error occurred while updating the DB2 Administration Server configuration parameters.

User Response: Depending on the reason code, verify the following:

1. The configuration parameter exists.
2. The value specified for the configuration parameter is in the correct range. See the description of the configuration parameter in the DB2 Administration Guide: Performance volume for the range of allowable values.
3. Refer to the DB2 Administration Server's First Failure Data Capture Log for additional information or contact IBM Customer Support.

SQL22212N A DB2 Administration Server communication error has been detected. Client system: *client-ip-address*. **Server system:** *server-ip-address*.

Explanation: A DB2 Administration Server communication error has been detected. Possible causes include the following:

1. The DB2 Administration Server at the server system was shut down by the system administrator.
2. The DB2 Administration Server at the server system was terminated due to an internal or system error.
3. The DB2 Administration Server is not installed at the server system.
4. The DB2 Administration Server has not been cataloged correctly at the client system.

5. The communication subsystem at the client system or server system has not been configured correctly, or has not been started successfully.
6. A network error has caused the connection to go down.
7. An internal error in the DB2 Administration Server caused the connection to go down.

User Response: Verify the following:

1. The DB2 Administration Server has not been shut down.
2. The DB2 Administration Server has not been terminated.
3. The DB2 Administration Server has been installed at the server system.
4. The remote DB2 Administration Server is cataloged properly on the client system.
5. The communication subsystems on both client and server systems are configured and started up properly.
6. The network is operating properly.
7. Nothing to verify.

If the problem persists, consult with your network administrator or contact IBM Customer Support.

SQL22213N DB2ADMIN processing was successful. The scheduler was not started successfully.

Explanation: The DB2ADMIN command successfully started the DB2 Administration Server, but the DB2 Administration Server was unable to start the scheduler.

User Response: Verify the following:

• Ensure the Tools Catalog Database configuration in the DB2 Administration Server configuration parameters is correct. For instructions on configuring the Tools Catalog Database, refer to the section on the DB2 Administration Server in the DB2 Administration Guide.

• Ensure the database manager containing the Tools Catalog Database has been started.

- If the Tools Catalog Database is remote to the Tools Catalog Instance, ensure the scheduler user ID and password are set correctly. The scheduler user ID must have SYSADM authority on the Tools Catalog Database. You can change the scheduler user ID and password using the DB2ADMIN SETSCHEDID command.
- Ensure the Tools Catalog Database is not being used by another scheduler.

After verifying the above, try stopping and then restarting the DB2 Administration Server. If you continue to receive this error message after attempting the suggested response, please contact DB2 Customer Support.

SQL22214N   The admin node *node-name* **does not exist in the DB2 node directory.**

Explanation: The admin node *node-name* is invalid. The node name does not exist in the DB2 node directory.

User Response: Verify that the node name *node-name* is cataloged in the admin node directory using the LIST ADMIN NODE DIRECTORY command. If the admin node is not listed in the admin node directory, submit a CATALOG ADMIN ... NODE command to catalog the admin node. If you continue to receive this error message after attempting the suggested response, please contact DB2 Customer Support.

SQL22215W   The DB2 Administration Server configuration parameters were updated successfully.

Explanation: The configuration parameters were updated successfully, however, you must restart the DB2 Administration Server to have the changes become effective.

User Response: Restart the DB2 Administration Server to have the changes become effective.

SQL22216N   An error occurred while performing an extended console operation. Reason code = *reason-code***.**

Explanation: While performing the operation, an unexpected error occurred. The possible reason codes are as follows:

1   The console name used to active the console has more than 8 characters.

2   The operator command that was issued is longer than the allowed 126 characters.

3   For an activate request, the issuer does not have read authority to resource name, MVS.MCSOPER.*, where * is the name of the console.

4   For an activate request, the console was already active.

User Response: The action corresponding to the reason code is:

1   Check if the user name supplied as the console name has more than 8 characters.

2   Check if the operator command is more than 126 characters in length.

3   Provide the required read authority to the resource MVS.MCSOPER.*, where * is the name of the console.

4   Ensure that the extended console you want to activate is not active.

SQL22220N   The DB2 Administration Server encountered a script error. Script error code *error-code***.**

Explanation: A script error was encountered while the DB2 Administration server was executing the script. The script exited with error code *error-code*.

User Response: Verify that the script is correct.

If you continue to receive this error message after attempting the suggested response, refer to the DB2 Administration Server's First Failure

Data Capture Log for additional information or contact IBM Customer Support.

SQL22221N   **The scheduler is not active. Reason code** *reason-code***.**

Explanation: The scheduler was unable to handle the request because the scheduler is not active for one of the following reasons:

1. The scheduler is not enabled.

2. The scheduler is not configured properly.

User Response: Check the following:

1. Verify that the scheduler has been enabled. The scheduler is enabled using the SCHED_ENABLE DB2 Administration Server configuration parameter. Verify the current setting of the SCHED_ENABLE parameter by viewing the DB2 Administration Server configuration parameters using the GET ADMIN CONFIGURATION command. To modify the value of any configuration parameter, use the UPDATE ADMIN CONFIGURATION command.

2. Verify that the scheduler is properly configured. To configure the scheduler, refer to the section on the DB2 administration server in the DB2 Administration Guide.

If you continue to receive this error message after attempting the suggested response, refer to the DB2 Administration Server's First Failure Data Capture Log for additional information or contact IBM Customer Support.

SQL22222N   **The logon user account for the Scheduler is invalid.**

Explanation: The scheduler requires a valid logon account to connect to the tools catalog database if the tools catalog database is remote to the DB2 Administration Server. This error is caused by the account not being setup, or the logon user account does not contain a valid DB2 user ID.

User Response: If a logon user account has been set up, then ensure that the account uses a valid DB2 user ID. You can set up the logon user

account for the Scheduler by using the following command:

```
DB2ADMIN SETSCHEDID <userid> <password>
```

SQL22223N   **The instance** *instance-name* **does not exist on host** *hostname***.**

Explanation: The DB2 Administration Server on host *hostname* failed to execute a request because the instance *instance-name* provided in the request does not exist on the host.

User Response: Verify the following:

• The remote instance field of the node entry in the local node directory corresponding to the instance *instance-name* is correct.

• The instance *instance-name* exists on host *hostname*.

If you continue to receive this error message after attempting the suggested response, refer to the DB2 Administration Server's First Failure Data Capture Log for additional information or contact IBM Customer Support.

SQL22230N   **The data set** *dataset* **could not be located on the system.**

Explanation: One of the following reasons may apply:

1. The data set has not been cataloged.

2. The volume on which the data set is located is not mounted.

3. The data set does not exist on the volume specified in the catalog.

4. A data set or PDS/E member name required for this operation was not specified.

5. The data set or PDS/E member name contains invalid characters or is longer than the maximum length allowed by the system.

For more information on the first three of these reasons refer to the MVS system messages IGW01021T, IGW01508T and IGW01511E.

User Response: Ensure that the name of the data set is entered correctly, that the data set has been cataloged and that the volume on which it is located is mounted on the system.

SQL22231N The requested operation is not supported for the data set *dataset*.

Explanation: Certain properties of the specified data set prevent it from being used in conjunction with the requested operation, which may include an inapplicable record format or data set type. For example, a member name cannot be specified for a PS data set.

User Response: Ensure that the data set has properties that are appropriate for the selected operation. Refer to the DB2 Administration Server's (DAS) First Failure Data Capture Log for additional information.

SQL22232N The job ID *jobid* **was not found on the Job Entry Subsystem (JES).**

Explanation: Possible reasons include:

1. The job with the specified job ID has not been submitted to the JES.
2. The job with the specified job ID has been purged from the JES.

User Response: Ensure that the job ID belongs to a job that has been submitted to the JES and that has not been purged yet.

SQL22236N A file I/O error occurred. Reason code = *reason-code*.

Explanation: An error occurred while accessing the file system. The reasons codes are as follows:

1. An invalid directory was specified.
2. An attempt was made to open a non-existent file.
3. An attempt was made to create an existing file.

User Response: The action is based on the reason code as follows.

1. Specify a valid directory.
2. Specify a file that exists.
3. Specify a file that does not already exist.

SQL22245N JCL generation failed. Reason (code[, token]) = *reason-code*.

Explanation: JCL generation failed as indicated by the following reason code:

01   A card with the reserved JCL skeleton parameter &JOB is not found or is misplaced in the main JCL skeleton. This card should be the first non-comment card in the main JCL skeleton after the TEMPLATE card.

02   Incorrect use of a reserved JCL skeleton parameter. The token in this message contains the name of the parameter that caused the problem.

03   There are more occurrences of the reserved JCL skeleton parameter &CTLSTMT in the main JCL skeleton than expected. The token in this message is set to the expected number of the &CTLSTMT parameters.

04   There are more occurrences of the reserved JCL skeleton parameter &STEPLIB in the main JCL skeleton than expected. The token in this message is set to the expected number of the &STEPLIB parameters.

05   The main JCL skeleton does not begin with the TEMPLATE statement. This statement should be the first non-comment statement in the main JCL skeleton.

06   The jobname or stepname in the JCL skeleton does not contain the JCL skeleton built-in function &SEQ required for job or step sequencing. The JCL skeleton built-in function &SEQ should be specified in the jobname or stepname of the JCL skeleton. The token in this message contains a fragment of the incorrect JCL statement.

07   The keyword JOB is not found in the first non-comment statement in the JCL skeleton for //JOB-statement. It might

be commented out, mistyped, or missed delimiting spaces, especially after the keyword.

08   The main JCL skeleton is incomplete or has incorrect structure. The required standard JCL statements may be commented out, missed or misplaced, especially if in-stream JCL procedures are being used in this JCL skeleton.

09   The syntax of jobname, stepname, or ddname in the JCL skeleton is invalid. Possible reasons for this: incorrect length of the name field, or the field includes some non-alphanumerical characters. The token in this message contains a fragment of the incorrect JCL statement.

10   An incorrect use of ampersand in the JCL skeleton. The JCL skeleton contains one or more stand-alone ampersand characters. An ampersand is the first symbol of any JCL skeleton parameter and should not be used without an accompanying identifier. The token in this message contains a fragment of the incorrect JCL statement.

11   The reserved JCL skeleton parameter &OBJECT was not found in the JCL skeleton.

12   A user-defined JCL skeleton parameter should not appear in the JCL skeleton. The token in this message contains the name of the JCL skeleton parameter that caused the problem.

13   A reserved JCL skeleton parameter should not appear in the JCL skeleton. The token in this message contains the name of the reserved JCL skeleton parameter that caused the problem.

14   The generated JCL is too long and the buffer for it cannot be allocated. The token in this message contains requested size that caused the problem. To avoid this problem, decrease the number of database objects selected for processing.

User Response: Correct the problem according to the given explanation. If the problem persists, contact your Database Administrator or DB2 Customer Support.

SQL22247N   The known discovery operation failed.

Explanation: Discovery is disabled on the target DB2 Administration Server.

User Response: Enable discovery by modifying the value of the DISCOVER configuration parameter on the DB2 Administration Server. Change the value to KNOWN if you only want known discovery enabled. Change the value to SEARCH if you want both known and search discovery enabled. To modify the value of the configuration parameter, use the UPDATE ADMIN CONFIGURATION command.

SQL22250W   Usage: DASMIGR migrates a DB2 Administration Server.

Explanation: The DASMIGR command syntax is:

```
DASMIGR <source server> <target server>
        /h
```

The command options are:

source server
         the name of the DB2 Administration Server to be migrated

target server
         the name of the new DB2 Administration Server that will contain the migrated information from the source server

/h   display the help information

User Response: Issue the DASMIGR command with one of the valid command options.

SQL22251N Migration of the DB2 Administration Server from *source-dasname* **to** *target-dasname* **failed. Reason code** *reason-code***.**

Explanation: The DB2 Administration Server *source-dasname* could not be migrated for one of the following reasons:

1. The Tools Catalog Database has not been created at the target DB2 Administration Server.

2. The Tools Catalog Database has been configured improperly at the target DB2 Administration Server.

User Response: Attempt the following at the target DB2 Administration Server *target-dasname*:

1. Verify that the Tools Catalog Database exists.

2. Verify the TOOLSCAT_DB and TOOLSCAT_SCHEMA DB2 Administration Server configuration parameters are correct. They should be set to the name and schema of the Tools Catalog Database. View the current settings of TOOLSCAT_DB and TOOLSCAT_SCHEMA using the GET ADMIN CONFIGURATION command. To modify the value of any DB2 Administration Server configuration parameter, use the UPDATE ADMIN CONFIGURATION command

If you continue to receive this error message after attempting the suggested response, please contact IBM Customer Support.

SQL22252N DAS migration failed with reason code *reason-code***.**

Explanation: An error has occurred while migrating the DB2 Administration Server. The reason codes are as follows:

1. The system resources available were insufficient to complete the migration.

2. The DB2 Administration Server's configuration parameters do not identify a valid tools catalog.

3. A non-severe error occurred during migration.

User Response:

1. Verify that enough system resources are available for DAS migration.

2. Verify that a tools catalog is created and is identified correctly by the DB2 Administration Server's configuration parameters.

3. Refer to the DB2 Administration Server's First Failure Data Capture Log for additional information.

SQL22255W Usage:

dasauto [-h|-?] -on|-off

Explanation: An incorrect argument was entered. Valid arguments for this command are:

-h|-?   display the usage information

-on|-off
        enables or disables autostarting of the DB2 Administration Server

User Response: Enter the command again as follows:

dasauto [-h|-?] -on|-off

SQL22256W The dasauto command was successful.

Explanation: All processing completed normally.

User Response: No further action is required.

SQL22270N The contact, or contact group, with name *name* **cannot be added to the contact list.**

Explanation: The contact, or contact group, already exists in the contact list.

User Response: Create a new contact, or contact group, with a unique name.

**SQL22271N The contact, or contact group,
information with name** *name*
could not be found.

Explanation: The contact, or contact group,
could not be found in the contact list.

User Response: Check and correct the name,
and try again.

SQL22272N The record with key *key* **cannot be
inserted.**

Explanation: The system tried to insert a record
and the new record's key *key* already exists in
the system.

User Response: Remove the duplicate from the
system; or, insert a new record with another key.

SQL22273N The record with key *key* **could not
be found.**

Explanation: The record with key *key* could not
be found because the record does not exist.

User Response: Check and correct the key, and
try again.

**SQL22280N This action cannot be performed
because the scheduler is quiesced.**

Explanation: No action can be performed until
the scheduler is activated again.

User Response: Activate the scheduler.

SQL22281N The scheduler is already active.

Explanation: The scheduler cannot be made
active because it is already active.

User Response: No further action is required.

**SQL22282N The scheduler is unable to access
the Tools Catalog Database.
Reason code =** *reason-code* **and
SQLCODE** *sqlcode***.**

Explanation: The scheduler function cannot be
processed because the scheduler is unable to
access the Tools Catalog Database for the reason

specified by the *reason-code* as follows:

1. The given user ID, or password, is incorrect.

2. Incorrect Tools Catalog Database
   configuration in the DB2 Administration
   Server configuration parameters. One or more
   of the following DB2 Administration Server
   configuration parameters is incorrect:
   - TOOLSCAT_DB
   - TOOLSCAT_SCHEMA

3. The Tools Catalog Database is already being
   used by another scheduler on the same
   system.

4. The Tools Catalog Database is only supported
   through TCP/IP as a communication
   protocol.

5. Unexpected errors. Refer to SQLCODE *sqlcode*
   for more details.

User Response: The action corresponding to the
reason code is:

1. Check your user ID and password and try
   again.

2. Verify that the DB2 Administration Server
   TOOLSCAT_DB and TOOLSCAT_SCHEMA
   configuration parameters have been set
   correctly. View the current settings of
   TOOLSCAT_DB and TOOLSCAT_SCHEMA
   using the GET ADMIN CONFIGURATION
   command. To modify the value of any DB2
   Administration Server configuration
   parameter, use the UPDATE ADMIN
   CONFIGURATION command. If you
   continue to receive this error message after
   attempting the suggested response, please
   contact IBM Support.

3. Contact your system administrator.

4. Attempt the following:
   - If the tools catalog database is local to the
     scheduler, then set up the instance
     specified by the DB2 Administration Server
     configuration parameter toolscat_inst to
     use TCP/IP communications.
   - If the tools catalog database is remote to
     the scheduler, then ensure that the node
     used by the toolscat_db admin
     configuration parameter supports TCP/IP
     as the communication protocol.

5. Refer to the DB2 Administration Server's First Failure Data Capture Log for additional information.

Contact IBM Support if you continue to receive this message after attempting the suggested solutions.

SQL22283N Task *taskid.suffix* **does not exist in the Tools Catalog Database.**

Explanation: The scheduler cannot retrieve the task from the Tools Catalog Database. The task might have been removed from the Tools Catalog Database or never existed.

User Response: Check for the existence of the task using the Task Center.

SQL22284N Task *taskid.suffix* **is not scheduled.**

Explanation: The requested action, Run a scheduled task now, cannot be performed because the task is not currently scheduled to run.

User Response: Attempt one of the following:

• Perform a Run now instead

• Activate all of the task's schedules and perform Run a scheduled task now again.

SQL22285N **The scheduler is unable to get the corresponding hostname from the partitioned database on partition number** *partition-number*.

Explanation: The requested partition number does not exist in the partitioned database.

User Response: Verify that the partition number exists by displaying the list of nodes using the LIST DBPARTITIONNUMS command.

SQL22286N **Unable to execute the required task.**

Explanation: The scheduler failed to run the task because an unexpected error occurred.

User Response: Refer to the DB2 Administration Server's First Failure Data

Capture Log for additional information.

If trace was active, invoke the Independent Trace Facility at the operating system command prompt.

SQL22287N **User** *userid* **does not have permission to execute Task** *taskid.suffix*.

Explanation: The task failed to execute because the user does not have sufficient authority to run the task.

User Response: Users must be granted run permission by the owner of the task before they can run the task.

SQL22295N **Unable to communicate successfully with the SMTP server at host** *host-name* **using port** *port-number*. **Reason code =** *reason-code*.

Explanation: The SMTP server at host *host-name* using port *port-number* failed to successfully negotiate the protocol to send mail. An explanation of the error is indicated by the following reason code:

1. There is no SMTP server running at *host-name* on port *port-number*.

2. An unexpected error was encountered while communicating with the SMTP server.

User Response:

1. Verify that the SMTP server name is specified correctly and that the SMTP port specified in the services file corresponds to the port number being used by the SMTP server.

2. Invoke the Independent Trace Facility at the operating system command prompt. Contact IBM Customer Support if the problem persists.

SQL22296N **Unable to send mail using the SMTP protocol due to invalid recipient addresses.**

Explanation: The SMTP server failed to successfully negotiate the send mail protocol for

all the specified recipients.

User Response: Verify that the recipient addresses are specified correctly.

This error might also be returned if the SMTP server is unable to reach the recipient addresses. This can be independently verified by using any other mail client to send mail to the recipients using the same SMTP server. If this is successful, invoke the Independent Trace Facility at the operating system command prompt. Contact IBM Customer Support if the problem persists.

SQL22297I   Configuration changes will not be effective until the DB2 Administration Server is restarted.

Explanation: The DB2 Administration Server has been successfully configured, but the changes

will not take effect immediately. The changes will take effect when the DB2 Administration Server is restarted.

User Response: To have the change take effect at the next DB2 Administration Server restart, no further action is required.

SQL27900 - SQL27999

SQL27902N   LOAD RESTART/TERMINATE is not allowed on a table that is not in LOAD PENDING state.

Explanation: No RESTART or TERMINATE is necessary. The RESTART and TERMINATE modes of the LOAD utility should be used to resume or cancel a previously failed/interrupted LOAD operation. They are only supported on tables where a previous load operation failed and the table is in LOAD PENDING state. For tables that are not in LOAD PENDING state, only the INSERT and REPLACE modes of the LOAD utility are supported.

User Response: To populate the table with the contents of the input source re-issue your LOAD command using the INSERT or REPLACE mode. You can use the LOAD QUERY utility to find out the current state of the table.

SQL27903I   *agenttype* has started on partition *partitionnumber* at time *timestamp*.

Explanation: This is an informational message indicating that a DB2 agent is about to begin on a given partition.

User Response: No action is required.

SQL27904I   The input file was not found. Using standard input as the input.

Explanation: This is an informational message indicating that an DB2 agent is about to begin on a given partition.

User Response: No action is required.

SQL27907I   Maximum input record length *record-length*.

Explanation: This is an informational message indicating the record length used for the Load operation.

User Response: No action is required.

SQL27908I   Program is running with check level: *check-level*.

Explanation: This is an informational message indicating whether truncation checking is enabled or not.

User Response: No action is required.

SQL27909I   Tracing *numberofrecords*
non-delimited record(s).

Explanation: This is an informational message
indicating that tracing is enabled for the first
numofrecords record(s).

User Response: No action is required.

SQL27910I   The string delimiter is *chardel***, the
column delimiter is** *coldel***, and the
decimal point is** *decept***. Tracing**
numofrecords **delimited record(s).**

Explanation: This is an informational
message indicating what delimiters are used for string,
column and decimal point. In addition, this
message also shows tracing is enabled for the
first *numofrecords* record(s).

User Response: No action is required.

SQL27911I   The input partition map file
filename **was opened successfully
for reading.**

Explanation: This informational message
indicates the input partition map file was opened
successfully for reading.

User Response: No action is required.

**SQL27912I   Reading of input partitioning map
is in progress.**

Explanation: This informational message
indicates that reading of the input partition map
file is in progress.

User Response: No action is required.

**SQL27913I   Input partitioning map was
successfully read.**

Explanation: This informational message
indicates that the input partition map file was
successfully read.

User Response: No action is required.

SQL27914I   The mode of operation is *mode***.**

Explanation: This informational message
indicates the mode of operation for the
partitioning agent.

There are two different modes of operation:

- Partition
- Analyze

User Response: No action is required.

**SQL27915I   Output partitioning map file not
used.**

Explanation: This is an informational message
indicating that the output partitioning map file is
not used.

User Response: No action is required.

SQL27916I   The output partitioning map file
filename **was opened successfully.**

Explanation: This is an informational message
indicating that the output partitioning map file
was opened successfully.

User Response: No action is required.

SQL27918I   Distribution file name: *filename***.**

Explanation: This is an informational message
indicating the name of the distribution file used
in the utility.

User Response: No action is required.

SQL27919I   The distribution file *filename* **was
opened successfully for writing.**

Explanation: This informational message
indicates the distribution file was opened
successfully for writing.

User Response: No action is required.

SQL27920I   **This utility is using** *numkeys* **partitioning keys.**

Explanation: This informational message indicates the number of partitioning keys used in the utility.

User Response: No action is required.

SQL27921I   *keyname* **Start:***index* **Len:***length* **Position:***position* **Type:***type***.**

Explanation: This informational message describes attributes of a partitioning key.

User Response: No action is required.

SQL27922I   **Output data files are not created because the run type is ANALYZE.**

Explanation: This is an informational message indicating that the output data files are not created because the mode of operation is ANALYZE.

User Response: No action is required.

SQL27926I   **Output data for partition** *partitionnum* **will be sent to standard output.**

Explanation: This is an informational message indicating that the output data on partition *partitionnum* will be sent to standard output.

User Response: No action is required.

SQL27927I   **Processed** *numofrecords* **records (or lines).**

Explanation: This is an informational message indicating that *numofrecords* records (or lines) were processed.

User Response: No action is required.

SQL27928I   **Processing record number** *num***.**

Explanation: This informational message indicates which record is being processed. This

message will only be reported when the TRACE option is specified.

User Response: No action is required.

SQL27929I   **Partitioning number returned from hash function:** *hexvalue* **(hex)** *decvalue* **(decimal).**

Explanation: This informational message reports the result of the hashing function in both hex and decimal format.

User Response: No action is required.

SQL27931I   **Writing output partition map to file** *filename***.**

Explanation: This is an informational message indicating that the output partition map is written to *filename*.

User Response: No action is required.

SQL27932I   **Writing distribution map to** *filename***.**

Explanation: This is an informational message indicating that the distribution map is written to *filename*.

User Response: No action is required.

SQL27935I   *agenttype* **has ended on partition** *partitionnumber* **at time** *timestamp***.**

Explanation: This is an informational message indicating that an DB2 agent has just completed operation on a given partition.

User Response: No action is required.

SQL27936I   **Elapsed time:** *hr* **hours,** *min* **minutes,** *sec* **seconds.**

Explanation: This informational message reports the total time elapsed.

User Response: No action is required.

SQL27937I Throughput: *throughput*
records/sec.

Explanation: This is an informational message indicating the throughput of a given DB2 agent.

User Response: No action is required.

SQL27939I Record counts for output partitions: partition number *partitionnum*. **Record count:** *numofrecords*.

Explanation: This informational message indicates the number of records processed for a given partition.

User Response: No action is required.

SQL27941I Program terminated abnormally.

Explanation: This informational message indicates the utility program has terminated abnormally.

User Response: No action is required.

SQL27942I There are *numofwarnings* **warning message(s) and** *numofdisrec* **discarding records.**

Explanation: This informational message reports there are *numofwarn* warning message(s) and *numofdisrec* discarding record(s) encountered during the operation.

User Response: No action is required.

SQL27945I Key Index: *index*. **Data:** *data1 data2 data3 data4 data5*.

Explanation: This informational message reports the attributes of a given partition key.

User Response: No action is required.

SQL27947I The input data file specified at line *linenum* **of the configuration file is ignored.**

Explanation: The input data file specified at line *linenum* of the configuration file is ignored. The

command line option will be used if it is specified; otherwise, the first specification of input data file in the configuration file will be used.

User Response: No action is required.

SQL27948I The output data file specified at line *linenum* **of the configuration file is ignored.**

Explanation: The output data file specified at line *linenum* of the configuration file is ignored. The command line option will be used if it is specified; otherwise, the first specification of output data in the configuration file will be used.

User Response: No action is required.

SQL27949I The distribution data file specified at line *linenum* **of the configuration file is ignored.**

Explanation: The distribution file specified at line *linenum* of the configuration file is ignored. The command line option will be used if it is specified; otherwise, the first specification of distribution file in the configuration file will be used.

User Response: No action is required.

SQL27950I The type of the input data file is *filetype*.

Explanation: This is an informational message indicating that the format of the input data is *filetype*.

- 0-ASC
- 1-DEL
- 2-BIN
- 3-CUR

User Response: No action is required.

SQL27951I The NEWLINE flag is ignored in case of binary or delimited input data file

Explanation: This informational message reports that the NEWLINE flag is ignored for binary or

delimited input data files.

User Response: No action is required.

SQL27952I   The NEWLINE flag is turned on.

Explanation: This informational message reports that the NEWLINE flag is in effect.

User Response: No action is required.

SQL27953I   Usage: db2split
                        [-c configuration-file-name]
                        [-d distribution-file-name]
                        [-i input-file-name]
                        [-o output-file-name]
                        [-h help message]

Explanation:

-c   option will run this program using a
           user specified configuration file

-d   option specifies the distribution file

-i   option specifies the input file

-o   option specifies the output file

-h   option will generate the help message

SQL27959N   The partitioned database
                        configuration option *option-name* **is**
                        invalid. Reason code = *reason-code*.

Explanation: The partitioned database configuration option named in the error message is incorrectly specified or is incompatible with one of the other load options specified.

The possible reason codes are as follows:

1   Partitioned database configuration
           options cannot be specified in a
           non-partitioned database environment
           or when the
           DB2_PARTITIONEDLOAD_DEFAULT
           registry variable is set to OFF.

2   Partitioned database configuration
           options may not be specified more than
           once.

3   Invalid pointer detected in the
           piPartLoadInfoIn input structure passed
           to the db2Load API.

4   Invalid pointer detected in the
           poPartLoadInfoOut output structure
           passed to the db2Load API.

5   The argument supplied to the MODE
           option must be one of the following:
- PARTITION_AND_LOAD
- PARTITION_ONLY
- LOAD_ONLY
- LOAD_ONLY_VERIFY_PART
- ANALYZE

6   The maximum number of partitioning
           agents must be less than or equal to the
           maximum number of partitions
           allowable in a cluster.

7   The maximum number of partition
           numbers in a partition list must be less
           than or equal to the maximum number
           of partitions allowable in a cluster.

8   The arguments supplied to the
           ISOLATE_PART_ERRS option must be
           one of the following:
- SETUP_ERRS_ONLY
- LOAD_ERRS_ONLY
- SETUP_AND_LOAD_ERRS
- NO_ISOLATION

9   The value supplied to the
           STATUS_INTERVAL option must be in
           the range 1-4000.

10   The maximum port number must be
           greater than or equal to the minimum
           port number.

11   The only legal arguments for the
           CHECK_TRUNCATION, NEWLINE and
           OMIT_HEADER options are TRUE and
           FALSE.

12   The argument supplied to
           RUN_STAT_DBPARTNUM must be a
           legal partition number.

13   If the mode is ANALYZE, the
           MAP_FILE_OUTPUT option must be
           specified.

14   If the mode is PARTITION_ONLY the
           PART_FILE_LOCATION option must be
           specified. If the mode is LOAD_ONLY
           or LOAD_ONLY_VERIFY_PART, the
           PART_FILE_LOCATION option must be
           specified if the input file name is not
           fully qualified.

15   The load actions RESTART and
           TERMINATE can only be used when the
           mode is PARTITION_AND_LOAD,
           LOAD_ONLY or
           LOAD_ONLY_VERIFY_PART.

16   The HOSTNAME option cannot be
           specified unless the
           FILE_TRANSFER_CMD option is also
           specified.

17   The partition isolation error modes
           LOAD_ERRS_ONLY and
           SETUP_AND_LOAD_ERRS are not
           compatible with the ALLOW READ
           ACCESS or COPY YES options of the
           load command.

18   The LOAD_ONLY and
           LOAD_ONLY_VERIFY_PART modes are
           not compatible with the CLIENT option
           of the load command.

User Response: Resubmit the LOAD command
with the correct partitioned database
configuration options.

**SQL27960N   PART_FILE_LOCATION must
               specify a fully qualified base
               filename when the file type is
               CURSOR.**

Explanation: When loading from an input file
of type CURSOR, the PART_FILE_LOCATION
option must specify a fully qualified base
filename instead of a directory.

User Response: Resubmit the LOAD command
with a proper argument for the
PART_FILE_LOCATION partitioned database
configuration option.

**SQL27961N   ROWCOUNT cannot be specified
               in a partitioned database
               environment unless the mode is
               ANALYZE.**

Explanation: The ROWCOUNT option of the
load command is not supported except when the
mode is ANALYZE.

User Response: Resubmit the load command
without the ROWCOUNT option.

**SQL27990W   During loading, at least one row
               was found to be on the wrong
               partition.**

Explanation: When loading a partitioned
database using the load utility with
LOAD_ONLY_VERIFY_PART mode, any rows in
the input file that are found to be on the wrong
partition will be discarded. If the "dumpfile"
modifier is specified, the discarded rows will be
saved to the dump file. This message will appear
only once per partition per load job, even when
there are multiple partition violations present.

User Response: If the discarded rows are saved
to a dump file, they can be loaded to the correct
partition by simply issuing another load
command with the MODE option set to
PARTITION_AND_LOAD.

**SQL27991W   Too many partitioning agents
               specified in the load command.**

Explanation: Too many partitioning agents have
been specified in the
PARTITIONING_DBPARTNUMS partitioned
database configuration option of the load
command. The maximum number of partitioning
agents is specified by the
MAX_NUM_PART_AGENTS load option, which
defaults to 25 if not specified.

User Response: Resubmit the command with
the correct number of partitioning agents.
Alternatively, the
PARTITIONING_DBPARTNUMS option can be
omitted and the load utility will automatically
select an appropriate number of partitioning
agents.

SQL27992N **The data file contains a valid partitioning map, but the load mode is PARTITITON_AND_LOAD.**

Explanation: Load has detected a valid partitioning map at the beginning of the file. PARTITION_AND_LOAD mode would treat this as user data, possibly causing undesired results.

User Response: If the reason the data file begins with a partitioning map header is that the data in the file has already been partitioned, then use LOAD_ONLY mode to load the file. In this case it will also be necessary to use the OUTPUT_DBPARTNUMS option to specify the single partition to which the data should be loaded.

If the partitioning map detected by load at the beginning of the file is actually user data, then use PARTITION_AND_LOAD mode with the IGNOREHEADER modifier to disable the partition map detection and load the entire data file.

SQL29000 - SQL29099

SQL29000N **The value specified for DYN_QUERY_MGMT is not valid. DB2 Query Patroller is not installed on this server.**

Explanation: The attempt to update the database configuration parameter DYN_QUERY_MGMT to ENABLE failed because the DB2 Query Patroller server is not installed.

User Response: Install the DB2 Query Patroller server.

SQL29001N **This database client level is not compatible with the level of DB2 Query Patroller server you are running on.**

Explanation: The client and server code are not compatible.

SQL29002N **DB2 Query Patroller is not installed on this server.**

Explanation: DB2 Query Patroller server tables do not exist on the server.

User Response: Install the DB2 Query Patroller server.

SQL29003N **DB2 Query Patroller could not load Java class** *class-name*, **reason code** *reason-code*.

Explanation: An error occurred while trying to load the Java class *class-name*. The reason codes are:

1 The class was not found on the CLASSPATH.

User Response: Ensure the *class-name* is installed in the CLASSPATH.

SQL29004N **DB2 Query Patroller class** *class-name* **cannot call method** *method-name* **with signature** *signature*.

Explanation: The Java method *method-name* cannot be found.

User Response: Ensure that the correct version of the DB2 Query Patroller client is installed.

SQL29005N **User** *user-ID* **is not defined in the user profile table.**

Explanation: User *user-ID* is not defined in the user profile table. The user, or the group to which the user belongs, must be defined in the user profile table.

User Response: Use Query Administration to define the user in the user profile table.

SQL29006N **Java exception** *exception-string* **was thrown.**

Explanation: Java exception *exception-string* was thrown.

User Response: If the problem persists contact your technical service representative.

SQL29007N **DB2 Query Patroller Server is not available.**

Explanation: DB2 Query Patroller server is not running.

User Response: Start DB2 Query Patroller server and resubmit your query.

SQL29008N **Error encountered during job sequence number generation.**

Explanation: An error was encountered during job sequence number generation.

User Response: If the problem persists contact your technical service representative.

SQL29009N **Error encountered when submitting a new job. Reason code** *reason-code***.**

Explanation: An error was encountered during submission of a new job.

User Response: Examine the reason code and resubmit the query.

SQL29010N **The query has been cancelled.**

Explanation: User has cancelled the query.

SQL29011I **Job** *job-ID* **has been submitted.**

Explanation: Job *job-ID* has been submitted to the DB2 Query Patroller server.

SQL29012N **Job** *job-ID* **has been aborted.**

Explanation: Job *job-ID* has been aborted.

User Response: Use QueryMonitor or the iwm_cmd command to examine the reason for the aborted job and resubmit the query.

SQL29013I **Job** *job-ID* **has been held.**

Explanation: Job *job-ID* has been put on hold.

SQL29014N **The value specified for DB2 registry variable** *registry-variable* **is invalid.**

Explanation: The value specified for *registry-variable* is invalid because of the following restrictions by *registry-variable*:

DQP_NTIER
>   The value must be one of OFF, RUN[:timeout], or CHECK[:timeout].

DQP_LAST_RESULT_DEST
>   The length must not exceed 32 characters.

DQP_TRACEFILE
>   The length must not exceed 256 characters.

User Response: Correct the value of the DB2 registry variable *registry-variable* with the db2set command and resubmit the query.

SQL29015N **Error encountered during job cancellation. Reason code:** *reason-code***.**

Explanation: An error was encountered while trying to cancel a job. The reason code maps to an SQL or DB2 message.

User Response: Examine the reason code, correct the error, and retry the action again.

SQL29016N **The data source on the client does not match the data source on the DB2 Query Patroller server.**

Explanation: The data source to which the client is connected does not match the data source that is used by the DB2 Query Patroller server.

User Response: Ensure that the DB2DBDFT profile variable on the DB2 Query Patroller server matches the client data source name.

SQL29017N   Job *job-id* **has been cancelled.**

Explanation: Job *job-id* has been cancelled.

User Response: None.

SQL29018N   The DB2 Query Patroller Client is not installed.

Explanation: The user is querying a database which has enabled the database configuration parameter DYN_QUERY_MGMT. However, the DB2 Query Patroller client code is not installed on the client machine.

User Response: Install the QueryEnabler component of the DB2 Query Patroller client.

SQL29019N   Jobs have been scheduled on the DB2 Query Patroller server.

Explanation: Jobs have been scheduled on the DB2 Query Patroller server.

User Response: None.

SQL30000 - SQL30099

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)*.

Explanation: 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.

User Response: 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.

Required information:
- Problem description
- SQLCODE and reason code
- SQLCA contents if possible
- Trace file if possible.

sqlcode: -30000

sqlstate: 58008

SQL30002N   The SQL statement cannot be executed due to a prior condition in a chain of statements.

Explanation: An SQL statement was chained to PREPARE, but the PREPARE statement has received a warning SQLCODE that requires the program or end user to either re-issue the chained statement or issue a different SQL request. This error can occur only in a client/server environment.

- A distributed client using DRDA has chained an OPEN statement to a PREPARE, but the PREPARE statement received SQLCODE +1140.

The statement cannot be executed as chained.

User Response: The statement must be sent again as a separate request.

sqlcode: -30002

sqlstate: 57057

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***.**

Explanation: 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.

Reason code 124C indicates that the DRDA data stream for the request contains a syntax error.

The command or statement cannot be processed. The current transaction is rolled back and the application is disconnected from the remote database.

User Response: 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.

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.

1254 A command sent from the client has resulted in an unarchitected and implementation specific condition on the remote server for which there is no architected message to return to the client.

If the remote server is DB2 UDB for OS/390,zOS then check the console log for information about this error.

If the remote server is DB2 UDB for iSeries, the job log of the server job, and/or a first failure data capture (FFDC) spooled file is usually necessary to determine the cause of the error.

If the remote server is DB2 UDB for Unix/Windows then check the remote database server's administration notification log for information regarding this error.

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 DB2 Connect 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.**

Explanation: 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.

User Response: 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. 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*.

Explanation: 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.

User Response: 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*.

Explanation: The application cannot process the command or SQL statement because resources are not available at the remote database. This SQLCODE is produced at statement compilation or processing time.

Federated system users: this situation can also be detected by the data source.

If the *resource name* is "Encrypt" or "Decrypt", the user exits for password encryption or decryption are unavailable or had an error.

The command or statement cannot be processed.

User Response: Investigate the remote database system environment.

Federated system users:

- If the password encryption or decryption user exit failed, *reason* is the integer value that the federated server received when invoking the user exit. Ensure that no errors occurred when the encryption and decryption user exits were link-edited with the federated server.

  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 the federated server, and reissue 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 reissue the failing command or statement.

sqlcode: -30041

sqlstate: 57013

SQL30050N *number* **command or SQL statement invalid while bind is in progress.**

Explanation: 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.

User Response: 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.

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.**

Explanation: 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.

User Response: 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.

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.**

Explanation: The SQL statement being compiled is not recognized by the precompiler and cannot be processed by the database.

The statement cannot be processed.

User Response: 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.

Explanation: 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.

User Response: 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***.**

Explanation: 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.

User Response: 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.**

Explanation: The database name is not an existing database at the remote database node.

The statement cannot be processed.

User Response: Resubmit the command with the correct database name or alias.

Federated system 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.**

Explanation: 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.

User Response: 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.

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.**

Explanation: 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.

User Response: 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,

If trace was active, invoke the Independent Trace Facility at the operating system command prompt.

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.**

Explanation: 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.

User Response: 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.

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.**

Explanation: 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.

User Response: 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. 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 DB2 Connect 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.**

Explanation: 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.

User Response: 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.

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.**

Explanation: 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 Guide (SC31-6156)*.

User Response: Ensure that both the database manager and Communications Manager are started at the Database Server and that all Communications Manager configuration parameters are correct.

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***.**

Explanation: 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.
- 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 name resolution 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 administration notification log to see if an error message has been logged.
  - The connection may have been closed by the remote server at the TCP/IP level.
  - 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.

- 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.

- NETBIOS

  <rc1> contains the return code from the call to NetBIOS. <rc2> and <rc3> are not applicable.

- 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.

See the Communications Errors appendix of the *Message Reference* for more information on specific communication error codes.

User Response:

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 *Message 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 administration notification log for more information.

6. A communication subsystem or network error has occurred.

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***).**

Explanation: 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.

16 (NEW PASSWORD INVALID)
> the password specified on a change password request did not meet the server's requirements.

17 (UNSUPPORTED FUNCTION)
> the security mechanism specified by the client is invalid for this server. Some typical examples:
> - The client sent a new password value to a server that does not support the change password function.

- The client sent SERVER_ENCRYPT authentication information to a server that does not support password encryption.
- The client sent a userid (but no password) to a server that does not support authentication by userid only.
- The client has not specified an authentication type, and the server has not responded with a supported type. This may include the server returning multiple types from which the client is unable to choose.

18 (NAMED PIPE ACCESS DENIED)
> The named pipe is inaccessible due to a security violation.

19 (USERID DISABLED or RESTRICTED)
> The userid has been disabled, or the userid has been restricted from accessing the operating environment at this time.

20 (MUTUAL AUTHENTICATION FAILED)
> The server being contacted failed to pass a mutual authentication check. The server is either an imposter, or the ticket sent back was damaged.

21 (RESOURCE TEMPORARILY UNAVAILABLE)
> Security processing at the server was terminated because a resource was temporarily unavailable. For example, on AIX, no user licenses may have been available.

24 (USERNAME AND/OR PASSWORD INVALID)
> The username specified, password specified, or both, are invalid.

User Response: Ensure that the proper userid and/or password is supplied.

The userid may be disabled, the userid may be restricted to accessing specific workstations, or the userid may be restricted to certain hours of operation.

For reason code 17, retry the command with a supported authentication type.

For reason code 20, make sure the authentication mechanism for the server is started, and retry.

sqlcode: -30082

sqlstate: 08001

SQL30083N Attempt to change password for user id *uid* **failed with security reason** *reason-code* **(***reason-string***).**

Explanation: The attempt to change password 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* values.

The following is a list of reason codes and corresponding reason strings:

0 (NOT SPECIFIED)
> The specific security error is not specified.

1 (CURRENT PASSWORD INVALID)
> The old password specified in the request is not valid.

2 (NEW PASSWORD INVALID)
> The password specified in the request is not valid under the password rules imposed by the system where password was to be changed.

3 (CURRENT PASSWORD MISSING)
> The request did not include an old password.

4 (NEW PASSWORD MISSING)
> The request did not include a new password.

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. Passwords can not be changed for revoked userids.

14 (INSTALLATION EXIT FAILED)
> The installation security exit failed.

15 (PROCESSING FAILURE)
> Security processing at the server failed.

17 (UNSUPPORTED FUNCTION)
> Change password function is not supported by the system.

19 (USERID DISABLED or RESTRICTED)
> The userid has been disabled, or the userid has been restricted from accessing the operating environment at this time.

23 (CHGPWD_SDN in DCS entry is not configured)
> To change an MVS password on a host system connected via SNA, the DCS database must be cataloged with the ,,,,,,,CHGPWD_SDN parameter string. The ,,,,,,,CHGPWD_SDN parameter string identifies the symbolic destination name for Password Expiration Management (PEM).

24 (USERNAME AND/OR PASSWORD INVALID)
> The username specified, password specified, or both, are invalid.

User Response: Ensure that the proper userid, current and new passwords are supplied.

The userid may be disabled, the userid may be restricted to accessing specific workstations, or the userid may be restricted to certain hours of operation.

Instructions for some specific reason codes follow:

14   Check the file db2pem.log in the instance subdirectory (typically "db2") for a detailed description of the problem encountered.

23   Catalog the DCS database using the ,,,,,,,CHGPWD_SDN parameter, as specified in the *DB2 Connect User's Guide*.

sqlcode: -30083

sqlstate: 08001

SQL30090N **Operation invalid for application execution environment. Reason code =** *reason-code*

Explanation: The operation is invalid for the application execution environment. For example, an operation might be invalid for applications that have special restrictions on statements or APIs - applications such as those that operate in an XA Distributed Transaction Processing environment, such as CICS; those that operate with CONNECT type 2 connection settings; or those that use federated system functionality to update multiple heterogeneous data sources. The operation 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
- federated server 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 insert/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:

- Fujitsu RDB2 requires the existence of a unique index.

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 federated server 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 SET CLIENT INFORMATION was issued while a Unit of Work is in progress.

21 The operation that you want performed at the specified data source(s) is not supported by the wrapper by which DB2 accesses the data source(s). Consult the documentation to find out what operations this wrapper supports.

22 A federated insert, update, or delete operation is invalid in an application execution environment where a

- SAVEPOINT is in effect
- scrollable cursor is used
- target view contains multiple tables or nicknames
- trigger is in effect.

23 The API, data structure, or setting is not supported.

User Response: Correct the problem by doing one of the following steps:

For reasons 01, 02, 03, 04, 06, or 19,
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 (federated server users), isolate the problem to the data source failing the request (see the Troubleshooting 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 *SQL Reference* 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.

For reason 20,
>submit a Commit or Rollback before invoking the API.

For reason 22,
>Either remove the federated insert, update or delete operation or:
>* release the savepoint that is in effect
>* remove the use of the scrollable cursor
>* redefine the target view to refer to only one table or nickname
>* execute the federated insert, update, or delete operation outside of the trigger.

For reason 23,
>Refer to the Administrative API Reference for a list of supported APIs, data structures, and settings.

sqlcode: -30090

sqlstate: 25000

SQL30100 - SQL30199

SQL30101W Bind options specified on REBIND request are ignored.

Explanation: 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.

User Response: 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

SQL30104N Error in BIND or PRECOMPILE option *option-name* **with value** *value-name***.**

Explanation: While processing the BIND or PRECOMPILE parameters, either the BIND or PRECOMPILE option or the value is not acceptable, or the option and value pair is not appropriate.

The statement cannot be processed.

User Response: Examine the command options and values to determine the error and resubmit the command.

sqlcode: -30104

sqlstate: 56095

SQL30106N Invalid input data detected for a multiple row INSERT operation. INSERT processing is terminated.

Explanation: An error was detected with the input data for one row of a multiple row INSERT operation. No further rows will be inserted. For an atomic operation, all inserted rows are rolled back. For a non-atomic operation, rows inserted successfully before the row containing the invalid input data was encountered are not rolled back.

User Response: Correct the row containing the invalid input data and submit the multiple row

INSERT again for the rows that did not get inserted.

sqlcode: -30106

sqlstate: 22527

Chapter 3. SQLSTATE Messages

This section lists the SQLSTATEs and their meaning. The SQLSTATEs are grouped by class code; for subcodes, see the corresponding tables.

Table 1. SQLSTATE Class Codes

Class Code	Meaning	For subcodes, refer to...
00	Unqualified Successful Completion	Table 2 on page 556
01	Warning	Table 3 on page 556
02	No Data	Table 4 on page 560
07	Dynamic SQL Error	Table 5 on page 560
08	Connection Exception	Table 6 on page 561
09	Triggered Action Exception	Table 7 on page 561
0A	Feature Not Supported	Table 8 on page 561
0D	Invalid Target Type Specification	Table 9 on page 562
0F	Invalid Token	Table 10 on page 562
0K	Invalid RESIGNAL Statement	Table 11 on page 562
20	Case Not Found for CASE Statement	Table 12 on page 562
21	Cardinality Violation	Table 13 on page 562
22	Data Exception	Table 14 on page 563
23	Constraint Violation	Table 15 on page 564
24	Invalid Cursor State	Table 16 on page 565
25	Invalid Transaction State	Table 17 on page 565
26	Invalid SQL Statement Identifier	Table 18 on page 566
28	Invalid Authorization Specification	Table 19 on page 566
2D	Invalid Transaction Termination	Table 20 on page 566
2E	Invalid Connection Name	Table 21 on page 567
34	Invalid Cursor Name	Table 22 on page 567
36	Cursor Sensitivity Exception	Table 23 on page 567
38	External Function Exception	Table 24 on page 567
39	External Function Call Exception	Table 25 on page 568
3B	Invalid SAVEPOINT	Table 26 on page 569
40	Transaction Rollback	Table 27 on page 569

Table 1. SQLSTATE Class Codes (continued)

Class Code	Meaning	For subcodes, refer to...
42	Syntax Error or Access Rule Violation	Table 28 on page 569
44	WITH CHECK OPTION Violation	Table 29 on page 582
46	Java DDL	Table 30 on page 582
51	Invalid Application State	Table 31 on page 582
53	Invalid Operand or Inconsistent Specification	Table 32 on page 583
54	SQL or Product Limit Exceeded	Table 33 on page 584
55	Object Not in Prerequisite State	Table 34 on page 585
56	Miscellaneous SQL or Product Error	Table 35 on page 586
57	Resource Not Available or Operator Intervention	Table 36 on page 587
58	System Error	Table 37 on page 588

Class Code 00 Unqualified Successful Completion

Table 2. Class Code 00: Unqualified Successful Completion

SQLSTATE Value	Meaning
00000	Execution of the operation was successful and did not result in any type of warning or exception condition.

Class Code 01 Warning

Table 3. 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.
01007	A privilege was not granted.
0100C	One or more adhoc result sets were returned from the procedure.
0100D	The cursor that was closed has been re-opened on the next result set within the chain.

Table 3. Class Code 01: Warning (continued)

SQLSTATE Value	Meaning
0100E	The procedure generated more than the maximum allowed number of result sets. Only the first integer result sets have been returned to the caller.
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.
01515	The null value has been assigned to a host variable, because the non-null value of the column is not within the range of the host variable.
01516	An inapplicable WITH GRANT OPTION has been ignored.
01517	A character that could not be converted was replaced with a substitute character.
01519	The null value has been assigned to a host variable, because a numeric value is out of range.
01524	The result of a column function does not include the null values that were caused by evaluating an arithmetic expression.
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.
01560	A redundant GRANT has been ignored.
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.
01564	The null value has been assigned to a host variable, because division by zero occurred.
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.

Table 3. Class Code 01: Warning (continued)

SQLSTATE Value	Meaning
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).
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.
01609	The procedure generated more than the maximum allowed number of result sets. Only the first integer result sets have been returned to the caller.
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.
01616	The estimated CPU cost exceeds the resource limit.
01618	Redistribute nodegroup is required to change data partitioning.
01620	Some base tables of UNION ALL may be the same table.
01621	The retrieved LOB value may have been changed.

Table 3. Class Code 01: Warning (continued)

SQLSTATE Value	Meaning
01622	Statement completed successfully but a system error occurred after the statement completed.
01623	Value of DEGREE is ignored.
01626	The database has only one active bufferpool.
01627	The DATALINK value may not be valid because the table is in reconcile pending or reconcile is not possible state.
01632	The number of concurrent connections has exceeded the defined entitlement for the product.
01633	The materialized query table may not be used to optimize the processing of queries.
01636	Integrity of non-incremental data remains unverified by the database manager.
01637	Debugging is not enabled.
01639	The federated object may require the invoker to have necessary privileges on data source objects.
01641	Datalink type attributes limit the use of the structured type.
01642	Column not long enough for the largest possible USER default value.
01643	Assignment to an SQLSTATE or SQLCODE variable in an SQL routine may be over-written and does not activate any handler.
01645	The executable for the SQL procedure is not saved in the database catalog.
01648	COMPRESS column attribute ignored because VALUE COMPRESSION is deactivated for the table.
01649	The buffer pool configuration has been completed but will not take effect until the next database restart.
01650	Index and table statistics are inconsistent.
01651	The event monitor was activated successfully, however some monitoring information may be lost.
01652	The isolation clause is ignored because of the statement context.
01653	The authorizations were granted to USER. Groups were not considered because the authorization name is more than 8 bytes.
01654	The buffer pool is not started.
01655	The event monitor was created successfully but at least one event monitor target table already exists.
01657	The bufferpool operation will not take effect until the next database startup.

Table 3. Class Code 01: Warning (continued)

SQLSTATE Value	Meaning
01H51	An MQSeries Application Messaging Interface message was truncated.
01HXX	Valid warning SQLSTATEs returned by a user-defined function or external procedure CALL.

Class Code 02 No Data

Table 4. 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.
02501	The cursor position is not valid for a FETCH of the current row.
02502	Delete or update hole detected

Class Code 07 Dynamic SQL Error

Table 5. Class Code 07: Dynamic SQL Error

SQLSTATE Value	Meaning
07001	The number of host variables is not correct for 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 or the INTO 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.

Class Code 08 Connection Exception

Table 6. 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.
08007	Transaction resolution unknown.
08502	The CONNECT statement issued by an application process running with a SYNCPOINT of TWOPHASE has failed, because no transaction manager is available.
08504	An error was encountered while processing the specified path rename configuration file.

Class Code 09 Triggered Action Exception

Table 7. Class Code 09: Triggered Action Exception

SQLSTATE Value	Meaning
09000	A triggered SQL statement failed.

Class Code 0A Feature Not Supported

Table 8. Class Code 0A: Feature Not Supported

SQLSTATE Value	Meaning
0A001	The CONNECT statement is invalid, because the process is not in the connectable state.
0A502	The action or operation is not enabled for this database instance.
0A503	Federated insert, update, or delete operation cannot be compiled because of potential data inconsistency.

Class Code 0D Invalid Target Type Specification

Table 9. Class Code 0D: Invalid Target Type Specification

SQLSTATE Value	Meaning
0D000	The target structured data type specification is a proper subtype of the source structured data type.

Class Code 0F Invalid Token

Table 10. Class Code 0F: Invalid Token

SQLSTATE Value	Meaning
0F001	The LOB token variable does not currently represent any value.

Class Code 0K Invalid RESIGNAL statement

Table 11. Class Code 0K: Invalid RESIGNAL statement

SQLSTATE Value	Meaning
0K000	RESIGNAL statement is not within a handler.

Class Code 20 Case Not Found for Case Statement

Table 12. Class Code 20: Case Not Found for Case Statement

SQLSTATE Value	Meaning
20000	The case was not found for the CASE statement.

Class Code 21 Cardinality Violation

Table 13. 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.

Table 13. Class Code 21: Cardinality Violation (continued)

SQLSTATE Value	Meaning
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.
21505	A row function must return not more than one row.

Class Code 22 Data Exception

Table 14. 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.
22004	A null value cannot be returned either from a procedure defined as PARAMETER STYLE GENERAL, or from a type-preserving method invoked with a non-null argument.
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.
2200G	The most specific type does not match.
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.

Table 14. Class Code 22: Data Exception (continued)

SQLSTATE Value	Meaning
22025	The LIKE predicate string pattern contains an invalid occurrence of an escape character.
2202D	Null instance used with a mutator method.
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.
22526	A key transform function generated no rows or duplicate rows.
22527	Invalid input data detected for a multiple row INSERT operation.

Class Code 23 Constraint Violation

Table 15. 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.
23510	A violation of a constraint on the use of the command imposed by the RLST table 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.

Table 15. Class Code 23: Constraint Violation (continued)

SQLSTATE Value	Meaning
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.
23522	The range of values for an identity column or sequence is exhausted.

Class Code 24 Invalid Cursor State

Table 16. 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.
24510	An UPDATE or DELETE operation was attempted against a delete hole or update hole.
24512	The result table does not agree with base table.
24513	FETCH NEXT, PRIOR, CURRENT or RELATIVE is not allowed because the cursor position is not known.
24514	A previous error has disabled this cursor.
24516	A cursor has already been assigned to a result set.
24517	A cursor was left open by an external function or method.

Class Code 25 Invalid Transaction State

Table 17. Class Code 25: Invalid Transaction State

SQLSTATE Value	Meaning
25000	An update operation is invalid for the application execution environment.

Table 17. Class Code 25: Invalid Transaction State (continued)

SQLSTATE Value	Meaning
25001	The statement is only allowed as the first statement in a unit of work.
25501	The statement is only allowed as the first statement in a unit of work.

Class Code 26 Invalid SQL Statement Identifier

Table 18. Class Code 26: Invalid SQL Statement Identifier

SQLSTATE Value	Meaning
26501	The statement identified does not exist.

Class Code 28 Invalid Authorization Specification

Table 19. Class Code 28: Invalid Authorization Specification

SQLSTATE Value	Meaning
28000	Authorization name is invalid.

Class Code 2D Invalid Transaction Termination

Table 20. Class Code 2D: Invalid Transaction Termination

SQLSTATE Value	Meaning
2D521	SQL COMMIT or ROLLBACK are invalid in the current operating environment.
2D522	COMMIT and ROLLBACK are not allowed in an ATOMIC Compound statement.
2D528	Dynamic COMMIT is invalid for the application execution environment.
2D529	Dynamic ROLLBACK is invalid for the application execution environment.

Class Code 2E Invalid Connection Name

Table 21. Class Code 2E: Invalid Connection Name

SQLSTATE Value	Meaning
2E000	Connection name is invalid.

Class Code 34 Invalid Cursor Name

Table 22. Class Code 34: Invalid Cursor Name

SQLSTATE Value	Meaning
34000	Cursor name is invalid.

Class Code 36 Invalid Cursor Specification

Table 23. Class Code 36: Invalid Cursor Specification

SQLSTATE Value	Meaning
36001	A SENSITIVE cursor cannot be defined for the specified select-statement.

Class Code 38 External Function Exception

Table 24. Class Code 38: External Function Exception

SQLSTATE Value	Meaning
38XXX	Valid error SQLSTATEs returned by an external routine, or trigger.
38001	The external routine is not allowed to execute SQL statements.
38002	The routine attempted to modify data, but the routine was not defined as MODIFIES SQL DATA.
38003	The statement is not allowed in a routine.
38004	The routine attempted to read data, but the routine was not defined as READS SQL DATA.
38501	Error occurred while calling a user-defined function, external procedure, or trigger (using the SIMPLE CALL or SIMPLE CALL WITH NULLS calling convention).
38502	The external function is not allowed to execute SQL statements.
38503	A user-defined function has abnormally terminated (abend).

Table 24. Class Code 38: External Function Exception (continued)

SQLSTATE Value	Meaning
38504	A user-defined function has been interrupted by the user to stop a probable looping condition.
38505	An SQL statement is not allowed in a routine on a FINAL CALL.
38506	Function failed with error from OLE DB provider.
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
38553	A routine in a system schema has terminated with an error.
38H01	An MQSeries function failed to initialize.
38H02	MQSeries Application Messaging Interface failed to terminate the session.
38H03	MQSeries Application Messaging Interface failed to properly process a message.
38H04	MQSeries Application Messaging Interface failed in sending a message.
38H05	MQSeries Application Messaging Interface failed to read/receive a message.
38H06	An MQSeries Application Messaging Interface (un)subscription request failed.

Class Code 39 External Function Call Exception

Table 25. Class Code 39: External Function Call Exception

SQLSTATE Value	Meaning
39001	A user-defined function has returned an invalid SQLSTATE.
39004	A null value is not allowed for an IN or INOUT argument.

Table 25. Class Code 39: External Function Call Exception (continued)

SQLSTATE Value	Meaning
39501	The eye-catcher associated with an argument was modified.

Class Code 3B Invalid SAVEPOINT

Table 26. Class Code 3B: Invalid SAVEPOINT

SQLSTATE Value	Meaning
3B001	The savepoint is not valid.
3B002	The maximum number of savepoints has been reached.
3B501	A duplicate savepoint name was detected.
3B502	A RELEASE or ROLLBACK TO SAVEPOINT was specified, but a savepoint does not exist.
3B503	A SAVEPOINT, RELEASE SAVEPOINT, or ROLLBACK TO SAVEPOINT is not allowed in a trigger or global transaction.

Class Code 40 Transaction Rollback

Table 27. 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.
40506	The current transaction was rolled back because of an SQL error.
40507	The current transaction was rolled back as a result of a failure creating an index.

Class Code 42 Syntax Error or Access Rule Violation

Table 28. 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.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
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.
42509	SQL statement is not authorized, because of the STATICRULES option.
42511	Unable to retrieve a DATALINK value.
42601	A character, token, or clause is invalid or missing.
42602	A character that is invalid in a name has been detected.
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.
42620	Read-only SCROLL was specified with the UPDATE clause.
42621	The check constraint is invalid.
42622	A name or label is too long.
42623	A DEFAULT clause cannot be specified.
42625	A CASE expression is invalid.
42627	RETURNS clause must be specified prior to predicate specification using the EXPRESSION AS clause.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
42628	A TO SQL or FROM SQL transform function is defined more than once in a transform definition.
42629	Parameter names must be specified for SQL routines.
42630	An SQLSTATE or SQLCODE variable declaration must not be in a nested compound statement.
42631	A RETURN statement in an SQL function or method must include a return value.
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, attribute, or parameter 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	Duplicate column name or attribute name was detected in the object definition or ALTER statement.
42712	A duplicate table designator was detected in the FROM clause.
42713	A duplicate object was detected in a list of objects.
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.
42725	A function or method was referenced directly (not by either signature or by specific instance name), but there is more than one specific instance of that function or method.
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.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
42731	The container name is already used by this tablespace.
42732	A duplicate schema name in the SET CURRENT PATH statement was detected.
42734	A duplicate parameter-name, SQL variable-name, cursor-name, condition-name, or label was detected.
42735	Nodegroup for the table space is not defined for the buffer pool.
42736	The label specified on the LEAVE statement is not found or not valid.
42737	The condition specified in the handler is not defined.
42738	A duplicate column name or unnamed column was specified in a DECLARE CURSOR statement of a FOR statement.
42739	A duplicate transform was detected.
42740	No transforms were found for the specified type. No transforms were dropped.
42741	A transform group is not defined for a data type.
42742	Subtable or subview of the same type already exists in the typed table or typed view hierarchy.
42743	The search method is not found in the index extension.
42744	A TO SQL or FROM SQL transform function is not defined in a transform group.
42745	The routine would define an overriding relationship with an existing method.
42746	A method name cannot be the same as a structured type name within the same type hierarchy.
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.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
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	Multiple columns are returned from a subquery that is allowed only one column.
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.
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.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
42837	The column cannot be altered, because its attributes are not compatible with the current column attributes.
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 or reference 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.)
42853	Both alternatives of an option were specified, or the same option was specified more than once.
42854	A result column data type in the select list is not compatible with the defined type in a typed view or materialized query table definition.
42855	The assignment of the LOB to this host variable is not allowed. The target host variable for all fetches of this LOB value for this cursor must be a locator or a LOB variable.
42858	Operation cannot be applied to the specified object.
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 from the sourced function or RETURN statement in the function body.
42872	FETCH statement clauses are incompatible with the cursor definition.
42875	The object to create within CREATE SCHEMA must have the same qualifier as the schema name.
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.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
42880	The CAST TO and CAST FROM data types are incompatible, or would always result in truncation of a fixed string.
42881	Invalid use of a row based function.
42882	The specific instance name qualifier is not equal to the function name qualifier.
42883	No function or method was found with a matching signature.
42884	No routine 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.
42886	The IN, OUT, or INOUT parameter attributes do not match.
42887	The function is not valid in the context where it occurs.
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.
42891	A duplicate UNIQUE constraint already exists.
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 INTEGRITY statement either does not have the proper structure, or it has been defined with generated columns, constraints, or triggers.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
428A6	An exception table named in a SET INTEGRITY 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 INTEGRITY statement.
428A8	Cannot reset the check-pending state using the SET INTEGRITY statement on a descendent table while a parent table is in the check-pending state.
428A9	The node range is invalid.
428AA	The column name is not a valid column for an event monitor table.
428B0	Illegal nesting inside ROLLUP, CUBE, or GROUPING SETS.
428B1	Incorrect number of table space container specifications that are not designated for specific nodes.
428B2	The path name for the container is not valid.
428B3	An invalid SQLSTATE was specified on RAISE_ERROR.
428B7	A number specified in an SQL statement is out of the valid range.
429BE	The primary key or unique key is a subset of the columns in the dimensions clause.
428C0	The node cannot be dropped, because it is the only node in the nodegroup.
428C1	Only one ROWID column can be specified for a table.
428C2	Examination of the function body indicates that the given clause should have been specified on the CREATE FUNCTION statement.
428C4	The number of elements on each side of the predicate operator is not the same.
428C5	No data type mapping was found for a data type from the data source.
428C9	A ROWID column cannot be specified as the target column of an INSERT or UPDATE.
428CA	A table in append mode cannot have a clustered index.
428CB	The page size for a table space must match the page size of the associated bufferpool.
428D1	Unable to access a file referenced by a DATALINK value.
428D4	A cursor specified in a FOR statement cannot be referenced in an OPEN, CLOSE, or FETCH statement.
428D5	The ending label does not match the beginning label.
428D6	UNDO is not allowed for NOT ATOMIC statements.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
428D7	The condition value is not allowed.
428D8	The declaration or use of the SQLSTATE or SQLCODE variable is not valid.
428DB	An object is not valid as a supertype, supertable, or superview.
428DC	The function or method is not valid as the transform for this type.
428DD	A required transform is not defined.
428DE	PAGESIZE value is not supported.
428DF	Data types specified in CREATE CAST are not valid.
428DG	Function specified in CREATE CAST is not valid.
428DH	Operation is not valid for typed tables.
428DJ	Inherited column or attribute cannot be changed or dropped.
428DK	The scope for the reference column is already defined.
428DL	Parameter of external or sourced function has a scope defined.
428DM	The scope table or view is not valid for the reference type.
428DN	SCOPE is not specified in the RETURNS clause of an external function or is specified in the RETURNS clause of a sourced function.
428DP	The type is not a structured type.
428DQ	A subtable or subview cannot have a different schema name than its supertable or superview.
428DR	Operation cannot be applied to a sub-table.
428DS	Index on the specified columns cannot be defined on subtable.
428DT	Operand of expression is not a valid scoped reference type.
428DU	A type is not included in the required type hierarchy.
428DV	Invalid left operand of a dereference operator.
428DW	Object identifier column cannot be referenced using the dereference operator.
428DX	Object identifier column is required to define the root table or root view of a typed table or typed view hierarchy.
428DY	Table statistics cannot be updated for a subtable.
428DZ	An object identifier column cannot be updated.
428E0	Definition of index does not match the definition of the index extension.
428E1	Result of the range-producing table function is inconsistent with that of the key transformation table function for the index extension.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
428E2	Number or the type of key-target parameters does not match with the number or type of key transform function for the index extension.
428E3	Argument for function in index extension is not valid.
428E4	Function is not supported in CREATE INDEX EXTENSION statement.
428E5	SELECTIVITY clause can only be specified with a user-defined predicate.
428E6	The search argument of method in the user-defined predicate does not match the one in the corresponding search method of the index extension.
428E7	Type of the operand following the comparison operator in the user-defined predicate does not match the RETURNS data type.
428E8	A search target or search argument parameter does not match a parameter name of the function being created.
428E9	An argument parameter name cannot appear as both a search target and search argument in the same exploitation rule.
428EA	A fullselect in a typed view is not valid.
428EB	A column in a subview cannot be read only when the corresponding column in the superview is updatable.
428EC	The fullselect specified for the materialized query table is not valid.
428ED	Structured types with Datalink or Reference type attributes cannot be constructed.
428EE	Option is not valid for this data source.
428EF	Value for the option is not valid for this data source.
428EG	Missing required option for this data source.
428EH	Cannot ADD an option that is already defined.
428EJ	Cannot SET or DROP an option that has not been ADDed.
428EK	The qualifier for a declared global temporary table name must be SESSION.
428EL	A transform function is not valid for use with a function or method.
428EM	The TRANSFORM GROUP clause is required.
428EN	A transform group is specified that is not used.
428EP	A structured type cannot depend on itself either directly or indirectly.
428EQ	The returns type of the routine is not the same as the subject type.
428ER	A method specification cannot be dropped before the method body is dropped.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
428ES	A method body does not correspond to the language type of the method specification.
428EU	TYPE or VERSION is not specified in the server definition.
428EV	Pass-through facility is not supported for the type of data source.
428EW	The table cannot be converted to or from a materialized query table.
428EX	Routine cannot be used as a transform function because it is either a builtin function or a method.
428EY	The data type of the search target in a user-defined predicate does not match the data type of the source key of the specified index extension.
428EZ	A window specification for an OLAP function is not valid.
428F0	A ROW function must include at least two columns.
428F1	An SQL TABLE function must return a table result.
428F2	The data type of the RETURN statement value in an SQL procedure must be INTEGER.
428F3	SCROLL and WITH RETURN are mutually exclusive.
428F4	The SENSITIVITY specified on FETCH is not allowed for the cursor.
428F6	Cursor is scrollable, but the result table involves output from a table function.
428F7	An operation that applies only to SQL routines was attempted on an external routine.
428F9	A sequence expression cannot be specified in this context.
428FA	The scale of the decimal number must be zero.
428FB	The sequence-name must not be a sequence generated by the system for an identity column.
428FC	The length of the encryption password is not valid.
428FD	The password used for decryption does not match the password used to encrypt the data.
428FE	The data is not a result of the ENCRYPT function.
428FF	The buffer pool specification is not valid.
428FG	The table used to define a staging table is not valid.
428FH	The SET INTEGRITY option is not valid
428FI	ORDER OF was specified, but the table-designator does not contain an ORDER BY clause.
428FJ	ORDER BY is not allowed in the outer fullselect of a view or materialized query table.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
428FP	One INSTEAD OF trigger is allowed for each kind of operation on a subject view.
428FQ	The view name of an INSTEAD OF trigger must not specify a view that is defined using WITH CHECK OPTION (a symmetric view) or a view on which a symmetric view has been defined directly or indirectly.
428FU	Built-in type returned from the FROM SQL transform function or method does not match the corresponding built-in type for the TO SQL transform function or method.
428FV	Cannot define the method as an overriding method.
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.
42904	The SQL procedure was not created because of a compile error.
42907	The string is too long.
42908	The statement does not include a required column list.
42910	The statement is not allowed in a Compond statement.
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 or altered.
42918	A user-defined data type cannot be created with a system-defined data type name (for example, INTEGER).
42919	Nested compound statements are not allowed.
42921	Containers cannot be added to the tablespace.
42925	Recursive named derived tables cannot specify SELECT DISTINCT and must specify UNION ALL.
42928	WITH EMPTY TABLE cannot be specified for the table.
42932	The program preparation assumptions are incorrect.
42939	The name cannot be used because the specified identifier is reserved for system use.

Table 28. Class Code 42: Syntax Error or Access Rule Violation (continued)

SQLSTATE Value	Meaning
42962	A long column, LOB column, or structured type column cannot be used in an index, a key, or a constraint.
42968	The connection failed, because there is no current software license.
42969	The package was not created and the current unit of work was rolled back, because of internal limitations or an invalid section number.
42972	An expression in a join-condition references columns in more than one of the operand tables.
42985	The statement is not allowed in a routine.
42987	The statement is not allowed in a procedure or trigger.
42989	A GENERATED column that is based on an expression cannot be used in a BEFORE 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.
42995	The requested function does not apply to global temporary tables.
42997	Capability is not supported by this version of the DB2 application requester, DB2 application server, or the combination of the two.
429A0	A foreign key cannot reference a parent table defined as not logged initially.
429A1	Nodegroup is not valid for the table space.
429A9	SQL statement cannot be processed by DataJoiner.
429B2	Specified inline length value for the structured type or column is too small.
429B3	Object may not be defined on a subtable.
429B4	Data filter function cannot be a LANGUAGE SQL function.
429B5	Data type of instance parameter in index extension is not valid.
429B8	A routine defined with PARAMETER STYLE JAVA cannot have a structured type as a parameter or returns type.
429B9	DEFAULT or NULL cannot be used in an attribute assignment.
429BA	The FEDERATED keyword must be used with a reference to a federated database object.
429BB	A data type specified for a parameter or variable is not supported in a SQL routine.
429BC	There are multiple container actions in the ALTER TABLESPACE statement.

Class Code 44 WITH CHECK OPTION Violation

Table 29. 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.

Class Code 46 Java DDL

Table 30. Class Code 46: Java DDL

SQLSTATE Value	Meaning
46001	Java DDL - Invalid URL.
46002	Java DDL - Invalid jar name.
46003	Java DDL - Invalid class deletion.
46007	Java DDL - Invalid signature.
46008	Java DDL - Invalid method specification.
46103	A Java routine encountered a ClassNotFound exception.
46501	Java DDL - Optional component not implemented.

Class Code 51 Invalid Application State

Table 31. Class Code 51: Invalid Application State

SQLSTATE Value	Meaning
51002	The package corresponding to an SQL statement execution request was not found.
51003	Consistency tokens do not match.
51004	An address in the SQLDA is invalid.
51005	Thr previous system error has disabled this function.
51008	The release number of the precompiled program is not valid.
51015	An attempt was made to execute a section that was found to be in error at bind time.
51017	The user is not logged on.
51021	SQL statements cannot be executed until the application process executes a rollback operation.

Table 31. Class Code 51: Invalid Application State (continued)

SQLSTATE Value	Meaning
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 INTEGRITY statement is not valid since a table is a user maintained materialized query table or is not in the check-pending state.
51028	A package cannot be used, because it is marked inoperative.
51030	The procedure referenced in a ALLOCATE CURSOR, or an ASSOCIATE LOCATORS statement has not yet been called within the application process.
51034	A routine defined with MODIFIES SQL DATA is not valid in the context in which it is invoked.
51035	A PREVVAL expression cannot be used because a value has not been generated for the sequence yet in this session.
51038	SQL statements may no longer be issued by the routine.
51039	The ENCRYPTION PASSWORD value is not set.

Class Code 53 Invalid Operand or Inconsistent Specification

Table 32. Class Code 53: Invalid Operand or Inconsistent Specification

SQLSTATE Value	Meaning
53040	The specified buffer pool does not exist on the specified database partition.

Class Code 54 SQL or Product Limit Exceeded

Table 33. 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, has too many columns, or a key column is too long.
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.
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	An internal object 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 nested routines or triggers was exceed.
54045	Maximum levels of a type hierarchy exceeded.
54046	The maximum allowable parameters is exceeded in an index extension.
54047	The maximum size of a table space is exceeded.
54048	A temporary table space with sufficient page size does not exist.
54049	Length of an instance of a structured type exceeds system limit.
54050	The maximum allowable attributes is exceeded in a structured type.
54052	The number of block pages for a buffer pool is too large for the size of the buffer pool.
54053	The value specified for BLOCKSIZE is not in the valid range.

Class Code 55 Object Not in Prerequisite State

Table 34. 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.
55012	A clustering index already exists on the table.
55019	The table is in an invalid state for the operation.
55022	The file server is not registered with this database.
55023	An error occured calling a routine.
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	The event monitor is in an invalid state for the operation.
55035	The table cannot be dropped, because it is protected.
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 34. Class Code 55: Object Not in Prerequisite State (continued)

SQLSTATE Value	Meaning
55043	Attributes of a structured type cannot be altered when a typed table or typed view based on the type exists.
55045	The SQL Archive (SAR) file for the routine cannot be created because a necessary component is not available at the server.
55046	The specified SQL archive does not match the target environment.
55047	A routine declared as NOT FEDERATED attempted to access a federated object.
55048	Encrypted data cannot be encrypted.
55049	The event monitor table is not properly defined.
55051	The ALTER BUFFERPOOL statement is currently in progress.
55054	Cannot define the method as an overriding method.

Class Code 56 Miscellaneous SQL or Product Error

Table 35. 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.
56033	The insert or update value of a long string column must be a host variable or NULL.
56038	The requested feature is not supported in this environment.
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.
56095	A bind option is invalid.
56097	LONG VARCHAR and LONG VARGRAPHIC fields are not permitted in TABLESPACEs which are built on DEVICEs.
56098	An error occurred during implicit rebind or prepare.
56099	The REAL data type is not supported by the target database.
560A0	Action on a LOB value failed.
560AA	The clause or scalar function is invalid, because UCS-2 is not supported on this system.

Table 35. Class Code 56: Miscellaneous SQL or Product Error (continued)

SQLSTATE Value	Meaning
560AC	Wrapper definition cannot be used for the specified type or version of data source.
560AF	PREPARE statement is not supported when using gateway concentrator.
560B0	Invalid new size value for table space container resizing.
560B1	Invalid cursor specification in stored procedure.
560B7	For a multiple row INSERT, the usage of a sequence expression must be the same for each row.
560BB	For an INOUT parameter in a dynamically prepared CALL statement, the same host variable must be used in both the USING and INTO clauses.
560BC	An error has occurred when accessing a file.
560BD	A federated server received an unexpected error code from a data source.
560BF	Encryption facility not available.

Class Code 57 Resource Not Available or Operator Intervention

Table 36. Class Code 57: Resource Not Available or Operator Intervention

SQLSTATE Value	Meaning
57001	The table is unavailable, because it does not have a primary index.
57003	The specified bufferpool has not been activated.
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.

Table 36. Class Code 57: Resource Not Available or Operator Intervention (continued)

SQLSTATE Value	Meaning
57022	The table could not be created, because the authorization ID of the statement does not own any suitable dbspaces.
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.
57050	The file server is not currently available.
57051	The estimated CPU cost exceeds the resource limit.
57052	Node is unavailable, because it does not have containers for all temporary table spaces.
57053	The operation cannot be performed on the table because of conflicting operations.
57055	A temporary table space with sufficient page size was not available.
57056	Package is not available because the database is in NO PACKAGE LOCK mode.
57057	The SQL statement cannot be executed due to a prior condition in a DRDA chain of SQL statements.
57059	There is not enough space in the table space for the specified action.

Class Code 58 System Error

Table 37. 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.

Table 37. Class Code 58: System Error (continued)

SQLSTATE Value	Meaning
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.
58023	A system error has caused the current program to be canceled.
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.

Appendix A. Communications Errors

If your application returns an 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.

The error codes are grouped by protocol as follows:
- "TCP/IP"
- "APPC" on page 595
- "NETBIOS" on page 596

TCP/IP

When using TCP/IP in a Unix environment, the following table lists some of the *errnos* which the user may encounter most frequently. It is NOT a complete list of errors. *Errnos* can be found in the file **/usr/include/sys/errno.h**. On Linux, the *errnos* are located in **/usr/include/asm/errno.h**. The *errno* number itself is indicated for each operating system.

Table 38. Unix TCP/IP errnos.

Errno	Errno Number				Description
	AIX	HP-UX	Solaris	Linux	
EINTR	4	4	4	4	Interrupted system call.
EAGAIN	11	11	11	11	Resource temporarily unavailable.
EBUSY	16	16	16	16	Resource busy.
EMFILE	24	24	24	24	The per-process file descriptor table is full.
EPIPE	32	32	32	32	Broken pipe.
EADDRINUSE	67	226	125	98	The specified address is already in use.
ENETDOWN	69	228	127	100	The network is down.
ENETUNREACH	70	229	128	101	No route to the network is available.
ENETRESET	71	230	129	102	The network has dropped the connection on reset.

Table 38. Unix TCP/IP errnos. (continued)

Errno	Errno Number				Description
	AIX	HP-UX	Solaris	Linux	
ECONNRESET	73	232	131	104	Connection has been reset by the partner.
ENOBUFS	74	233	132	105	Insufficient buffer space resources were available in the system to complete the call.
ENOTCONN	76	235	134	107	The socket is not connected.
ETIMEDOUT	78	238	145	110	The connection timed out.
ECONNREFUSED	79	239	146	111	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 have 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	241	147	112	The host is down.
EHOSTUNREACH	81	242	148	113	No route to the host is available.

For more information on Unix TCP/IP communication errors, refer to the technical reference manuals for the appropriate operating system. You can also issue the following command:

```
man function-name
```

where *function-name* represents the name of the function that returned an error. The man pages might provide you with additional information about errors returned by specific functions.

When using TCP/IP on a Windows operating system, the following is a list of error codes which the user may encounter most frequently. It is NOT a complete list of errors. Errors returned by WSAGetLastError() can be found in the file **winsock2.h**. This file may not be installed on your system if you do not have a development environment installed. Detailed information about

errors returned by specific functions are documented in the Windows Sockets 2 Application Programming Interface. You can get a copy of this specification from the following website: http://www.sockets.com/winsock2.htm#Docs.

- WSAEINVAL (10022): Invalid argument. It can also mean that the current state of the socket is not compatible with the TCP function being called, or the socket may no longer be valid.

- WSAEMFILE (10024): Too many open files. Too many open sockets. Each implementation may have a maximum number of socket handles available, either globally, per process, or per thread.

- WSAEWOULDBLOCK (10035): Resource temporarily unavailable. This error is returned from operations on non-blocking sockets that cannot be completed immediately.

- WSAEINPROGRESS (10036): A blocking Windows Sockets operation is in progress. Windows Sockets only allows a single blocking operation to be outstanding per task (or thread), and if any other function call is made, the function fails with WSAEINPROGRESS.

- WSAENOPROTOOPT (10042): Bad protocol option. An unknown, invalid, or unsupported option or level was specified in a getsockopt() or setsockopt() call.

- WSAEADDRINUSE (10048): Address already in use. Only one usage of each socket address (protocol/IP address/port) is normally permitted. This error occurs if an application attempts to bind() a socket to an IP address/port that has already been used for an existing socket, or a socket that wasn't closed properly, or one that is still in the process of closing.

- WSAENETDOWN (10050): Network is down. A socket operation encountered a dead network. This could indicate a serious failure of the network system (i.e. the protocol stack that the WinSock DLL runs over), the network interface, or the local network itself.

- WSAENETUNREACH (10051): Network is unreachable. A socket operation was attempted to an unreachable network. This usually means the local software knows no route to reach the remote host.

- WSAENETRESET (10052): Network dropped connection on reset. The connection has been broken due to "keep-alive" activity detecting a failure while the operation was in progress. May also be returned by stsockopt() if an attempt is made to set SO_KEEPALIVE on a connection that has already failed.

- WSAECONNABORTED (10053): Software caused connection abort. An established connection was aborted by the software on your machine, possibly due to a data transmission timeout or protocol error.

- WSAECONNRESET (10054): Connection reset by peer. An existing connection was forcibly closed by the remote peer. This normally results if the peer application on the remote machine is suddenly stopped, the machine is rebooted, or the peer application used a "hard close" on the

remote socket. This error may also result if a connection was broken due to "keep-alive" activity detecting a failure while one or more operations are in progress. Operations in progress fail with WSAENETRESET. Subsequent operations fail with WSAECONNRESET.

- WSAENOBUFS (10055): No buffer space available. An operation on a socket could not be performed because the system lacked sufficient buffer space or because a queue was full.
- WSAENOTCONN (10057): The socket is not connected.
- WSAETIMEDOUT (10060): Connection timed out. A connection attempt failed because the connected party did not properly respond after a period of time.
- WSAECONNREFUSED (10061): The connection has been refused. If you are trying to attach to the instance or connect to the database, check that the database manager and TCP/IP protocol support at the server has been started successfully.
- WSAEHOSTUNREACH (10065): No route to host. A socket operation was attempted to an unreachable machine.
- WSASYSNOTREADY (10091): The underlying network subsystem is not ready for network communication. User should check that the appropriate Windows Sockets DLL is in the current path and that they are not trying to use more than one WinSock implementation simultaneously. If there is more than one WinSock DLL on you system, be sure the first one in the path is appropriate for the network subsystem currently loaded and that all necessary components are currently installed and configured correctly.
- WSAVERNOTSUPPORTED (10092): The version of Windows Sockets API support requested is not provided by this particular Windows Sockets implementation. Check that no old Windows Sockets DLL files are being accessed.
- WSA_E_NO_MORE (10110), WSAENOMORE (10102): There is no more data available. In Windows Sockets Version 2, conflicting error codes are defined for WSAENOMORE and WSA_E_NO_MORE. The error code WSAENOMORE will be removed in a future version and only WSA_E_NO_MORE will remain.
- WSAHOST_NOT_FOUND (11001): Host not found.
- WSATRY_AGAIN (11002): Host not found. Request to retrieve the hostname's IP address from the name server failed because the local machine did not receive a response from the name server.
- WSANO_DATA (11004): Valid name, no data record of requested type. The name server or hosts file does not recognize the hostname, or the service name is not specified in the services file.

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 which the user may 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. 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. Ensure that the database manager and APPC protocol support at the server have 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.
- 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 Communication Server for AIX, 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.

- CM_RESOURCE_FAILURE_NO_RETRY (26): This error occurs when the conversation terminates prematurely (on either the remote or local end) from an error related to resources (such as sessions or links).
- 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 Communication Server for AIX, 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 Communication Server for AIX 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.

On HP-UX, for SNAPlus2, please refer to the /usr/include/sys/errno.h file for a description of the error.

For more information on CPI Communications errors, refer to *Systems Application Architecture Common Programming Interface Communications Reference*.

NETBIOS

When using NetBIOS, the following is a list of hexadecimal return codes that the user may encounter most frequently. 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 used on the network. Ensure that the nname parameter defined in the database manager configuration file is unique on the network.
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 another environment

34	Environment not defined, RESET must be issued
35	Required operating system resources exhausted, retry later
36	Max applications exceeded
37	No SAPs available
38	Requested resources not available
39	Invalid NCB address
3A	Invalid reset
3B	Invalid NCB DD ID
3C	Segment lock unsuccessful
3F	Device driver open error.
40	OS error detected
4F	Perm ring status
F6	Unexpected CCB error
F8	Adapter open error
F9	Adapter support software internal error
FA	Adapter check
FB	NetBIOS not operational
FC	Open failures
FD	Unexpected adapter close
FF	Command in process

For more information on NetBIOS return codes, refer to the Local Area Network technical reference.

Appendix B. DB2 Universal Database technical information

Overview of DB2 Universal Database technical information

DB2 Universal Database technical information can be obtained in the following formats:

- Books (PDF and hard-copy formats)
- A topic tree (HTML format)
- Help for DB2 tools (HTML format)
- Sample programs (HTML format)
- Command line help
- Tutorials

This section is an overview of the technical information that is provided and how you can access it.

FixPaks for DB2 documentation

IBM may periodically make documentation FixPaks available. Documentation FixPaks allow you to update the information that you installed from the *DB2 HTML Documentation CD* as new information becomes available.

Note: If you do install documentation FixPaks, your HTML documentation will contain more recent information than either the DB2 printed or online PDF manuals.

Categories of DB2 technical information

The DB2 technical information is categorized by the following headings:

- Core DB2 information
- Administration information
- Application development information
- Business intelligence information
- DB2 Connect information
- Getting started information
- Tutorial information
- Optional component information
- Release notes

The following tables describe, for each book in the DB2 library, the information needed to order the hard copy, print or view the PDF, or locate the HTML directory for that book. A full description of each of the books in

the DB2 library is available from the IBM Publications Center at
www.ibm.com/shop/publications/order

The installation directory for the HTML documentation CD differs for each
category of information:

`htmlcdpath/doc/htmlcd/%L/category`

where:
- *htmlcdpath* is the directory where the HTML CD is installed.
- *%L* is the language identifier. For example, en_US.
- *category* is the category identifier. For example, core for the core DB2
  information.

In the PDF file name column in the following tables, the character in the sixth
position of the file name indicates the language version of a book. For
example, the file name db2d1e80 identifies the English version of the
Administration Guide: Planning and the file name db2d1g80 identifies the
German version of the same book. The following letters are used in the sixth
position of the file name to indicate the language version:

Language	Identifier
Arabic	w
Brazilian Portuguese	b
Bulgarian	u
Croatian	9
Czech	x
Danish	d
Dutch	q
English	e
Finnish	y
French	f
German	g
Greek	a
Hungarian	h
Italian	i
Japanese	j
Korean	k
Norwegian	n
Polish	p
Portuguese	v
Romanian	8
Russian	r
Simp. Chinese	c
Slovakian	7
Slovenian	l

Spanish	z
Swedish	s
Trad. Chinese	t
Turkish	m

No form number indicates that the book is only available online and does not have a printed version.

Core DB2 information

The information in this category covers DB2 topics that are fundamental to all DB2 users. You will find the information in this category useful whether you are a programmer, a database administrator, or you work with DB2 Connect, DB2 Warehouse Manager, or other DB2 products.

The installation directory for this category is doc/htmlcd/%L/core.

Table 39. Core DB2 information

Name	Form Number	PDF File Name
IBM DB2 Universal Database Command Reference	SC09-4828	db2n0x80
IBM DB2 Universal Database Glossary	No form number	db2t0x80
IBM DB2 Universal Database Master Index	SC09-4839	db2w0x80
IBM DB2 Universal Database Message Reference, Volume 1	GC09-4840	db2m1x80
IBM DB2 Universal Database Message Reference, Volume 2	GC09-4841	db2m2x80
IBM DB2 Universal Database What's New	SC09-4848	db2q0x80

Administration information

The information in this category covers those topics required to effectively design, implement, and maintain DB2 databases, data warehouses, and federated systems.

The installation directory for this category is doc/htmlcd/*%L*/admin.

Table 40. Administration information

Name	Form number	PDF file name
IBM DB2 Universal Database Administration Guide: Planning	SC09-4822	db2d1x80
IBM DB2 Universal Database Administration Guide: Implementation	SC09-4820	db2d2x80
IBM DB2 Universal Database Administration Guide: Performance	SC09-4821	db2d3x80
IBM DB2 Universal Database Administrative API Reference	SC09-4824	db2b0x80
IBM DB2 Universal Database Data Movement Utilities Guide and Reference	SC09-4830	db2dmx80
IBM DB2 Universal Database Data Recovery and High Availability Guide and Reference	SC09-4831	db2hax80
IBM DB2 Universal Database Data Warehouse Center Administration Guide	SC27-1123	db2ddx80
IBM DB2 Universal Database Federated Systems Guide	GC27-1224	db2fpx80
IBM DB2 Universal Database Guide to GUI Tools for Administration and Development	SC09-4851	db2atx80
IBM DB2 Universal Database Replication Guide and Reference	SC27-1121	db2e0x80
IBM DB2 Installing and Administering a Satellite Environment	GC09-4823	db2dsx80
IBM DB2 Universal Database SQL Reference, Volume 1	SC09-4844	db2s1x80
IBM DB2 Universal Database SQL Reference, Volume 2	SC09-4845	db2s2x80
IBM DB2 Universal Database System Monitor Guide and Reference	SC09-4847	db2f0x80

Application development information

The information in this category is of special interest to application developers or programmers working with DB2. You will find information about supported languages and compilers, as well as the documentation required to access DB2 using the various supported programming interfaces, such as embedded SQL, ODBC, JDBC, SQLj, and CLI. If you view this information online in HTML you can also access a set of DB2 sample programs in HTML.

The installation directory for this category is doc/htmlcd/%L/ad.

Table 41. Application development information

Name	Form number	PDF file name
IBM DB2 Universal Database Application Development Guide: Building and Running Applications	SC09-4825	db2axx80
IBM DB2 Universal Database Application Development Guide: Programming Client Applications	SC09-4826	db2a1x80
IBM DB2 Universal Database Application Development Guide: Programming Server Applications	SC09-4827	db2a2x80
IBM DB2 Universal Database Call Level Interface Guide and Reference, Volume 1	SC09-4849	db2l1x80
IBM DB2 Universal Database Call Level Interface Guide and Reference, Volume 2	SC09-4850	db2l2x80
IBM DB2 Universal Database Data Warehouse Center Application Integration Guide	SC27-1124	db2adx80
IBM DB2 XML Extender Administration and Programming	SC27-1234	db2sxx80

Business intelligence information

The information in this category describes how to use components that enhance the data warehousing and analytical capabilities of DB2 Universal Database.

The installation directory for this category is doc/htmlcd/%L/wareh.

Table 42. Business intelligence information

Name	Form number	PDF file name
IBM DB2 Warehouse Manager Information Catalog Center Administration Guide	SC27-1125	db2dix80
IBM DB2 Warehouse Manager Installation Guide	GC27-1122	db2idx80

DB2 Connect information
The information in this category describes how to access host or iSeries data using DB2 Connect Enterprise Edition or DB2 Connect Personal Edition.

The installation directory for this category is doc/htmlcd/%L/conn.

Table 43. DB2 Connect information

Name	Form number	PDF file name
APPC, CPI-C, and SNA Sense Codes	No form number	db2apx80
IBM Connectivity Supplement	No form number	db2h1x80
IBM DB2 Connect Quick Beginnings for DB2 Connect Enterprise Edition	GC09-4833	db2c6x80
IBM DB2 Connect Quick Beginnings for DB2 Connect Personal Edition	GC09-4834	db2c1x80
IBM DB2 Connect User's Guide	SC09-4835	db2c0x80

Getting started information
The information in this category is useful when you are installing and configuring servers, clients, and other DB2 products.

The installation directory for this category is doc/htmlcd/%L/start.

Table 44. Getting started information

Name	Form number	PDF file name
IBM DB2 Universal Database Quick Beginnings for DB2 Clients	GC09-4832	db2itx80

Table 44. Getting started information (continued)

Name	Form number	PDF file name
IBM DB2 Universal Database Quick Beginnings for DB2 Servers	GC09-4836	db2isx80
IBM DB2 Universal Database Quick Beginnings for DB2 Personal Edition	GC09-4838	db2i1x80
IBM DB2 Universal Database Installation and Configuration Supplement	GC09-4837	db2iyx80
IBM DB2 Universal Database Quick Beginnings for DB2 Data Links Manager	GC09-4829	db2z6x80

Tutorial information

Tutorial information introduces DB2 features and teaches how to perform various tasks.

The installation directory for this category is doc/htmlcd/%L/tutr.

Table 45. Tutorial information

Name	Form number	PDF file name
Business Intelligence Tutorial: Introduction to the Data Warehouse	No form number	db2tux80
Business Intelligence Tutorial: Extended Lessons in Data Warehousing	No form number	db2tax80
Development Center Tutorial for Video Online using Microsoft Visual Basic	No form number	db2tdx80
Information Catalog Center Tutorial	No form number	db2aix80
Video Central for e-business Tutorial	No form number	db2twx80
Visual Explain Tutorial	No form number	db2tvx80

Optional component information

The information in this category describes how to work with optional DB2 components.

The installation directory for this category is doc/htmlcd/%L/opt.

Table 46. Optional component information

Name	Form number	PDF file name
IBM DB2 Life Sciences Data Connect Planning, Installation, and Configuration Guide	GC27-1235	db2lsx80
IBM DB2 Spatial Extender User's Guide and Reference	SC27-1226	db2sbx80
IBM DB2 Universal Database Data Links Manager Administration Guide and Reference	SC27-1221	db2z0x80
IBM DB2 Universal Database Net Search Extender Administration and Programming Guide **Note:** HTML for this document is not installed from the HTML documentation CD.	SH12-6740	N/A

Release notes
The release notes provide additional information specific to your product's release and FixPak level. They also provides summaries of the documentation updates incorporated in each release and FixPak.

Table 47. Release notes

Name	Form number	PDF file name
DB2 Release Notes	See note.	See note.
DB2 Installation Notes	Available on product CD-ROM only.	Available on product CD-ROM only.

Note: The HTML version of the release notes is available from the Information Center and on the product CD-ROMs. To view the ASCII file on UNIX-based platforms, see the Release.Notes file. This file is located in the DB2DIR/Readme/%L directory, where *%L* represents the locale name and DB2DIR represents:
- /usr/opt/db2_08_01 on AIX
- /opt/IBM/db2/V8.1 on all other UNIX operating systems

Printing DB2 books from PDF files

You can print DB2 books from the PDF files on the *DB2 PDF Documentation* CD. Using Adobe Acrobat Reader, you can print either the entire book or a specific range of pages.

Prerequisites:

Ensure that you have Adobe Acrobat Reader. It is available from the Adobe Web site at www.adobe.com

Procedure:

To print a DB2 book from a PDF file:

1. Insert the *DB2 PDF Documentation* CD. On UNIX operating systems, mount the DB2 PDF Documentation CD. Refer to your *Quick Beginnings* book for details on how to mount a CD on UNIX operating systems.
2. Start Adobe Acrobat Reader.
3. Open the PDF file from one of the following locations:
   - On Windows operating systems:

     x:\doc*language* directory, where *x* represents the CD-ROM drive letter and *language* represents the two-character territory code that represents your language (for example, EN for English).
   - On UNIX operating systems:

     /cdrom/doc/*%L* directory on the CD-ROM, where */cdrom* represents the mount point of the CD-ROM and *%L* represents the name of the desired locale.

Ordering printed DB2 books

Procedure:

To order printed books:

- Contact your IBM authorized dealer or marketing representative. To find a local IBM representative, check the IBM Worldwide Directory of Contacts at www.ibm.com/planetwide
- Phone 1-800-879-2755 in the United States or 1-800-IBM-4YOU in Canada.
- Visit the IBM Publications Center at www.ibm.com/shop/publications/order

You can also obtain printed DB2 manuals by ordering Doc Packs for your DB2 product from your IBM Reseller. The Doc Packs are subsets of the manuals in the DB2 library selected to help you to get started using the DB2 product that

you purchased. The manuals in the Doc Packs are the same as those that are available in PDF format on the *DB2 PDF Documentation CD* and contain the same content as the documentation that is available on the *DB2 HTML Documentation CD*.

Accessing online help

The online help that comes with all DB2 components is available in three types:
- Window and notebook help
- Command line help
- SQL statement help

Window and notebook help explain the tasks that you can perform in a window or notebook and describe the controls. This help has two types:
- Help accessible from the **Help** button
- Infopops

The **Help** button gives you access to overview and prerequisite information. The infopops describe the controls in the window or notebook. Window and notebook help are available from DB2 centers and components that have user interfaces.

Command line help includes Command help and Message help. Command help explains the syntax of commands in the command line processor. Message help describes the cause of an error message and describes any action you should take in response to the error.

SQL statement help includes SQL help and SQLSTATE help. DB2 returns an SQLSTATE value for conditions that could be the result of an SQL statement. SQLSTATE help explains the syntax of SQL statements (SQL states and class codes).

Note: SQL help is not available for UNIX operating systems.

Procedure:

To access online help:
- For window and notebook help, click **Help** or click that control, then click **F1**. If the **Automatically display infopops** check box on the **General** page of the **Tool Settings** notebook is selected, you can also see the infopop for a particular control by holding the mouse cursor over the control.
- For command line help, open the command line processor and enter:
  - For Command help:

```
    ? command
```

where *command* represents a keyword or the entire command.

For example, ? `catalog` displays help for all the CATALOG commands, while ? `catalog database` displays help for the CATALOG DATABASE command.

- For Message help:
    ```
    ? XXXnnnnn
    ```

where *XXXnnnnn* represents a valid message identifier.

For example, ? `SQL30081` displays help about the SQL30081 message.

- For SQL statement help, open the command line processor and enter:
    ```
    ? sqlstate or ? class code
    ```

where *sqlstate* represents a valid five-digit SQL state and *class code* represents the first two digits of the SQL state.

For example, ? `08003` displays help for the 08003 SQL state, while ? `08` displays help for the 08 class code.

Finding topics by accessing the DB2 Information Center from a browser

The DB2 Information Center accessed from a browser enables you to access the information you need to take full advantage of DB2 Universal Database and DB2 Connect. The DB2 Information Center also documents major DB2 features and components including replication, data warehousing, metadata, and DB2 extenders.

The DB2 Information Center accessed from a browser is composed of the following major elements:

Navigation tree
> The navigation tree is located in the left frame of the browser window. The tree expands and collapses to show and hide topics, the glossary, and the master index in the DB2 Information Center.

Navigation toolbar
> The navigation toolbar is located in the top right frame of the browser window. The navigation toolbar contains buttons that enable you to search the DB2 Information Center, hide the navigation tree, and find the currently displayed topic in the navigation tree.

Content frame
> The content frame is located in the bottom right frame of the browser window. The content frame displays topics from the DB2 Information

Center when you click on a link in the navigation tree, click on a search result, or follow a link from another topic or from the master index.

Prerequisites:

To access the DB2 Information Center from a browser, you must use one of the following browsers:
- Microsoft Explorer, version 5 or later
- Netscape Navigator, version 6.1 or later

Restrictions:

The DB2 Information Center contains only those sets of topics that you chose to install from the *DB2 HTML Documentation CD*. If your Web browser returns a File not found error when you try to follow a link to a topic, you must install one or more additional sets of topics from the *DB2 HTML Documentation CD*.

Procedure:

To find a topic by searching with keywords:
1. In the navigation toolbar, click **Search**.
2. In the top text entry field of the Search window, enter one or more terms related to your area of interest and click **Search**. A list of topics ranked by accuracy displays in the **Results** field. The numerical ranking beside the hit provides an indication of the strength of the match (bigger numbers indicate stronger matches).

   Entering more terms increases the precision of your query while reducing the number of topics returned from your query.
3. In the **Results** field, click the title of the topic you want to read. The topic displays in the content frame.

To find a topic in the navigation tree:
1. In the navigation tree, click the book icon of the category of topics related to your area of interest. A list of subcategories displays underneath the icon.
2. Continue to click the book icons until you find the category containing the topics in which you are interested. Categories that link to topics display the category title as an underscored link when you move the cursor over the category title. The navigation tree identifies topics with a page icon.
3. Click the topic link. The topic displays in the content frame.

To find a topic or term in the master index:

1. In the navigation tree, click the "Index" category. The category expands to display a list of links arranged in alphabetical order in the navigation tree.
2. In the navigation tree, click the link corresponding to the first character of the term relating to the topic in which you are interested. A list of terms with that initial character displays in the content frame. Terms that have multiple index entries are identified by a book icon.
3. Click the book icon corresponding to the term in which you are interested. A list of subterms and topics displays below the term you clicked. Topics are identified by page icons with an underscored title.
4. Click on the title of the topic that meets your needs. The topic displays in the content frame.

Finding product information by accessing the DB2 Information Center from the administration tools

The DB2 Information Center provides quick access to DB2 product information and is available on all operating systems for which the DB2 administration tools are available.

The DB2 Information Center accessed from the tools provides six types of information.

Tasks   Key tasks you can perform using DB2.

Concepts
Key concepts for DB2.

Reference
DB2 reference information, such as keywords, commands, and APIs.

Troubleshooting
Error messages and information to help you with common DB2 problems.

Samples
Links to HTML listings of the sample programs provided with DB2.

Tutorials
Instructional aid designed to help you learn a DB2 feature.

Prerequisites:

Some links in the DB2 Information Center point to Web sites on the Internet. To display the content for these links, you will first have to connect to the Internet.

Procedure:

To find product information by accessing the DB2 Information Center from the tools:

1. Start the DB2 Information Center in one of the following ways:
   - From the graphical administration tools, click on the **Information Center** icon in the toolbar. You can also select it from the **Help** menu.
   - At the command line, enter **db2ic**.
2. Click the tab of the information type related to the information you are attempting to find.
3. Navigate through the tree and click on the topic in which you are interested. The Information Center will then launch a Web browser to display the information.
4. To find information without browsing the lists, click the **Search** icon to the right of the list.

   Once the Information Center has launched a browser to display the information, you can perform a full-text search by clicking the **Search** icon in the navigation toolbar.

Viewing technical documentation online directly from the DB2 HTML Documentation CD

All of the HTML topics that you can install from the *DB2 HTML Documentation CD* can also be read directly from the CD. Therefore, you can view the documentation without having to install it.

Restrictions:

As the Tools help is installed from the DB2 product CD and not from the *DB2 HTML Documentation CD*, you must install the DB2 product to view the help.

Procedure:

1. Insert the *DB2 HTML Documentation* CD. On UNIX operating systems, mount the *DB2 HTML Documentation CD*. Refer to your *Quick Beginnings* book for details on how to mount a CD on UNIX operating systems.
2. Start your HTML browser and open the appropriate file:
   - For Windows operating systems:
     ```
     e:\program files\IBM\SQLLIB\doc\htmlcd\%L\index.htm
     ```

     where *e* represents the CD-ROM drive, and %L is the locale of the documentation that you wish to use, for example, **en_US** for English.
   - For UNIX operating systems:
     ```
     /cdrom/program files/IBM/SQLLIB/doc/htmlcd/%L/index.htm
     ```

where */cdrom/* represents where the CD is mounted, and %L is the locale of the documentation that you wish to use, for example, **en_US** for English.

Updating the HTML documentation installed on your machine

It is now possible to update the HTML installed from the *DB2 HTML Documentation CD* when updates are made available from IBM. This can be done in one of two ways:

- Using the Information Center (if you have the DB2 administration GUI tools installed).
- By downloading and applying a DB2 HTML documentation FixPak .

Note: This will NOT update the DB2 code; it will only update the HTML documentation installed from the *DB2 HTML Documentation CD*.

Procedure:

To use the Information Center to update your local documentation:
1. Start the DB2 Information Center in one of the following ways:
   - From the graphical administration tools, click on the **Information Center** icon in the toolbar. You can also select it from the **Help** menu.
   - At the command line, enter **db2ic**.
2. Ensure your machine has access to the external Internet; the updater will download the latest documentation FixPak from the IBM server if required.
3. Select **Information Center** —> **Update Local Documentation** from the menu to start the update.
4. Supply your proxy information (if required) to connect to the external Internet.

The Information Center will download and apply the latest documentation FixPak, if one is available.

To manually download and apply the documentation FixPak :
1. Ensure your machine is connected to the Internet.
2. Open the DB2 support page in your Web browser at: www.ibm.com/software/data/db2/udb/winos2unix/support.
3. Follow the link for Version 8 and look for the "Documentation FixPaks" link.

4. Determine if the version of your local documentation is out of date by comparing the documentation FixPak level to the documentation level you have installed. This current documentation on your machine is at the following level: **DB2 v8.1 GA**.

5. If there is a more recent version of the documentation available then download the FixPak applicable to your operating system. There is one FixPak for all Windows platforms, and one FixPak for all UNIX platforms.

6. Apply the FixPak:
   - For Windows operating systems: The documentation FixPak is a self extracting zip file. Place the downloaded documentation FixPak in an empty directory, and run it. It will create a **setup** command which you can run to install the documentation FixPak.
   - For UNIX operating systems: The documentation FixPak is a compressed tar.Z file. Uncompress and untar the file. It will create a directory named `delta_install` with a script called **installdocfix**. Run this script to install the documentation FixPak.

Copying files from the DB2 HTML Documentation CD to a Web server

The entire DB2 information library is delivered to you on the *DB2 HTML Documentation CD* and may be installed on a Web server for easier access. Simply copy to your Web server the documentation for the languages that you want.

Note: You might encounter slow performance if you access the HTML documentation from a Web server through a low-speed connection.

Procedure:

To copy files from the *DB2 HTML Documentation CD* to a Web server, use the appropriate source path:

- For Windows operating systems:

  `E:\program files\IBM\SQLLIB\doc\htmlcd\%L\*.*`

  where *E* represents the CD-ROM drive and *%L* represents the language identifier.

- For UNIX operating systems:

  `/cdrom/program files/IBM/SQLLIB/doc/htmlcd/%L/*.*`

  where *cdrom* represents the mount point for the CD-ROM drive and *%L* represents the language identifier.

Troubleshooting DB2 documentation search with Netscape 4.x

Most search problems are related to the Java support provided by web browsers. This task describes possible workarounds.

Procedure:

A common problem with Netscape 4.x involves a missing or misplaced security class. Try the following workaround, especially if you see the following line in the browser Java console:

```
Cannot find class  java/security/InvalidParameterException
```

- On Windows operating systems:

  From the *DB2 HTML Documentation CD*, copy the supplied `x:`program `files\IBM\SQLLIB\doc\htmlcd\`*locale*`\InvalidParameterException.class` file to the `java\classes\java\security\` directory relative to your Netscape browser installation, where *x* represents the CD-ROM drive letter and *locale* represents the name of the desired locale.

  Note: You may have to create the `java\security\` subdirectory structure.

- On UNIX operating systems:

  From the *DB2 HTML Documentation CD*, copy the supplied `/cdrom/`program `files/IBM/SQLLIB/doc/htmlcd/`*locale*`/InvalidParameterException.class` file to the `java/classes/java/security/` directory relative to your Netscape browser installation, where *cdrom* represents the mount point of the CD-ROM and *locale* represents the name of the desired locale.

  Note: You may have to create the `java/security/` subdirectory structure.

If your Netscape browser still fails to display the search input window, try the following:

- Stop all instances of Netscape browsers to ensure that there is no Netscape code running on the machine. Then open a new instance of the Netscape browser and try to start the search again.
- Purge the browser's cache.
- Try a different version of Netscape, or a different browser.

Searching the DB2 documentation

You can search the library of DB2 documentation to locate information that you need. A pop-up search window opens when you click the search icon in the navigation toolbar of the DB2 Information Center (accessed from a browser). The search can take a minute to load, depending on the speed of your computer and network.

Prerequisites:

You need Netscape 6.1 or higher, or Microsoft's Internet Explorer 5 or higher. Ensure that your browser's Java support is enabled.

Restrictions:

The following restrictions apply when you use the documentation search:
- Search is not case sensitive.
- Boolean searches are not supported.
- Wildcard and partial searches are not supported. A search on *java** (or *java*) will only look for the literal string *java** (or *java*) and would not, for example, find *javadoc*.

Procedure:

To search the DB2 documentation:
1. In the navigation toolbar, click the **Search** icon.
2. In the top text entry field of the Search window, enter one or more terms (separated by a space) related to your area of interest and click **Search**. A list of topics ranked by accuracy displays in the **Results** field. The numerical ranking beside the hit provides an indication of the strength of the match (bigger numbers indicate stronger matches).

   Entering more terms increases the precision of your query while reducing the number of topics returned from your query.
3. In the **Results** list, click the title of the topic you want to read. The topic displays in the content frame of the DB2 Information Center.

Note: When you perform a search, the first (highest-ranking) result is automatically loaded into your browser frame. To view the contents of other search results, click on the result in the results list.

Online DB2 troubleshooting information

With the release of DB2® UDB Version 8, there will no longer be a *Troubleshooting Guide*. The troubleshooting information once contained in this guide has been integrated into the DB2 publications. By doing this, we are able to deliver the most up-to-date information possible. To find information on the troubleshooting utilities and functions of DB2, access the DB2 Information Center from any of the tools.

Refer to the DB2 Online Support site if you are experiencing problems and want help finding possible causes and solutions. The support site contains a large, constantly updated database of DB2 publications, TechNotes, APAR

(product problem) records, FixPaks, and other resources. You can use the support site to search through this knowledge base and find possible solutions to your problems.

Access the Online Support site at www.ibm.com/software/data/db2/udb/winos2unix/support, or by clicking the **Online Support** button in the DB2 Information Center. Frequently changing information, such as the listing of internal DB2 error codes, is now also available from this site.

Accessibility

Accessibility features help users with physical disabilities, such as restricted mobility or limited vision, to use software products successfully. These are the major accessibility features in DB2® Universal Database Version 8:

- DB2 allows you to operate all features using the keyboard instead of the mouse. See "Keyboard Input and Navigation".
- DB2 enables you customize the size and color of your fonts. See "Accessible Display".
- DB2 allows you to receive either visual or audio alert cues. See "Alternative Alert Cues" on page 618.
- DB2 supports accessibility applications that use the Java™ Accessibility API. See "Compatibility with Assistive Technologies" on page 618.
- DB2 comes with documentation that is provided in an accessible format. See "Accessible Documentation" on page 618.

Keyboard Input and Navigation

Keyboard Input
You can operate the DB2 Tools using only the keyboard. You can use keys or key combinations to perform most operations that can also be done using a mouse.

Keyboard Focus
In UNIX-based systems, the position of the keyboard focus is highlighted, indicating which area of the window is active and where your keystrokes will have an effect.

Accessible Display

The DB2 Tools have features that enhance the user interface and improve accessibility for users with low vision. These accessibility enhancements include support for customizable font properties.

Font Settings
The DB2 Tools allow you to select the color, size, and font for the text in menus and dialog windows, using the Tools Settings notebook.

Non-dependence on Color

You do not need to distinguish between colors in order to use any of the functions in this product.

Alternative Alert Cues

You can specify whether you want to receive alerts through audio or visual cues, using the Tools Settings notebook.

Compatibility with Assistive Technologies

The DB2 Tools interface supports the Java Accessibility API enabling use by screen readers and other assistive technologies used by people with disabilities.

Accessible Documentation

Documentation for the DB2 family of products is available in HTML format. This allows you to view documentation according to the display preferences set in your browser. It also allows you to use screen readers and other assistive technologies.

DB2 tutorials

The DB2® tutorials help you learn about various aspects of DB2 Universal Database. The tutorials provide lessons with step-by-step instructions in the areas of developing applications, tuning SQL query performance, working with data warehouses, managing metadata, and developing Web services using DB2.

Before you begin:

Before you can access these tutorials using the links below, you must install the tutorials from the *DB2 HTML Documentation* CD.

If you do not want to install the tutorials, you can view the HTML versions of the tutorials directly from the *DB2 HTML Documentation CD*. PDF versions of these tutorials are also available on the *DB2 PDF Documentation CD*.

Some tutorial lessons use sample data or code. See each individual tutorial for a description of any prerequisites for its specific tasks.

DB2 Universal Database tutorials:

If you installed the tutorials from the *DB2 HTML Documentation* CD, you can click on a tutorial title in the following list to view that tutorial.

Business Intelligence Tutorial: Introduction to the Data Warehouse Center
Perform introductory data warehousing tasks using the Data Warehouse Center.

Business Intelligence Tutorial: Extended Lessons in Data Warehousing
> Perform advanced data warehousing tasks using the Data Warehouse Center.

Development Center Tutorial for Video Online using Microsoft® Visual Basic
> Build various components of an application using the Development Center Add-in for Microsoft Visual Basic.

Information Catalog Center Tutorial
> Create and manage an information catalog to locate and use metadata using the Information Catalog Center.

Video Central for e-business Tutorial
> Develop and deploy an advanced DB2 Web Services application using WebSphere® products.

Visual Explain Tutorial
> Analyze, optimize, and tune SQL statements for better performance using Visual Explain.

DB2 Information Center accessed from a browser

The DB2® Information Center gives you access to all of the information you need to take full advantage of DB2 Universal Database™ and DB2 Connect™ in your business. The DB2 Information Center also documents major DB2 features and components including replication, data warehousing, the Information Catalog Center, Life Sciences Data Connect, and DB2 extenders.

The DB2 Information Center accessed from a browser has the following features if you view it in Netscape Navigator 6.1 or later or Microsoft Internet Explorer 5 or later. Some features require you to enable support for Java or JavaScript:

Regularly updated documentation
> Keep your topics up-to-date by downloading updated HTML.

Search
> Search all of the topics installed on your workstation by clicking **Search** in the navigation toolbar.

Integrated navigation tree
> Locate any topic in the DB2 library from a single navigation tree. The navigation tree is organized by information type as follows:
> - Tasks provide step-by-step instructions on how to complete a goal.
> - Concepts provide an overview of a subject.
> - Reference topics provide detailed information about a subject, including statement and command syntax, message help, requirements.

Master index

Access the information installed from the *DB2 HTML Documentation CD* from the master index. The index is organized in alphabetical order by index term.

Master glossary

The master glossary defines terms used in the DB2 Information Center. The glossary is organized in alphabetical order by glossary term.

Appendix C. Notices

IBM may not offer the products, services, or features discussed in this
document in all countries. Consult your local IBM representative for
information on the products and services currently available in your area. Any
reference to an IBM product, program, or service is not intended to state or
imply that only that IBM product, program, or service may be used. Any
functionally equivalent product, program, or service that does not infringe
any IBM intellectual property right may be used instead. However, it is the
user's responsibility to evaluate and verify the operation of any non-IBM
product, program, or service.

IBM may have patents or pending patent applications covering subject matter
described in this document. The furnishing of this document does not give
you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the
IBM Intellectual Property Department in your country/region or send
inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

**The following paragraph does not apply to the United Kingdom or any
other country/region where such provisions are inconsistent with local law:**
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS
PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER
EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY,
OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow
disclaimer of express or implied warranties in certain transactions; therefore,
this statement may not apply to you.

This information could include technical inaccuracies or typographical errors.
Changes are periodically made to the information herein; these changes will
be incorporated in new editions of the publication. IBM may make

improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product, and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information that has been exchanged, should contact:

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems, and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements, or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious, and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information may contain sample application programs, in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (*your company name*) (*year*). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. *_enter the year or years_*. All rights reserved.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both, and have been used in at least one of the documents in the DB2 UDB documentation library.

ACF/VTAM	LAN Distance
AISPO	MVS
AIX	MVS/ESA
AIXwindows	MVS/XA
AnyNet	Net.Data
APPN	NetView
AS/400	OS/390
BookManager	OS/400
C Set++	PowerPC
C/370	pSeries
CICS	QBIC
Database 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000
DataRefresher	S/370
DB2	SP
DB2 Connect	SQL/400
DB2 Extenders	SQL/DS
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational	SystemView
Database Architecture	Tivoli
DRDA	VisualAge
eServer	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WebExplorer
IBM	WebSphere
IMS	WIN-OS/2
IMS/ESA	z/OS
iSeries	zSeries

The following terms are trademarks or registered trademarks of other companies and have been used in at least one of the documents in the DB2 UDB documentation library:

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Index

A
accessing help   1
ADM messages   3
AUD messages   3

D
DIA messages   3

G
GOV messages   3

H
Help   1

I
introduction to messages   1

M
message prefix   1
message structure   1
message suffix   1
Messages   1
   ADM   3
   AUD   3
   DIA   3
   GOV   3
   SQL   5
   SQLSTATE   555
messages, introduction   1

O
online help   1
other message sources   4

S
SQL messages   5
SQLSTATE messages   555

Contacting IBM

In the United States, call one of the following numbers to contact IBM:

- 1-800-237-5511 for customer service
- 1-888-426-4343 to learn about available service options
- 1-800-IBM-4YOU (426-4968) for DB2 marketing and sales

In Canada, call one of the following numbers to contact IBM:

- 1-800-IBM-SERV (1-800-426-7378) for customer service
- 1-800-465-9600 to learn about available service options
- 1-800-IBM-4YOU (1-800-426-4968) for DB2 marketing and sales

To locate an IBM office in your country or region, check IBM's Directory of Worldwide Contacts on the web at www.ibm.com/planetwide

Product information

Information regarding DB2 Universal Database products is available by telephone or by the World Wide Web at www.ibm.com/software/data/db2/udb

This site contains the latest information on the technical library, ordering books, client downloads, newsgroups, FixPaks, news, and links to web resources.

If you live in the U.S.A., then you can call one of the following numbers:

- 1-800-IBM-CALL (1-800-426-2255) to order products or to obtain general information.
- 1-800-879-2755 to order publications.

For information on how to contact IBM outside of the United States, go to the IBM Worldwide page at www.ibm.com/planetwide

IBM ®

Printed in U.S.A.

Spine information:

IBM® DB2 Universal Database™   Message Reference, Volume 2

Version 8

IBM