IBM File Export for z/OS

Messages and Codes

Version 1 Release 1

SC31-6871-01

IBM File Export for z/OS

Messages and Codes

Version 1 Release 1

SC31-6871-01

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

Second Edition (January 2005)

This edition applies to Version 1 Release 1 of IBM File Export for z/OS (product number 5697-I12) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

About this book v
Who should read this book v
Conventions used in this book v
Terminology used in this book vi
Service updates and support information vi
Using LookAt to obtain message explanations. vi
Where to find information vi
Accessibility features vii
How to send your comments vii

Part 1. Introduction 1

Chapter 1. Introduction 3
Overview 3
Using IBM File Export 3
Sources, targets, and associated ddnames. 3
Data manipulation 4
Finding more information 5

Part 2. Messages and Return Codes . 7

Chapter 2. Messages ABX006--ABX1099 . 9

Chapter 3. Messages ABX1100--ABX1887 33

Chapter 4. Messages ABX2101--ABX2841 65

Chapter 5. Messages ABX8001--ABX9999 109

Chapter 6. Return Codes. 139
Repository XML Loader Return Codes 139

Part 3. Appendixes. 141

Appendix A. Notices 143
Trademarks. 145

Glossary 147

About this book

This book provides instructions for installing and using IBM File Export for z/OS.

This book is designed to help database administrators, system programmers, application programmers, and system operators perform these tasks:

- Plan for the installation of IBM File Export for z/OS
- Install and operate IBM File Export for z/OS
- Customize your IBM File Export for z/OS environment
- Diagnose and recover from IBM File Export for z/OS problems
- Use IBM File Export for z/OS with other DB2 or IMS products

Always check the DB2 and IMS Tools Library page for the most current version of this publication: www.ibm.com/software/data/db2imstools/library.html

Who should read this book

This book is intended for those persons responsible for installing, customizing, and using IBM File Export for z/OS and assumes a working knowledge of:

- The OS/390® or z/OS® operating system
- ISPF
- SMP/E
- IMS
- DB2

Conventions used in this book

This book uses the following highlighting conventions:

- **Boldface type** indicates commands or user interface controls such as names of fields, folder, icons, or menu choices.
- `Monospace type` indicates examples of text that you enter exactly as shown.
- *Italic type* indicates variables that you should replace with a value. It is also used to indicate book titles and to emphasize significant words.

The following labels identify significant elements within this book:

- **Definition:** is used to identify and define terminology unique to this product.
- **Example:** is used to identify example code or scenarios.
- **In this chapter:** is used to identify the significant subsections within each chapter.
- **Recommendation:** is used to provide guidance when more than one option is available.
- **Related reading:** is used to refer you to other publications that contain relevant information.
- **Restriction:** is used to identify a restriction or limitation with this product or an associated procedure.

Terminology used in this book

In this book, IBM File Export for z/OS is referred to as "IBM IBM File Export." In cases where the context makes the meaning clear, IBM IBM File Export is referred to as "IBM File Export."

The following terms are used as indicated:

DB2® Represents either the DB2 licensed program or a particular DB2 subsystem.

IMS Represents the IMS licensed program.

In general, for simplicity, the term *file* as used in this manual refers to IMS, VSAM or sequential files, as well as DB2 tables, unless these are specifically differentiated.

Service updates and support information

To find service updates and support information, including software Fix Packs, PTFs, Frequently Asked Question (FAQs), technical notes, troubleshooting information, and downloads, refer to the following Web page:

http://www.ibm.com/software/awdtools/fileexport/

Using LookAt to obtain message explanations

LookAt is an online facility that displays explanations for most messages that you encounter, as well as for some system abends and codes. You can access LookAt:

- From the Internet at:

  www.ibm.com/eserver/zseries/zos/bkserv/lookat/

- From anywhere in z/OS where you can access a TSO/E command line (for example, a TSO/E prompt, ISPF, or z/OS UNIX System Services running OMVS). To use LookAt as a TSO/E command, LookAt must be installed on your host system. Obtain the LookAt code for TSO/E from a disk on your z/OS Collection (SK3T-4269) or from the LookAt ftp site:

  ftp.software.ibm.com/ps/products/ibmreader/tools/lookat/ZOS

- From your Palm™ VIIx personal data assistant (PDA). To use LookAt from your Palm VIIx PDA, LookAt must be installed on the PDA. Obtain the LookAt code from a disk on your z/OS Collection (SK3T-4269) or from the LookAt ftp site:

  ftp.software.ibm.com/ps/products/ibmreader/tools/lookat/PALM/

Where to find information

The IBM File Export for z/OS library comprises three manuals. These are:

- File Export for z/OS*: User's Guide*
- File Export for z/OS*: Reference*
- File Export for z/OS*: Messages and Codes*

In addition, each panel in the ISPF user interface has an associated online Help panel for your information.

The Web page listed below provides the most current product documentation that you can view, print, and download.

http://www.ibm.com/software/awdtools/fileexport/

Accessibility features

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use a software product successfully. The major accessibility features in IBM IBM File Export enable users to:

- Use assistive technologies such as screen readers and screen magnifier software. Consult the assistive technology documentation for specific information when using it to access z/OS interfaces.
- Customize display attributes such as color, contrast, and font size.
- Operate specific or equivalent features using only the keyboard. Refer to the z/OS ISPF User's Guide for information about accessing ISPF interfaces. This guide describes how to use ISPF, including the use of keyboard shortcuts or function keys (PF keys), includes the default settings for the PF keys, and explains how to modify their functions.

How to send your comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this book or any other IBM File Export documentation, print and fill out the reader comment form located at the back of this book. You can use the following methods to return the form:

- Give it to your local IBM branch office or IBM representative.
- Send it to the address printed on the reader comment form.
- Send your comments by e-mail to comments@us.ibm.com.

Part 1. Introduction

Chapter 1. Introduction 3
Overview 3
Using IBM File Export 3
Sources, targets, and associated ddnames 3
Data manipulation 4
Finding more information 5

This part contains an overview of IBM File Export for z/OS.

Chapter 1. Introduction

Use IBM File Export for z/OS to extract DB2, IMS, VSAM, and sequential data from one or more sources and create one or more targets. (In general, for simplicity, the term file as used in this manual refers to IMS, VSAM or sequential files, as well as DB2 tables unless these are specifically differentiated.)

Overview

IBM File Export gathers data from the sources you designate, applies data manipulation rules that you specify, and then generates the desired output targets. The term *source* refers to input data, such as an existing DB2 table or an existing IMS, VSAM, or sequential file. The term *target* refers to output data.

IBM File Export offers highly selective data copying capabilities designed to simplify populating test environments or migrating applications to new or additional environments. IBM File Export permits you to:

- Support DB2, IMS, VSAM, and sequential sources in one product
- Automatically export all or a subset of your production data for use in your test environment using highly-flexible data selection and manipulation rules
- Extract or create files or tables that demonstrate one-to-one, one-to-many, and many-to-one relationships
- Define application relationships between dissimilar file types
- Support DB2-defined and application-defined referential integrity (RI)
- Expedite data migration and simplify data consolidation. An occasion where you might want to do this is the case where Company X has acquired Company Y and they use different databases.
- Modify the content of files or tables during export, including the ability to augment field values and age related data. For example, you can use IBM File Export for z/OS to delete rows from target DB2 tables based on keys extracted from DB2, IMS, VSAM, or sequential sources.
- Protect sensitive data by reproducible "scrambling" on a field-by-field basis

Using IBM File Export

As a user, you will interface to IBM File Export by means of TSO/ISPF or by creating and running batch JCL jobs. Use the TSO/ISPF panels for "fill-in-the-blanks" specification of sources and targets, and the automatic generation of IBM File Export for z/OS processing instructions (control syntax). Alternatively, you can quickly learn File Export's control syntax to write IBM File Export for z/OS JCL streams from scratch. Information on the ISPF user interface and IBM File Export's control syntax is found in the File Export for z/OS*: Reference*.

Sources, targets, and associated ddnames

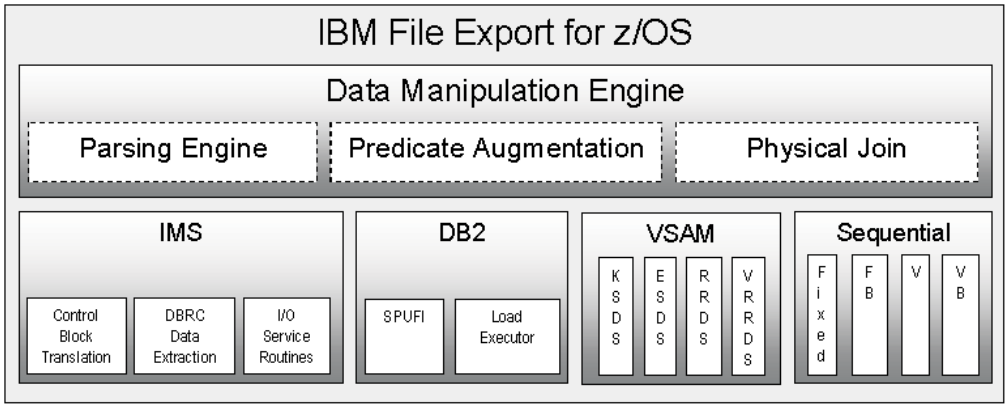
IBM File Export supports DB2, IMS (both full-function and Fast Path), VSAM, and sequential sources and targets. Well-recognized ddnames are used to identify the primary source, primary target, and data manipulation instructions to IBM File Export for z/OS.

- ABXIN identifies the primary source
- ABXOUT is the primary target
- ABXCTL is used to specify data gathering and manipulation instructions

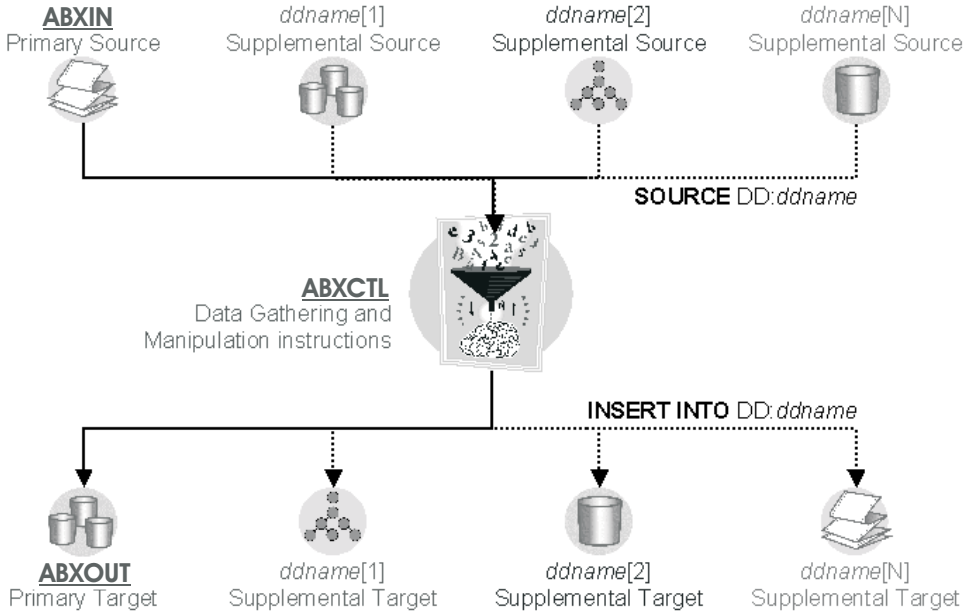
For information on other ddnames that are used by IBM File Export for z/OS, see the File Export for z/OS*: Reference*.

Data manipulation

Regardless of source or target file format, instructions provided to IBM File Export are processed by the Data Manipulation Engine which handles all aspects of data gathering, manipulation and generation.



For the purpose of extracting data from sources that are not relational, IBM File Export for z/OS has drawn on the syntax of the SQL (Structured Query Language) definitions of SELECT and INSERT. IBM File Export for z/OS reads data from a source or several sources, processes instructions which look very familiar to anyone acquainted with SQL, and then writes output to the requested target or targets, generating batch streams to automatically extract production data and load it into a test environment.



Finding more information

To find more information about IBM File Export, refer to the other manuals comprising the documentation set. These are

- *IBM File Export for z/OS: User's Guide*. This manual assists you in getting started with the product by providing a high-level overview of the File Export user interface and a number of examples of IBM File Export's use.
- *IBM File Export for z/OS: Reference Manual*. This manual documents the panels that comprise the user interface, describing the panel's fields and the valid entries you supply to define your export requests. It also documents the control statement syntax provided with ABXUTIL, the program by which you can define control statements to create JCL directly without using the product's ISPF user interface.

  This manual also contains the information required to install IBM File Export for z/OS
- *IBM File Export for z/OS: Messages and Codes*. This manual documents the messages produced by the product.

IBM File Export also provides online Help for each of the ISPF panels comprising the user interface to the product.

Part 2. Messages and Return Codes

Chapter 2. Messages ABX006--ABX1099 9

Chapter 3. Messages ABX1100--ABX1887 33

Chapter 4. Messages ABX2101--ABX2841 65

Chapter 5. Messages ABX8001--ABX9999 109

Chapter 6. Return Codes 139
Repository XML Loader Return Codes 139

This part contains information about the messages and return codes issued by IBM File Export for z/OS.

Chapter 2. Messages ABX006--ABX1099

All messages generated by IBM File Export for z/OS have a severity code printed as the last character of the message ID. The severity codes are described in the following table:

Table 1. Error message severity codes

Severity Code	Description
I	Information only. No user action required.
W	Warning message. Results may not be as expected.
E	Error message. Some may be user-correctable, read the User Response to determine the course of action.
S	Severe error -- note the information in the message and contact your support personnel.

ABX006E Invalid selection

Explanation: No load option has been specified on this panel.

System Action: The system waits for entry of a load option.

User Response: Select an appropriate load option before pressing Enter.

ABX010E You can't override global Force value for Auditing

Explanation: You have attempted to override the global auditing definition FORCE.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX018E Error during loading of DB2 table list

Explanation: An error has occurred when the DB2 table list was loaded.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX019E DB2 Table *tableid* not found

Explanation: The table ID you specified could not be found

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX021E You have to choose only one input type

Explanation: You have chosen more than one input type

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX022E You have to choose only one output type

Explanation: You have chosen more than one output type

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX023E Input type not specified

Explanation: You have not specified an input type.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX024E Output type is not specified

Explanation: You have not specified an output type.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX025E Invalid value - Please enter Storage Name

Explanation: You must enter a storage name to continue.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX026E Invalid value - Please enter DB2 Subsystem ID

Explanation: You must enter a DB2 subsystem ID to continue.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX027E Invalid value - Please enter Plan Name

Explanation: You must enter a DB2 plan name to continue.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX028E Invalid value - Please enter Data Set Name

Explanation: You must enter a data set name to continue.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX029E Error during loading columns for DB2 table *tableid*

Explanation: An error has occurred when loading DB2 table columns.

System Action: The program ends.

User Response: Correct the error and retry.

ABX031E You cannot edit DEFAULT type

Explanation: You have attempted to edit the DEFAULT type.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX032E You have to specify SQL Output DSN

Explanation: You must specify an output data set name to continue.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX033I Operation completed successfully.

Explanation: The current operation has completed successfully.

System Action: None required.

User Response: None required.

ABX034E You cannot modify DEFAULT record

Explanation: You have attempted to modify the DEFAULT record.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX035E You cannot delete DEFAULT record

Explanation: You have attempted to delete a DEFAULT record.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX036E You cannot unlink DEFAULT record

Explanation: You have attempted to unlink a DEFAULT record.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX037E **You cannot modify DEFAULT field**

Explanation: You have attempted to modify a DEFAULT field.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX038E **You cannot delete DEFAULT field**

Explanation: You have attempted to delete a DEFAULT field.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX043E **ABX API Error:** ***error message***

Explanation: An error occurred while processing the File Export Application Program Interface. The type of error is explained in the message.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX044E **Invalid Value -- Please enter Repository HLQ**

Explanation: You have entered an invalid repository high level qualifier.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX045E **Invalid Value -- Please enter Storage Name**

Explanation: You have entered an invalid or blank Storage Name.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX046E **Invalid Value -- Please enter DB2 Subsystem ID**

Explanation: You have entered an invalid or blank DB2 Subsystem ID.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX047E **Invalid Value -- Please enter Plan Name**

Explanation: You have entered an invalid or blank plan name.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX048E **Invalid Value -- Please enter Data Set Name**

Explanation: You have entered an invalid or blank data set name.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX049E **Error during loading columns for DB2 table** ***tableid***

Explanation: An error has occurred when loading DB2 table columns.

System Action: The program ends.

User Response: Correct the error and retry.

ABX050E **Invalid Value -- Please enter Record Name**

Explanation: You have entered an invalid or blank record name.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX051E **Invalid Value -- Please enter Field Name**

Explanation: You have entered an invalid or blank field name.

System Action: The system waits for you to correct the error.

User Response: Correct the error and retry.

ABX0550E **Required DD statement missing** ***ddname***

Explanation: The required DD statement described was not found.

System Action: The step terminates.

User Response: Include the DD statement in the step JCL and rerun the step.

ABX0551E **DBD *dbdname* missing from *ddname* concatenation**

Explanation: The DBD name described was not found within the data set concatenation for the DD statement named.

System Action: The step terminates.

User Response: Include the data set that contains the DBD in the DD statement concatenation. Rerun the step.

ABX0552E **BLDL (SVC 19) failed for *member type* member *member_name* -- Return code: *rc* Reason code: *rs***

Explanation: An attempt to perform a BLDL on the member named failed.

System Action: The step terminates.

User Response: Review the Completion Codes for the BLDL macro in the MVS Macro Instructions for Data Sets manual for a description of the problem. Correct any errors and rerun the step.

ABX0553E **LOAD (SVC 08) failed for *member type* member *member_name* -- Return code: *rc* Reason code: *rs***

Explanation: An attempt to perform a LOAD on the member named failed.

System Action: The step terminates.

User Response: Review the Return and Reason Codes for the LOAD macro in the MVS Assembler Services Reference manual for a description of the problem. Correct any errors and rerun the step.

ABX0554E **GETMAIN failed for *storage_type* Storage required *number* bytes.**

Explanation: An attempt was made to acquire the described amount of storage for the named storage type. The attempt failed.

System Action: The step terminates.

User Response: Increase the region size and rerun the step.

ABX0555E **DEVTYPE (SVC 24) failed for DDNAME *ddname* -- Return code: *rc* Reason code: *rs***

Explanation: An attempt to perform a DEVTYPE on the DD statement described failed.

System Action: The step terminates.

User Response: Review the Return and Reason Codes for the DEVTYPE macro in the DFSMSdfp Advanced Services manual for a description of the problem. Correct any errors and rerun the step.

ABX0556E **Invalid DBD name specified *dbdname***

Explanation: An invalid DBD name was supplied to program ABXDRIO0.

System Action: The step terminates.

User Response: Ensure that a valid DBD name was specified.

ABX0557E **Record with an invalid length of *length* encountered**

Explanation: An attempt to write a record failed as the length of the record was determined to be invalid.

System Action: The step terminates.

User Response: Unknown.

ABX0558E **Dynamic Allocation (SVC 99) function (*function*) failed for DD *ddname* -- Data set name: *dataset_name* Return code: *rc* Reason code: *rs***

Explanation: An attempt to perform the described SVC 99 function on the DD and data set names failed.

System Action: The step terminates.

User Response: Review the Interpreting DYNALLOC Return Codes section of the MVS Auth Assembler Services Guide. Correct any errors and rerun the step.

ABX0559E **MDA Member *member_name* is an unrecognized format**

Explanation: An IMS Dynamic Allocation member was successfully read but it is in a format which is unrecognized.

System Action: The step terminates.

User Response: Verify that the member name supplied is a valid IMS MDA member.

ABX0560E **Dynamic Allocation (SVC 99) function (*function*) failed for DD *ddname* Return code: *rc* Reason code: *rs***

Explanation: An attempt to perform the described SVC 99 function on the named DD failed.

System Action: The step terminates.

User Response: Review the Interpreting DYNALLOC Return Codes section of the MVS Auth Assembler Services Guide. Correct any errors and rerun the step.

ABX0561E **VSAM "GET" failed for DD *ddname* DATASET *dsname* -- Return code: *rc* Reason code: *rs***

Explanation: An attempt to retrieve a record from a VSAM data set failed.

System Action: The step terminates.

User Response: Review Record Management Return and Reason Codes in the MVS Macro Instructions for Data Sets manual for a description of the problem. Correct any errors and rerun the step.

ABX0562E **VSAM "OPEN" failed for DD *ddname* Data set *dsname* Return code: *rc* Reason code: *rs***

Explanation: An attempt to perform an OPEN on a VSAM data set failed.

System Action: The step terminates.

User Response: Review OPEN Return and Reason Codes in the MVS Macro Instructions for Data Sets manual for a description of the problem. Correct any errors and rerun the step.

ABX0563E **VSAM "GENCB" failed when building an *block type* -- Return code: *rc* Reason code: *rs***

Explanation: An attempt to generate a VSAM control block failed.

System Action: The step terminates.

User Response: Review Record Management Return and Reason Codes in the MVS Macro Instructions for Data Sets manual for a description of the problem. Correct any errors and rerun the step.

ABX0564E **VSAM "PUT" failed for DD *ddname* Data set *dsname* Return code: *rc* Reason code: *rs***

Explanation: An attempt to write a record to a VSAM data set failed.

System Action: The step terminates.

User Response: Review Record Management Return and Reason Codes in the MVS Macro Instructions for Data Sets manual for a description of the problem. Correct any errors and rerun the step.

ABX0565E **OPEN failed for DD *ddname***

Explanation: An attempt to OPEN the described DD statement failed.

System Action: The step terminates.

User Response: View any additional MVS-generated error messages. Correct any errors and rerun the step.

ABX0566E **Too many external references for DBD *dbdname* -- Limit is 256**

Explanation: When building a list of external database references for the DBD named, more than 256 external databases were externally referenced.

System Action: The step terminates.

User Response: This database cannot be used with the product. If this message was issued in error, please contact support.

ABX0567E **SWAREQ failed for DD *ddname* -- Return code *rc***

Explanation: An attempt to determine the address of the SEA failed.

System Action: The step terminates.

User Response: Determine the reason for the SWAREQ failure by referencing the Return code supplied in the MVS Programming: Authorized Assembler Services Guide. Correct any errors and rerun the step.

ABX0568E **VSAM input file *ddname* dataset *dsname* is empty**

Explanation: The VSAM input file described has a high-RBA value of zero.

System Action: The step terminates.

User Response: Provide an input data set containing records, or supply the LOAD parameter for the OPTIONS keyword in the ABXCTL stream.

ABX0569E **File specified by the *ddname* DD is not a VSAM data set**

Explanation: When attempting to READ from or WRITE to a VSAM data set, a non-VSAM data set was encountered.

System Action: The step terminates.

User Response: Provide the correct format data set. Correct the OPTIONS parameter in the ABXCTL input stream to specify the correct file type.

ABX0570E **Message table *msgtbl_name* not found -- Attempting to use ABXMTBL0. --Processing terminated.**

Explanation: An attempt to load the named message table failed.

System Action: If the named message table is not the default message table (ABXMTBL0), an attempt will be made to use ABXMTBL0. If successful, processing continues. If the attempt fails, or if the table named ABXMTBL0 is not found, the step terminates.

User Response: Ensure that the named message table is in the LINK, JOB, or STEPLIB data set concatenations. Rerun the step.

ABX0571E **Output VSAM file is not empty -- Data set: *dsn***

Explanation: When processing a VSAM output file, it was determined that the high-allocated RBA was not zero.

System Action: The step terminates.

User Response: Processing VSAM file types RRDS and VRDS requires that the output VSAM cluster be empty. Delete and redefine the VSAM cluster and rerun the step.

ABX0572E **DBD *dbdname* is not supported as it is a *type* database.**

Explanation: The database named is a database type that is not supported by this feature of the product.

System Action: The step terminates.

User Response: Review product documentation for supp[orted database types.

ABX0573E **HDAM input data set contains an invalid header record -- Dataset: *dsname* Record number: *ctr***

Explanation: When reading an HD UNLOAD format input data set, an invalid header record was encountered.

System Action: The step terminates.

User Response: Ensure that the input data set named is indeed an HD UNLOAD format file. Ensure that HD UNLOAD files are not concatenated. Examine the record number described. Correct any errors and rerun the step.

ABX0574E **HDAM input data set contains partition database migration data -- Partition Migration data is not supported. -- Dataset: *dsname* Record number: *ctr***

Explanation: When reading an HD UNLOAD format file, the partition migration bit in the header record was found to be on.

System Action: The step terminates.

User Response: HD UNLOAD data used to migrate to partitioned databases is not supported. Review product documentation for sample JCL to perform migration to partitioned databases.

ABX0575E **Segment name *seg_name* does not exist in database *dbdname* as segcode *segcode (X'segcode')***

Explanation: An attempt was made to write a segment to the HD UNLOAD ABXOUT file, but the supplied segment name and segment code pair does not match the pair in the database.

System Action: The step terminates.

User Response: Contact product support.

ABX0580I **Input processing started using *type* dataset *data set name***

Explanation: ABXUTIL has started reading input data from the named data set.

System Action: Processing continues.

User Response: None required.

ABX0581I **Output processing started using *type* dataset *data set name***

Explanation: ABXUTIL has started writing output to the named data set.

System Action: Processing continues.

User Response: None required.

ABX0582I **Input processing started using HD-UNLOAD data set *data set name* -- Using *type* Database *database name* from *dbdlib name* DD**

Explanation: ABXUTIL has started reading input data from the named IMS HD-UNLOAD data set.

System Action: Processing continues.

User Response: None required.

ABX0583I **Output processing started using HD-UNLOAD data set *data set name* --**

Using *type* Database *database name* from *dbdlib name* DD

Explanation: ABXUTIL has started writing output data to the data set named in IMS HD-UNLOAD format.

System Action: Processing continues.

User Response: None required.

ABX0584I **Input processing completed from data set *data set name***

Explanation: ABXUTIL has completed reading all the required data from the named data set.

System Action: Processing continues.

User Response: None required.

ABX0585I **Output processing completed for data set *data set name***

Explanation: ABXUTIL has completed writing data to the named data set.

System Action: Processing continues.

User Response: None required.

ABX0586E **RDJFCB (SVC64) failed for DDNAME *ddname* ---- Return Code*rc***

Explanation: An attempt to perform an RDJFCB on the DD named failed.

System Action: Processing terminates.

User Response: Review the Return and Reason Codes for the RDJFCB macro in the DFSMSdfp Advanced Services manual for a description of the problem. Correct any errors and rerun the step.

ABX0587E **Output DD *ddname* missing DCB attributes -- RECFM and LRECL are required in LOAD mode.**

Explanation: LOAD mode processing requires the output data set have the LRECL and RECFM DCB parameters specified.

System Action: Processing terminates.

User Response: Add the desired RECFM and LRECL parameters to the specified DD statement and rerun the step.

ABX0588E **DBD *dbdname* found in the *type* data set is not supported.**

Explanation: This database type is not supported by this feature of the product.

System Action: Processing terminates.

User Response: Review the product documentation for supported database types.

ABX0589E **VSAM "TEXTCB" failed for DD *ddname* ---- Return Code *rc* Reason Code *rs***

Explanation: An attempt to execute a TESTCB on the DD named failed.

System Action: Processing terminates.

User Response: Review the Record Management Return and Reason Codes in the MVS Macro Instructions for Data Sets manual for a description of the problem. Correct any errors and rerun the step.

ABX0590E **VSAM file type defined by the DD *ddname* is not supported.**

Explanation: An attempt was made to process an unsupported VSAM file type.

System Action: The step terminates.

User Response: VSAM file types supported are ESDS, KSDS, RRDS, and VRDS.

ABX0591E **File specified by the *ddname* DD has an unsupported file type.**

Explanation: The data set referenced by the DDname is not supported by this product.

System Action: The step terminates.

User Response: Supported sequential file types are fixed and variable.

ABX0593E **Output segment *segment_name* missing parent segment *parent_segment_name***

Explanation: While writing IMS HD-UNLOAD format data, it was determined that the segment hierarchical structure was incomplete.

System Action: The step terminates.

User Response: Review the input and the processing options for indications as to why the segment's parent is missing.

ABX0594E **Output segment *segment_name* out of twin sequence.**

Explanation: While writing IMS segment data, it was determined that the twin segment pointer chain was not in the correct sequence.

System Action: The step terminates.

User Response: Review the input and the processing options for indications as to why the segments are out of twin sequence.

ABX0595E **Inconsistent segment information between *ddname* header records and *dbdname* definition from *dbdlib* DD**

Explanation: A comparison between the IMS HD-UNLOAD header record and the DBDLIB member for the DBD named found an inconsistency.

System Action: The step terminates.

User Response: Review the input and the DBDLIB member to determine what the inconsistency is. It may lie in segment names, segment code, or segment hierarchical levels.

ABX0596E **Output segment *segment_name* out of hierarchical sequence.**

Explanation: While writing IMS segment data, it was determined that the segment's hierarchical sequence was incorrect.

System Action: The step terminates.

User Response: Review the input and the processing options for indications as to why the segments are out of hierarchical sequence.

ABX0597E **DD statement *ddname* cannot refer to "dummy."**

Explanation: The referenced DD statement is defined as DD DUMMY.

System Action: The step terminates.

User Response: Correct the DD statement specification.

ABX0598E **Invalid MODSTAT ID *(ID)* encountered in DD *ddname***

Explanation: While attempting to determine the most current ACBLIB data set, an invalid MODSTAT ID was encountered.

System Action: The step terminates.

User Response: Specify a valid MODSTAT data set.

ABX0599E **Member *dbdname* from DD *ddname* is not a *type* DMB**

Explanation: While attempting to process the DBD named, of the type specified, it was determined that the DMB type was not consistent with the request.

System Action: The step terminates.

User Response: Specify a DBD name of the correct type.

ABX0600E **Invalid ACBLIB DD Name *ddname* encountered in DD MODSTAT**

Explanation: While attempting to determine the most current ACBLIB data set, an invalid name was encountered in the ACBLIB DD name section of the MODSTAT data set.

System Action: The step terminates.

User Response: Specify a valid MODSTAT data set.

ABX0601E **DD *ddname* does not contain DMB information for DBD *dbdname***

Explanation: While reading the information contained in the named DBD, it was determined that the member name and internal DBD name were not consistent.

System Action: The step terminates.

User Response: Specify the correct data set for the DD named.

ABX0602E **Insufficient storage available to open VSAM dataset *dataset_name***

Explanation: While attempting to open the VSAM data set named, it was determined that there was not enough storage available.

System Action: The step terminates.

User Response: Increase the region size.

ABX0603E **An invalid type was supplied for DD *ddname***

Explanation: While attempting to process the DDname referenced, it was determined that the data set type was not valid for the processing requested.

System Action: The step terminates.

User Response: Specify a valid data set type for the processing requested.

ABX0604E **This step is not APF-Authorized**

Explanation: The step is not APF-authorized.

System Action: Processing terminates.

User Response: Review the installation process and determine why the step is not APF-authorized.

ABX0605E **AUDIT is FORCE and an SMF error was encountered. SMF reason= *reason_code*.**

Explanation: SMF audit was set to FORCE and a condition was encountered which prevented SMF auditing.

System Action: The step terminates.

User Response: Review the output to determine what caused the event to occur.

ABX0606E **Output DD *ddname* requires a RECFM of *recfm* and an LRECL of *lrecl* .**

Explanation: It was determined that the data set specified by the output DD statement indicated that either the RECFM or LRECL specified was unsuitable for the records to be produced.

System Action: The step terminates.

User Response: Do not specify a RECFM or LRECL. Allow the program to determine the optimal values or specify the RECFM and LRECL indicated.

ABX0607I **Skipping records for database *dbdname* as requested.**

Explanation: While executing program ABXFELD0, an EXCLDBD statement was encountered and this message indicates that the records for this database will be skipped.

System Action: Processing continues while bypassing inserts for the named database.

User Response: None.

ABX0608E **Processing prematurely terminated after processing *ctr* input records.**

Explanation: While executing program ABXFELD0, an error was encountered causing the step to terminate before processing all of the input records.

System Action: The step terminates.

User Response: Review any other messages that were issued and take corrective action.

ABX0609E **ABXFELD0 encountered an invalid record type in dataset *dataset_name* -- Record number : *ctr*.**

Explanation: While executing program ABXFELD0, an invalid record was encountered.

System Action: The step terminates.

User Response: Ensure that the data set supplied was created by program ABXUTIL with an output mode of FEUNLOAD.

ABX0610E **ABXFELD0 encountered an invalid record type in dataset *dataset_name* -- Record number : *ctr*.**

Explanation: While executing program ABXFELD0, an invalid record was encountered.

System Action: The step terminates.

User Response: Ensure that the data set supplied was created by program ABXUTIL with an output mode of FEUNLOAD.

ABX0611E **All databases have been EXCLUDED from processing - Step is terminating.**

Explanation: While executing program ABXFELD0, an EXCLDBD statement was encountered for every database contained in the input data set.

System Action: Processing terminates.

User Response: Remove the EXCLDBD statements for those databases you want to be loaded.

ABX0612E **ABXFELD0 encountered an invalid keyword *keyword*.**

Explanation: While executing program ABXFELD0, an invalid keyword was found in the control statement input stream.

System Action: The step terminates.

User Response: Correct the control syntax error.

ABX0613E **Invalid parameter supplied for keyword *keyword*.**

Explanation: While executing program ABXFELD0, a keyword supplied in the control statement input stream contained an invalid parameter.

System Action: The step terminates.

User Response: Correct the parameter in error.

ABX0614E **ABXFELD0 encountered an invalid control statement.**

Explanation: While executing program ABXFELD0, a control statement was encountered that was in error.

System Action: The step terminates.

User Response: Correct the statement in error.

ABX0615E **Duplicate keyword *keyword* supplied.**

Explanation: While executing program ABXFELD0, a keyword that may be specified only once per job step was encountered multiple times.

System Action: The step terminates.

User Response: Remove the offending duplicate keyword.

ABX0616E **ABXFELD0 required control statement *keyword* missing.**

Explanation: While executing program ABXFELD0, a control statement keyword that is required was not supplied.

System Action: The step terminates.

User Response: Include the required keyword in the control statement input stream.

ABX0617E **ABXFELD0 requires the use of Dynamic PSBs -- This feature is not available.**

Explanation: The execution of ABXFELD0 requires the use of the Dynamic PSB feature.

System Action: The step terminates.

User Response: Change the IMS options block or control statement syntax to include required keywords to enable the Dynamic PSB feature.

ABX0618E **ABXFACB0 unable to resolve logical external references. -- Prime DBD: *dbdname*. External DBD: *dbdname***

Explanation: While attempting to resolve logical external references in ACBLIB members, an error was encountered.

System Action: The step terminates.

User Response: Contact Support.

ABX0619E **Unknown SMF File Export Record type of *type*.**

Explanation: While executing program ABXPAUD to print File Export-specific SMF auditing records, an invalid record type was encountered.

System Action: The step terminates.

User Response: Ensure that the correct data set is being used as input to ABXPAUD.

ABX0620E **User I/O exit name not supplied.**

Explanation: A request to invoke a User Exit was made by either having the UXITACT=YES parameter specified in the Global Options block, or the EXITACTIVE(Y) control statement in the ABXUTIL control statement input stream. Neither the Global Options block nor the control statement stream supplied the name of the User Exit program to be executed.

System Action: The step terminates.

User Response: Modify the Global Options block or the ABXUTIL control statement input stream to include the User Exit program name.

ABX0621E **User I/O Exit *exit_name* encountered a fatal error.**

Explanation: While executing the User I/O exit named, a fatal error was encountered. This is indicated by the User I/O exit returning a value of 12 or 16 in the UIOXRC field of the ABXIOXA interface block.

System Action: The step terminates.

User Response: Correct the error that caused the User I/O exit to return a 12 or 16 in the UIOXRC field.

ABX0622I **User I/O Exit *exit_name* will be invoked for this execution.**

Explanation: This message indicates the name of the User I/O exit that will be invoked for this step.

System Action: Processing continues.

User Response: None

ABX0701E **Return code from Storage Obtain for named control block was *return_code*. Reason code *reason_code*.**

Explanation: The request for storage failed.

System Action: Processing terminates.

User Response: Increase the region size.

ABX0702E **ABEND code, Reason code *reason_code* in local routine *name*, Offset *offset*.**

Explanation: The named routine ABENDed at the described location for the reason listed.

System Action: Processing terminates.

User Response: Determine and correct the cause of the ABEND.

ABX0703E **Return code from initialization routines was *return_code*. Address space terminating.**

Explanation: The initialization process failed due to the return code listed.

System Action: Processing terminates.

User Response: Review any other issued messages and take appropriate action.

ABX0704E **Non-zero return code from MVS Load for module *module_name*. Return code *return_code*. Reason code *reason_code*.**

Explanation: An attempt to LOAD a module failed.

System Action: Processing terminates.

User Response: Correct the error.

ABX0705E **Control block pointer from *name1* to *name2* is invalid. Value *pointer*. Routine name *name3*.**

Explanation: An internal error occurred, causing an error in a control block chain.

System Action: Processing terminates.

User Response: Contact support.

ABX0706E **DDname *name* not present. Cannot continue.**

Explanation: A required DDname is not present in the JCL.

System Action: Processing terminates.

User Response: Add the missing DD statement to the JCL.

ABX0707E **Return code from OPEN of DD *name* was *return_code*.**

Explanation: An OPEN request for the named DD failed.

System Action: Processing terminates.

User Response: Determine the cause of the OPEN failure and correct the error.

ABX0708E **SSCT chain invalid. Cannot continue.**

Explanation: The chain of system SSCT pointers was found to be invalid.

System Action: Processing terminates.

User Response: Contact support.

ABX0709E **Return code from DEQ for *qname* was *return_code*.**

Explanation: An attempt to perform a dequeue failed.

System Action: Processing terminates.

User Response: Contact support.

ABX0710E **Control block ID not valid. Correct value = *value1*, value= *value2*, Routine name= *name*.**

Explanation: An internal error occurred causing an error in a control block chain.

System Action: Processing terminates.

User Response: Contact support.

ABX0711E **Non-zero return code from MVS function *name1*. Return code *return_code*, Reason code *reason_code*, Routine name= *name2*.**

Explanation: The MVS function named failed for the reason described.

System Action: Processing terminates.

User Response: Determine the cause of the failure.

ABX0712E **ABEND during MVS function call *name1*. ABEND code *abend*, Reason code *reason_code*, Routine name= *name2*.**

Explanation: The MVS function named ABENDed for the reason described.

System Action: Processing terminates.

User Response: Determine the cause of the failure.

ABX0713E **Non-zero return code from Dynamic Allocation of Internal Reader. Return code: *return_code*.**

Explanation: An attempt to dynamically allocate the internal reader failed.

System Action: Processing terminates.

User Response: Contact support.

ABX0714E **Dynamic allocation error code *return_code*. Info=*reason_code*.**

Explanation: A dynamic allocation attempt failed.

System Action: Processing terminates.

User Response: Review other informational and error messages issued in association with this message and take appropriate action.

ABX0715E ***text*.**

Explanation: This message is issued in association with other error messages. It contains text germane to the error.

System Action: None.

User Response: None.

ABX0716E **Non-zero return code from Dynamic Deallocation of the Internal Reader. Return code *return_code*.**

Explanation: An attempt to deallocate the Internal Reader failed.

System Action: Processing terminates.

User Response: Contact Support.

ABX0717E **Dynamic Deallocation error code *return_code*, Info=*reason_code*.**

Explanation: A deallocation attempt failed.

System Action: Processing terminates.

User Response: Review other informational and error messages issued in association with this message and take appropriate action.

ABX0718E **Logic error. Wrong record type passed to routine. Routine=*name1*. Record type *type1*, should be *type2*.**

Explanation: The routine named received a record type that was not expected.

System Action: Processing terminates.

User Response: Contact Support.

ABX0719E **Control file error. Record type =*type*. Sequence number= *number*, Routine=*name*.**

Explanation: The routine named received a control record that was either invalid or out of sequence.

System Action: Processing terminates.

User Response: Ensure the control file was sorted correctly.

ABX0720E **Possible control file error. JES did not return information for *name*. Record sequence number= *number*.**

Explanation: When querying JES to determine the status of the named job, the job was not found.

System Action: Processing terminates.

User Response: Contact Support.

ABX0721E **Invalid value passed on parm. Value=*text*.**

Explanation: An invalid parameter was passed to program ABXD2LEX.

System Action: Processing terminates.

User Response: Contact Support.

ABX0723E **Logic error. Table processing routine called for table record with status indicating complete. Record sequence number=*number*.**

Explanation: An internal error has occurred when attempting to schedule a LOAD of a DB2 table.

System Action: Processing terminates.

User Response: Contact Support.

ABX0724E **Control file error. Table record exists, but no owner/creator record found. Control number= *number*.**

Explanation: An internal error has occurred when attempting to schedule a LOAD of a DB2 table.

System Action: Processing terminates.

User Response: Contact Support.

ABX0725E **Control file error. Table name=*name*.**

Explanation: The message is issued in association with other messages indicating a specific control file error.

System Action: Processing terminates.

User Response: Contact Support.

ABX0726E **Error executing SQL Select. SQLCODE=*sqlcode* SQLSTATE= *sqlstate* Plan name=*name1* Routine name=*name2***

Explanation: An SQL Select request failed.

System Action: Processing terminates.

User Response: Determine the reason for the SQL error.

ABX0727E **Table not found in any SYSIBM.SYSTABLES row. Table name=*name*.**

Explanation: The named table does not exist in SYSTABLES.

System Action: Processing terminates.

User Response: Add the table name to SYSTABLES.

ABX0728E **Error returned from PUT to Internal Reader. RPL Feedback=*code*. Record=*text*.**

Explanation: An attempt to write a record to the Internal Reader failed.

System Action: Processing terminates.

User Response: Determine the reason for failure by examining the RPL feedback code.

ABX0729E **Logic error. Job execution tracking queue has no free elements, but job*jobname* was submitted.**

Explanation: The process used to submit DB2 LOAD jobs has run off the end of a chain.

System Action: Processing terminates.

User Response: Contact Support.

ABX0730E **Non-zero completion code for *name*. Job name =*jobname*.**

Explanation: A job submitted by ABXD2LEX resulted in a non-zero return code.

System Action: Processing terminates.

User Response: Examine messages issued by the job described.

ABX0731E **Connect to DB2 subsystem *name* failed. Return code: *return_code*, Reason code: *reason_code*.**

Explanation: A request to connect to the named DB2 subsystem failed.

System Action: Processing terminates.

User Response: Examine return and reason codes and take appropriate action.

ABX0732E **OPEN for plan *plan_name* failed. Return code: *return_code*, Reason code: *reason_code*.**

Explanation: A request to OPEN the named DB2 plan failed.

System Action: Processing terminates.

User Response: Examine return and reason codes and take appropriate action.

ABX0733E **CLOSE for plan *plan_name* failed. Return code: *return_code*, Reason code: *reason_code*.**

Explanation: A request to CLOSE the named DB2 plan failed.

System Action: Processing terminates.

User Response: Examine return and reason codes and take appropriate action.

ABX0734E **Disconnect from DB2 subsystem *name* failed. Return code: *return_code*, Reason code: *reason_code*.**

Explanation: A request to disconnect from the named DB2 subsystem failed.

System Action: Processing terminates.

User Response: Examine return and reason codes and take appropriate action.

ABX0734E **Maximum number of records exceeded *ctr* -- Record Type *type***

Explanation: While running the DB2 Load Executor, the capacity of an internal table has been exceeded.

System Action: The step terminates.

User Response: Contact Support.

ABX0760I ***type* JOB *name (jesid)*, for Table *table_name* submitted to the Internal Reader.**

Explanation: The named job was submitted to the internal reader to process data for the DB2 table described.

System Action: Processing continues.

User Response: None.

ABX0761I ***type* JOB *name (jesid)* for Table *table_name* completed successfully.**

Explanation: The named job that processed data for the DB2 table described completed.

System Action: Processing continues.

User Response: None.

ABX0762I **Connect to DB2 system *name* completed successfully.**

Explanation: A connect request to the named DB2 subsystem completed successfully.

System Action: Processing continues.

User Response: None.

ABX0763I **OPEN for plan *plan_name* completed successfully.**

Explanation: An attempt to OPEN the named DB2 plan completed successfully.

System Action: Processing continues.

User Response: None.

ABX0764I **CLOSE for plan *plan_name* completed successfully.**

Explanation: An attempt to CLOSE the named DB2 plan completed successfully.

System Action: Processing continues.

User Response: None.

ABX0765I **Disconnect from DB2 subsystem *subsystem_name* completed successfully.**

Explanation: A disconnect request to the named DB2 subsystem completed successfully.

System Action: Processing continues.

User Response: None.

ABX0766I **No input records for Table *table_name*. SORT and LOAD jobs are being bypassed.**

Explanation: As no input records were present for the named table, the JCL used to SORT and LOAD that table's data will not be submitted.

System Action: Processing continues.

User Response: None.

ABX0767E **Required record type *type* missing from DD ABXCNTL. Record Sequence: *number*, Subroutine: *name***

Explanation: A required record was missing from the input supplied by the ABXCNTL DD statement.

System Action: Processing terminates.

User Response: Contact Support.

ABX0768I **Sort not required for Table *table_name*.**

Explanation: It was determined that a sort was not required for the input for the DB2 table named.

System Action: Processing continues.

User Response: None.

ABX0769E ***msgtext***

Explanation: While running the DB2 Load Executor, an error occurred and the message issued reflects DSNTIAR processing.

System Action: The step terminates.

User Response: Review this and other messages to determine the problem and corrective action.

ABX0800E **Premature end of subtask DFSRRC00 in *program_name*.**

Explanation: DFSRRC00 was attached as a subtask of *program_name*. DFSRRC00 ended prematurely

System Action: The step terminates.

User Response: Review the output to determine what caused the event to occur.

ABX0801E **ABEND occurred in subtask DFSRRC00 .**

Explanation: An ABEND has occurred in DFSRRC00 or a program it has called.

System Action: The step terminates.

User Response: Review the output to determine what caused the event to occur.

ABX0802E **NAME/TOKEN service*service_name* failure--Return Code: *return_code*.**

Explanation: DFSRRC00 was attached as a subtask of *program_name*. DFSRRC00 ended prematurely

System Action: The step terminates.

User Response: Review the output to determine what caused the event to occur.

ABX0803E **ATTACH of DFSRRC00 failed -- Return Code: *return_code*.**

Explanation: The ATTACH to DFSRRC00 failed.

System Action: The step terminates.

User Response: Contact support.

ABX0804I **IMS input processing started using *IMS_regionid* services or - DBD *dbd_name* using PSB *psb_name***

Explanation: Informational message shows the IMS region, DBD name and PSB name used for input processing.

System Action: None

User Response: None required.

ABX0805I **IMS output processing started using *IMS_regionid* services or - DBD *dbd_name* using PSB *psb_name***

Explanation: Informational message shows the IMS region, DBD name and PSB name used for output processing.

System Action: None

User Response: None required.

ABX0806I **Input processing completed -- Total segments read: *count***

Explanation: Informational message shows the number of segments read on input.

System Action: None

User Response: None required.

ABX0807I **Output processing completed -- Total segments written: *count***

Explanation: Informational message shows the number of segments written on output.

System Action: None

User Response: None required.

ABX0809E **DB segment *name* not found in SEGFMT table.**

Explanation: The named segment was not found in the segment format table.

System Action: The step terminates.

User Response: Contact support.

ABX0810I **Duplicate Root segments replaced:*count***

Explanation: Informational message shows the number of root segments that were replaced in the output database.

System Action: None

User Response: None required.

ABX0811E **During Duplicate processing an unexpected Status Code was returned for either the GHU or DLET call -- Status Code=*status-code***

Explanation: During Duplicate processing, IBM File Export received a nonrecoverable status code for either the GHU or DLET on the root segment.

System Action: The step terminates.

User Response: Ascertain the cause of the status code.

ABX0812E **ACBGEN failed for PSB *psb_name*-- Return code: *return _code***

Explanation: While using the Dynamic PSB feature, an attempt to generate an ACB member for the named PSB has failed.

System Action: The step terminates.

User Response: Examine any other messages issued by the internal ACBGEN process and take corrective action.

ABX0813E **All possible dynamic PSB names have been used -- Dataset:*dataset_name*-- Name range: *name1* -- *name2***

Explanation: While using the Dynamic PSB feature, it was determined that the DOPT ACBLIB dataset already contained all possible generated PSB names.

System Action: The step terminates.

User Response: Wait until other concurrent users complete their runs and rerun your job. If this is a reoccurring problem, increase the possible number of PSB names that can be generated by increasing the value of the DYNPSFX option in the ABXOIMS0 control block or include the increased value as a DPSBSFX() control statement.

ABX0814E **Unable to find *type*Options block *block_name***

Explanation: During initialization, an attempt was made to find the Options block described in the STEPLIB concatenation of data sets.

System Action: The process terminates.

User Response: Ensure that the options block required was generated following the instructions in the IBM File Export for z/OS: *Reference Manual.*

ABX0815E **Unable to find the entry for *type* subsystem *ssid* in Options block *block_name***

Explanation: An attempt was made to find a reference to the subsystem name indicated in the message in the named options block. The attempt failed.

System Action: The process terminates.

User Response: Ensure that the options block required was generated with the subsystem name in question following the instructions in the IBM File Export for z/OSIBM File Export for z/OS: *Reference Manual.*

ABX0816E **STOW failed for member *member_name*-- Data set name: *dsn* -- Return code: *return_code* Reason code: *reason_code***

Explanation: While using the Dynamic PSB feature, an attempt was made to add the generated PSB member to the data set in question. The attempt failed.

System Action: The process terminates.

User Response: Determine the problem by examining the return and reason codes for the STOW macro in the *IBM Macro Instructions for Data Sets* manual.

ABX0817E **Unable to reposition on the input PCB after a SYNC call -- Status Code=*status_code***

Explanation: After a SYNC call, IBM File Export must reposition to the root segment that it is processing. During the GU process to reposition, a status code was received, preventing the repositioning.

System Action: The step terminates.

User Response: Determine the cause of the status code.

ABX0818E **Permission to use the Dynamic PSB function to *type*databases is denied.**

Explanation: During initialization, the security facility was queried to determine whether the user had permission to use the dynamic PSB function. The user did not.

System Action: The process terminates.

User Response: Review the section "Customizing IMS for use with IBM File Export" in the IBM File Export for z/OS: *Reference Manual.*

ABX0819E **Unexpected Status Code returned from IMS -- Status Code=*status_code***

Explanation: An unexpected status code was returned from IMS, preventing further processing.

System Action: The step terminates.

User Response: Determine the cause of the error.

ABX0820I **Input processing completed for DBD *dbdname*. Total segments read: *counter***

Explanation: The message indicates that processing has completed for the named database.

System Action: Processing continues.

User Response: None

ABX0821E **Duplicate segment. Processing Terminated.**

Explanation: While inserting segments to an IMS database, it was determined that a segment to be inserted already existed in the database.

System Action: The process terminates.

User Response: Validate the input data to ensure that duplicate segments are not contained in the input stream. Use the DLTDUPS(Y) keyword in the control statement stream if any existing segments should be deleted and replaced with the segments from the input stream.

ABX0822I ***pgmname*. invoked using PARM='*parm-list*'**

Explanation: This is an informational message indicating the name of the IMS region controller that is being executed to access IMS databases and the parameter list that was used.

System Action: Processing continues.

User Response: None

ABX1001I **Collect named *collect-name* defined in statement *statement-number* is never referenced**

Explanation: A COLLECT, that is named in the message, was defined but never referenced.

User Response: None required.

ABX1002I **Field named f*ield-name* defined in statement *statement-number* is never referenced**

Explanation: A FIELD, that was named in the message, was defined but never referenced.

User Response: None required.

ABX1003I **Literal named *literal-name* defined in statement *statement-number* is never referenced**

Explanation: A NAMED LITERAL, that is named in the message, was defined but never referenced.

User Response: None required.

ABX1004I **Start time (hhmmss) = *start-time***

Explanation: This message specifies the start time in hour-hour minute-minute and second-second format.

User Response: None required.

ABX1005W **Edit only was requested, so there will be no execution**

Explanation: The run was negated by user request. Return code 4 is produced.

User Response: If the execution is desired, remove the EDITONLY parameter from the OPTIONS statement and resubmit.

ABX1006E **In LOAD MODE there were no INSERT(s) found**

Explanation: Load mode was specified, but there were no insert directives specified. Return code 16 is produced.

User Response: Add an insert directive.

ABX1007E **Had run time errors**

Explanation: Editing was successful, but errors occurred at runtime. The specific errors are detailed in other error messages preceding this message.

User Response: Correct the cause of the runtime errors preceding this message.

ABX1008I **Had no run time errors**

Explanation: Editing was successful, no runtime errors were encountered.

User Response: None required.

ABX1009I **End time (hhmmss) = *end-time***

Explanation: This message specifies the end time in hour-hour minute-minute and second-second format.

User Response: None required.

ABX1010I **CPU time used = *cpu-time* seconds**

Explanation: The approximate CPU time used during execution is displayed in seconds.

User Response: None required.

ABX1011E **No memory - quitting processing**

Explanation: The processing ran out of memory.

User Response: Ensure that the region size is sufficient or break the run into parts. Each item in a relationship (SOURCE or CONTENT or a parent-child) and each UNIQUE request requires about twelve bytes in addition to their actual content size.

ABX1012I **Left Paren *parenthesis*: used = one or zero**

Explanation: This message is used for debugging.

User Response: None required

ABX1013I **Right Paren *parenthesis*: used = one or zero**

Explanation: This message is used for debugging.

User Response: None required

ABX1014I **word = *parsed-word* : used = *zero or one***

Explanation: This message is used for debugging.

User Response: None required

ABX1015E **Bad options parameter syntax - does not start with '('**

Explanation: The format of the OPTIONS statement starts with 'OPTIONS('.

User Response: Ensure that the OPTIONS declarative starts with ′(′

ABX1016E **You can only set the run type in the first statement**

Explanation: The run type (such as: DLI, SEQ, DB2, VSAM) can only be set in the first OPTIONS statement and this statement must be the first numbered statement in the input stream.

User Response: Remove run-type declarations from all OPTIONS statements except the first.

ABX1017E **You can only set the run type in the first statement**

Explanation: The run type (such as: DLI, SEQ, DB2, VSAM) can only be set in the first OPTIONS statement and this statement must be the first numbered statement in the input stream.

User Response: Remove run type declarations from all OPTIONS statements except the first.

ABX1018E Could not initialize the I/O routines

Explanation: A call was made (in Sequential or VSAM or DLI mode) to the I-O routines, that was not successful. The specific errors are detailed in other error messages preceding this message.

User Response: Correct the cause of the error messages preceding this message.

ABX1019E You can only set LOAD type in the first statement

Explanation: The LOAD type can only be set in the first OPTIONS statement and this statement must be the first numbered statement in the input stream.

User Response: Remove LOAD type declarations form all OPTIONS statements except the first.

ABX1020E Fixed is specified as FIXED (*nn*)

Explanation: In processing the OPTIONS declarative, FIXED was found with an incorrect format.

User Response: Correct the invalid FIXED declarative. The correct format is: FIXED(*nnn*) where *nnn* is an integer. FIXED only applies to Sequential processing.

ABX1021E Fixed is specified as FIXED (*nn*)

Explanation: In processing the OPTIONS declarative, FIXED was found with an incorrect format.

User Response: Correct the invalid FIXED declarative. The correct format is: FIXED(*nnn*) where *nnn* is an integer. FIXED only applies to Sequential processing.

ABX1022E Could not load the I/O routines

Explanation: In processing the OPTIONS declarative, FIXED was found with an incorrect format.

User Response: Correct the invalid FIXED declarative. The correct format is: FIXED(*nnn*) where *nnn* is an integer. FIXED only applies to Sequential processing.

ABX1023E QUALIFIER is specified as QUALIFIER (*xxx*)

Explanation: In processing the OPTIONS declarative, the word QUALIFIER was found with the wrong format. QUALIFIER must be followed by name: QUALIFIER(*name*)

User Response: Correct the QUALIFIER sub-declarative using the correct syntax: QUALIFIER(*name*)

ABX1024I Ignoring GROUP when not in DLI mode

Explanation: In processing the OPTIONS declarative, GROUP was found. GROUP is only valid when the specified run type is DLI. The GROUP declaration will be ignored.

User Response: If you require DLI processing, change the run-type to DLI, otherwise, no action is required.

ABX1025E Last SOURCEDBD had GROUP attribute but no select marked FINAL.

Explanation: In processing the OPTIONS declarative, GROUP was found, but no SELECT marked FINAL was present in the stream. When using GROUP, the final SELECT statement must be marked with the FINAL declaration.

User Response: Mark your final SELECT statement with the FINAL declaration to indicate that the GROUP is accepted or rejected.

ABX1026E Bad options parameter (*invalid-word*)

Explanation: In processing the OPTIONS declarative, a word (displayed above) was found, but is not a syntactically allowed word.

User Response: Correct the invalid word. Perhaps it is mis-spelled.

ABX1027E Bad options parameter syntax - does not end with ')'

Explanation: In processing the OPTIONS declarative, the last thing in the text for this statement must be a right parenthesis.

User Response: In processing the OPTIONS declarative, The last thing in the text for this statement must be a right parenthesis.

ABX1028I Ignoring fixed in DLI mode

Explanation: In processing the OPTIONS declarative, FIXED was found, but fixed does not apply to DLI mode of processing.

User Response: None required, unless the processing mode of DLI is incorrect.

ABX1029E Can only specify a source database name when processing DLI

Explanation: In processing the OPTIONS declarative, DLI was not declared, but a mandated DLI declarative (SOURCEDBD) was found.

User Response: Delete the SOURCEDBD statement, or declare DLI in the OPTIONS statement.

ABX1030E **Bad database name syntax - does not start with '('**

Explanation: In processing the SOURCEDBD declarative, the second 'word' should be a left parenthesis.

User Response: Correct the invalid declarative.

ABX1031E **Can only specify source database name once**

Explanation: Only one SOURCEDBD statement may appear in a run.

User Response: Delete the excessive declarative.

ABX1032E **The database name cannot exceed eight characters**

Explanation: The DBD name is only valid to eight characters.

User Response: Correct the invalid declarative.

ABX1033E **Bad database name syntax - does not end with ')'**

Explanation: the database name in the SOURCEDBD must be surrounded by parentheses.

User Response: Correct the invalid declarative.

ABX1034E **Expected DD name**

Explanation: When parsing the SOUCREDBD, that ddname was not found.

User Response: Supply the ddname for the SOURCE DBD.

ABX1036E **Can only specify a target database name when processing DLI**

Explanation: The OPTION (… DLI …) was not specified, yet a TARGETDBD declarative was encountered.

User Response: Either specify the DLI procession mode in the OPTION parameters, or remove the TARGETDBD statement.

ABX1037E **Bad database name syntax - does not start with '('**

Explanation: In specifying the TARGETDBD a left parenthesis is expected after the word TARGETDBD.

User Response: Correct the invalid declarative.

ABX1038E **Can only specify target database name once**

Explanation: More than one occurrence of a TARGETDBD statement was found.

User Response: Delete one of the TARGETDBD statements.

ABX1039E **The database name cannot exceed eight characters**

Explanation: The supplied target database name is excessive in length.

User Response: Correct the invalid declarative.

ABX1040E **Bad database name parameter syntax - does not end with ')'**

Explanation: The database name is contained in parentheses.

User Response: Correct the invalid declarative.

ABX1041E **Expected DD name**

Explanation: After the database name, in a TARGETDBD statement, must be the DDNAME pointing to the DBD file.

User Response: Correct the statement, adding the DDNAME (and the DD card id needed).

ABX1042E **Expected 'DD'**

Explanation: After the TARGETDBD name, the 'word' DD is required. This specifies the JCL DD that points to the library.

User Response: Correct the TARGETDBD statement.

ABX1043E **Name exceeds maximum of 32 characters**

Explanation: Any name (excluding table and owner in DB2 mode) is limited to 32 characters. Column names in DB2 mode are limited to 30 characters due to SQLDA restrictions.

User Response: Shorten the field/literal/collect or column name.

ABX1044E **A name cannot start with a number**

Explanation: Names (fields, collects, named literals, variables) must start with a alphabetic character - not a number.

User Response: Correct the appropriate name.

ABX1045E **Name has a disallowed character '*the disallowed character*'**

Explanation: A name (field, literal, collect) contains a character that is not permitted. Names can use alpha-numerics and the underscore and the national dependent characters: @,#,$. But the first character must be alphabetic.

User Response: Correct the name.

ABX1046E **Word 'field' must be followed by a left parenthesis**

Explanation: In parsing a FIELD statement, the first text character following 'FIELD' should be the left parenthesis opening the field-name declarative.

User Response: Correct the statement.

ABX1047E **Expected a field name**

Explanation: The format of a field statement is: FIELD (*field-name*) … The statement ended before the field name was supplied.

User Response: Correct the statement.

ABX1048E **Expected a right parenthesis**

Explanation: In a FIELD statement, the syntax is: FIELD (*field-name*). The statement ended before the right parenthesis.

User Response: Correct the FIELD or VARIABLE statement.

ABX1049E **Syntax requires a right parenthesis after the field name**

Explanation: In a field statement, the format is FIELD (*field-name*)... The next word after the *field-name* or variable-name is not a right parenthesis.

User Response: Correct the FIELD statement.

ABX1050E **Expected an equal sign**

Explanation: In a FIELD or VARIABLE statement, the format is: FIELD (*field-name*) =... The equal mark following the right parenthesis after the *field-name* or *variable-name* was not found.

User Response: Correct the statement.

ABX1051E **Syntax requires an equal sign after field name**

Explanation: In parsing a FIELD or VARIABLE statement, the statement is missing an equal after the field/variable-name definition.

User Response: Correct the statement.

ABX1052E **Expected a left parenthesis starting the position and length**

Explanation: In processing a FIELD or VARIABLE statement, the text following the name defines the attributes (CHARACTER, PACKED, INTEGER, etc.). This text starts with a left parenthesis (following the equal sign). This left parenthesis was not found.

User Response: Correct the statement.

ABX1053E **The attributes in a field start with a left parenthesis**

Explanation: In a FIELD or VARIABLE, the attribute definition (length, position, data type). This left parenthesis was not found.

User Response: Correct the statement.

ABX1054E **Expected the position field**

Explanation: In processing a FIELD or VARIABLE statement, the data attributes commence with the position or offset (see OPTIONS) value. This was not found.

User Response: Correct the statement.

ABX1055E **Expected minus sign in length after word END**

Explanation: In processing a FIELD statement, one way to define the position is with END - integer. The required minus sign after the word END is missing.

User Response: Correct the statement.

ABX1056E **Expected a number after 'END -'**

Explanation: In processing a FIELD statement, the position of the data is defined as END - followed by an integer. This integer is missing.

User Response: Correct the statement.

ABX1057E **Expected a minus sign after word END**

Explanation: In processing a FIELD statement whose position is defined with the END - integer method, the minus sign is missing. The 'END' was found'.

User Response: Correct the statement.

ABX1058E **Expected text in length after words 'END -'**

Explanation: In parsing a FIELD statement, the statement is not complete. There was no text where the integer following 'END -' belongs.

User Response: Correct the statement.

ABX1059E **Expected a number in the position field**

Explanation: In a FIELD statement, the position is not either 'END -' or a number (an integer). Neither of these were found.

User Response: Correct the statement,

ABX1060E **The position field is too long - max of 5 digits**

Explanation: In processing a FIELD statement, the position is excessive. The maximum supported record length is 32,767 bytes.

User Response: Correct the statement.

ABX1061E **The position can not be less than one**

Explanation: The specified position is zero or negative. The minimum position is zero, while the minimum offset is zero. See OPTIONS for setting offsets.

User Response: Correct the statement.

ABX1062E **Expected a comma**

Explanation: In processing a FIELD statement, the comma after the position was not found. This comma separates the position from the length designation.

User Response: Correct the statement.

ABX1064E **Expected a length field**

Explanation: In processing a FIELD statement the length indication was not found.

User Response: Correct the statement.

ABX1065E **The length field is too long - max of 5 digits**

Explanation: The field length is longer than 32,767 - the maximum record length.

User Response: Correct the statement.

ABX1066E **Expected a number in the length field**

Explanation: In processing a FIELD statement, the length indicated is not an integer.

User Response: Correct the statement.

ABX1067E **The length can not be less than one**

Explanation: In processing a FIELD statement, the length specified is not valid.

User Response: Correct the statement.

ABX1068E **An explicit length can not be greater than 256**

Explanation: For a FIELD or VARIABLE, a single item cannot exceed 256 bytes in length.

User Response: Correct the statement, or use two or more FIELDS or VARIABLES.

ABX1069E **For END - *nn* definition, the length (*specified_length*) cannot exceed the position (*specified_position*)**

Explanation: The FIELD is defined with the 'END - *nn*' style, and the length specified exceeds *nn*. Or the length extends beyond the record.

User Response: Correct the length specification.

ABX1070E **Expected a comma**

Explanation: In a FIELD statement, the position indictor must be followed with a comma.

User Response: Correct the FIELD statement.

ABX1072E **Expected field type**

Explanation: In a FIELD statement, the data type (packed, float, etc.) was expected, but not present.

User Response: Correct the statement.

ABX1073E **INT requires a length of 4**

Explanation: A data type of INT (integer) was specified for a FIELD, yet the length was not specified as 4.

User Response: Correct either the data type or the length as they are not compatible.

ABX1074E **SHORT requires a length of 2**

Explanation: In processing a FIELD, or a VARIABLE, the data type of SHORT is only comparable with a length of 2.

User Response: Correct either the data type or the length.

ABX1075E FLOAT requires a length of 4

Explanation: In a FIELD or a VARIABLE, the data type of FLOAT must be specified as a length of 4.

User Response: Correct either the length or the data type.

ABX1076E DOUBLE requires a length of 8

Explanation: In a FIELD or a VARIABLE, the data type of DOUBLE must be specified as a length of 8.

User Response: Correct either the length or the data type.

ABX1077E INTEGER requires a length of 4

Explanation: In a FIELD or a VARIABLE statement, the data type INTEGER is always a length of 4.

User Response: Correct either the data type or the length.

ABX1078E Unknown data type

Explanation: In specifying a FIELD or a VARIABLE, the data type is not a recognized word.

User Response: Correct the data type.

ABX1079E A type of *found-word* is invalid

Explanation: In specifying a FIELD or VARIABLE, the word specifying the data type is not a recognized word.

User Response: Correct the data type.

ABX1080E Length of '*' requires type HEX or CHAR

Explanation: In specifying a FIELD, the length specified for a field is '*' meaning the rest of the record. However only CHARACTER or HEX data types can have a length determined at run type.

User Response: Correct either the data type or the length.

ABX1081E Expected right parenthesis

Explanation: In defining either a FIELD or a VARIABLE, the data attributes (position, length, data type) ends with a right parenthesis - which was not found.

User Response: Correct the data attribute specification.

ABX1082E Expected right parenthesis after field attributes

Explanation: Same as ABX1081E above.

User Response: Correct data attribute specification.

ABX1083E Extraneous word after length fields

Explanation: In a FIELD, there was text length within the statement after the syntax was complete. That is after the right parenthesis ending the data attributes.

User Response: Remove the excessive text.

ABX1084E The name *field-name* or variable-name is already defined

Explanation: At this scope, the name chosen for this field (or variable) is already defined.

User Response: Choose a different name for this or the previous definition.

ABX1085E Table put failed with return code *nn*

Explanation: Unless this message is preceded with an 'out of memory' message, this is evidence of memory corruption and a program problem.

User Response: See if your maintenance is current. If so, please report a bug through your support channel.

ABX1086E The field *field-name* is already defined

Explanation: This is a redundant definition of the chosen field name at this scope.

User Response: Choose a new name for either this field or the previous field or capture of names literal using this same name at this scope.

ABX1087E Table put failed with return code *nn*

Explanation: Unless this message is preceded with an 'out of memory' message, there is evidence of memory corruption and a program problem.

User Response: Check to see if your maintenance is current. If so, please report a bug through your maintenance channels.

ABX1088E No memory for the select chaining

Explanation: Memory has been exhausted.

User Response: Increase the region size.

ABX1089E Segment *segment-name* not found

Explanation: The named segment is not defined in the DBD.

User Response: Either this is the wrong DBD, or the segment specified is in error. Correct the error.

ABX1090E **Type type-name not found**

Explanation: The referenced TYPE is not known.

User Response: Correct the statement referring to the TYPE, or rename the TYPE.

ABX1091E **LAST_SEGMENT specified with no active (previous) segment**

Explanation: A LAST SEGMENT statement was found without a preceding (valid) segment statement.

User Response: Either correct a preceding SEGMENT statement, or relocate this LAST SEGMENT statement.

ABX1092E **The type name is redundant (type-name) within the segment**

Explanation: Each TYPE (within a segment) must have a unique name.

User Response: Correct the TYPE name. Modify it so that it is unique.

ABX1093E **The type name of DEFAULT must be the last type in the type definitions at the same level. Any select(s) prior to the first type (in a SEGMENT) create the DEFAULT type**

Explanation: When using TYPEs, no additional TYPE is valid after an explicit type named DEFAULT (within each segment if in DLI mode).

User Response: If you want a definitive default, express it before any succeeding TYPE.

ABX1094E **Expected a left parenthesis after the word TYPE**

Explanation: In parsing the type statement, the first character after TYPE is a left parenthesis opening the type name declarative. This was not the case.

User Response: Correct the syntax.

ABX1095E **Expected a right parenthesis after the TYPE name**

Explanation: The name of the type is enclosed in parentheses. An opening and closing set of parentheses is required.

User Response: Correct the syntax.

ABX1096E **Type must have predicate(s) except for DEFAULT**

Explanation: To identify the case(s) constituting the TYPE, predicates must be present.

User Response: Either complete the type definition or remove it.

ABX1097E **Found no use for word *found-word***

Explanation: In parsing the statement, a word was found that had no identified use.

User Response: Correct the error by removing the extraneous word, or perhaps the spelling of the erroneous word.

ABX1098E **Expected a count**

Explanation: In processing a LIMIT or SKIP statement, the COUNT was not found.

User Response: Correct the statement.

ABX1099E **Excessive length on the count field - *character_count_found***

Explanation: The count is limited to approximately 2 billion.

User Response: Correct the count.

Chapter 3. Messages ABX1100--ABX1887

All messages generated by IBM File Export for z/OS have a severity code printed as the last character of the message ID. The severity codes are described in the following table:

Table 2. Error message severity codes

Severity Code	Description
I	Information only. No user action required.
W	Warning message. Results may not be as expected.
E	Error message. Some may be user-correctable, read the User Response to determine the course of action.
S	Severe error -- note the information in the message and contact your support personnel.

ABX1100E Invalid count value *found-value*

Explanation: In processing a LIMIT or SKIP statement, the position expected to contain the count had the displayed data --which is not a valid count.

User Response: Correct the count field.

ABX1101E A global value of SKIP is not supported in DLI/IMS mode

Explanation: In DLI processing, SKIP can only be tied to a specific SEGMENT or TYPE within a segment. Global SKIP is supported in VSAM or sequential processing.

User Response: Remove or move the SKIP statement.

ABX1102E Extraneous word(s) after the count

Explanation: In a SKIP, the text ends with the count. Word(s) were found after this field.

User Response: Correct the syntax.

ABX1103E Expected a count

Explanation: In processing a SKIP statement, the statement terminated before finding the count field.

User Response: Correct the syntax.

ABX1104E Excessive length on the count field -- *found-length*

Explanation: The maximum length for the count field is ten digits.

User Response: Correct the count field.

ABX1105E Invalid count value *found-value*

Explanation: The *found-value* occupies the position of the count in a LIMIT statement, and is not valid.

User Response: Correct the count field.

ABX1106E A global value of LIMIT is not supported in DLI/IMS mode

Explanation: In DLI mode, LIMIT must be applied to either a SEGMENT or a TYPE. A global limit is valid in VSAM or SEQUENTIAL mode.

User Response: Correct the position (in the input stream) of the LIMIT statement.

ABX1107E Extraneous word(s) after the count

Explanation: In processing a LIMIT statement, an unrecognized word followed that count field. Perhaps this is spelling.

User Response: Correct the statement.

ABX1108E Extraneous word(s) after EXCLUDE

Explanation: While processing an EXCLUDE statement, extra words were found after the valid text.

User Response: Correct the syntax.

ABX1109E **An EXCLUDE can only apply to a SEGMENT or TYPE**

Explanation: An EXCLUDE statement was found where it has no meaning.

User Response: Delete or reposition the EXCLUDE statement.

ABX1110E **Can't get 34k buffer - quitting**

Explanation: During initialization, a request for memory was not satisfied. This message is usually in addition to an out of memory message.

User Response: Provide for a larger region through JCL parameters.

ABX1111E **Could not open table of names in memory**

Explanation: If this is accompanied with an 'out of memory' message, the memory for the region should be increased. If not, there is probable memory corruption.

User Response: If out of memory, increase the region.

ABX1112E **Could not open table of field names in memory**

Explanation: This is a memory problem if accompanied with an 'out of memory' message. Else it is possibly a memory corruption issue.

User Response: Allow for more memory by increasing the region.

ABX1113E **Could not open table of literal names in memory**

Explanation: This is a memory problem if accompanied with an 'out of memory' message. Else it is possibly a memory corruption issue.

User Response: Allow for more memory by increasing the region.

ABX1114E **Could not open table of COLLECT names in memory**

Explanation: This is a memory problem if accompanied with an 'out of memory' message. Else it is possibly a memory corruption issue.

User Response: Allow for more memory by increasing the region.

ABX1115E **Can't open input text file ddname - quitting**

Explanation: A DD (data definition) card is missing.

User Response: Correct the run-time JCL and provide the referenced DD card.

ABX1116I **Ignoring this statement in LOAD mode**

Explanation: OPTIONS (... LOAD...) was specified. This type of statement is not applicable to LOAD mode processing.

User Response: Either ignore the message (it is harmless to do so if you intend LOAD mode) or remove the statement.

ABX1117E **Select object (*found-table-name*) not matched with segment name (*current-segment-name*)**

Explanation: In DLI processing mode, A SELECT was parsed that used a name other than the current segment's name as the 'table name'. Possibly this SELECT is in an incorrect position in the control text stream.

User Response: Correct the *found-table-name*to be the applicable segment, or relocate the SELECT to the correct SEGMENT area.

ABX1118E **In DLI mode cannot have a select without a SEGMENT**

Explanation: A select is always associated with a SEGMENT in DLI mode (or a TYPE within a SEGMENT).

User Response: Position the select to follow the associated segment (and type).

ABX1119E **In DLI mode you cannot have a type without a segment**

Explanation: When in DLI mode, A TYPE quantifies a sub-definition of a SEGMENT. All TYPE statements follow SEGMENT statements in DLI mode of processing.

User Response: Reposition the TYPE statement to its correct position in the control syntax stream.

ABX1120E **Cannot have a segment when not in DLI mode**

Explanation: SEGMENTS only apply to DLI mode of processing.

User Response: Remove the segment statement unless you need to specify OPTIONS(DLI,...).

ABX1121E **COLLECT must immediately follow a valid select or another valid collect or a THEN INSERT...**

Explanation: COLLECT captures data from a selected record (even a SELECT NOTHING record) where the predicates are true. As such it is true if THEN INSERT

follows the associated SELECT, with possible intervening THEN INSERT statements.

User Response: Correct the order of the control statements.

ABX1122E No appropriate prior select

Explanation: When trying to anchor a COLLECT statement, no prior (valid) SELECT was found.

User Response: If a prior SELECT was found to be invalid, correct it. Else position the COLLECT after the associated SELECT.

ABX1123E Chaining of collect failed

Explanation: This may be accompanied with an 'out of memory' message.

User Response: If preceded with an 'out of memory' message, increase the region. Else there has been a failed prior statement type, or lastly there has been memory corruption.

ABX1124I Ignoring this statement as INSERT(s) are only valid in load mode

Explanation: In LOAD mode, there is an INSERT statement, but in non-LOAD mode the correct syntax is THEN INSERT...

User Response: Be sure that you are in the correct mode, or modify the syntax.

ABX1125E Chaining of INSERT failed

Explanation: This is usually an 'out of memory' condition when associated with this type of message.

User Response: If out of memory, increase the region. Else a prior (necessary) SELECT did not parse, or memory is corrupted.

ABX1126E THEN INSERT... must immediately follow a select or another THEN INSERT...or a collect

Explanation: THEN insert is associated with a select that chooses records. As such it must be preceded by a SELECT, or a CAPTURE or another THEN INSERT.

User Response: Correct the order of the control text statements.

ABX1127E Last select failed to chain

Explanation: A statement dependent on a prior SELECT could not be chained because the prerequisite was not accepted.

User Response: Correct either the preceding statement(s), or position this dependent statement correctly in the control syntax stream.

ABX1128E Chaining of THEN INSERT failed

Explanation: The prerequisites (SELECT or COLLECT or another THEN INSERT) was not processed (successfully) in the input stream. Or there is no memory.

User Response: Increase the REGION, or correct the prerequisite statement(s).

ABX1129E Error - Unknown or invalid statement

Explanation: The first word in the statement is not an identified statement type.

User Response: Correct this statement, or possibly the prior statement was not terminated (a semi-colon) in the appropriate place.

ABX1130E ***found-word***

Explanation: This accompanies ABX1129E above and identifies the unrecognized word(s).

User Response: Correct this or the preceding statement (if the preceding statement is terminated incorrectly).

ABX1131E Invalid statement: ***statement-text***

Explanation: The parsing did not recognize a statement. Perhaps the first word is misspelled.

User Response: Correct the statement, or perhaps comment it with a '::' in columns one and two.

ABX1132E Invalid statement (not terminated): ***statement-text***

Explanation: The last statement in the syntax stream is not terminated.

User Response: Correct this statement.

ABX1133E Had editing error(s)

Explanation: This message accompanies other statement error(s) found while parsing the control syntax. The program will suppress execution.

User Response: Correct the errors previously cited before this message.

ABX1134I Editing had no errors

Explanation: The parsing routines found no syntactical errors.

User Response: No action is required.

ABX1135E **Must specify a database name if run type is DL/I**

Explanation: If OPTIONS (... DLI...) is specified, the source database must be specified.

User Response: Include a SOURCEDBD definition.

ABX1136E **Must specify a new database name if run type is DL/I and LOAD mode**

Explanation: When in DLI mode, with LOAD, the target DBD must be specified.

User Response: Include a TARGETDBD statement.

ABX1137E **For field *field-name* the net position resulted in the field's being positioned before the record**

Explanation: For a FIELD definition of END - *nnn*, the specified '*nnn*' exceeds the logical record length resulting in a negative position.

User Response: Correct this FIELD statement, or adjust the LRECL of the input file.

ABX1138E **In field(*field-name*) a length of * resulted in negative or zero length**

Explanation: When specifying a FIELD using the asterisk length (the rest of the record) the starting position is too far to the 'right' so that the produced length would be minus.

User Response: Correct this field definition, or adjust the input record LRECL.

ABX1139E **A field (*field-name*) position (starting-position) + length (*specified-length*) exceeds record length (*logical record length*)**

Explanation: In specifying a field, the data would extend beyond the defined record's length.

User Response: Correct this field definition, or the logical record length.

ABX1140E **Invalid character in Scramble routine, field = *field-name***

Explanation: In a scramble routine at run time, a character was found that did not match the data type for the defined field. As an example, non-packed data was found in a packed field. Execution stops.

User Response: Correct the data, the field definition, or suppress this particular scrambling.

ABX1141E **An integer must have a length of four - field *field-name***

Explanation: In defining a field with a data type of integer, the length is always 4.

User Response: Correct this statement.

ABX1142E **An SHORT must have a length of two - field *field-name***

Explanation: A SHORT integer always has a length of two.

User Response: Correct this field definition.

ABX1143E **At scramble, data is not packed**

Explanation: During execution data specified as packed was found to be invalid. This usually follows message ABX1140E.

User Response: Check the field definition. If it is correct, then scrambling here cannot be performed due to invalid data in the file. Perhaps a TYPE is needed.

ABX1144E **Invalid data type in Scramble routine field = *field-name***

Explanation: The scrambling routine found data that is not allowed by the type of data defined. The execution is terminated.

User Response: You may not be able to scramble this field if the field definition is correct.

ABX1145E **Length at hex unpacking is too small: net-length**

Explanation: During unpacking data, the source length became less than one. A short record can cause this problem.

User Response: Perhaps a TYPE should be specified to avoid this record being processed as requested.

ABX1146E **Invalid hex character found-*character***

Explanation: In specifying a HEX literal, a character other than zero through nine or a through f or A through F was found.

User Response: Correct this hexadecimal literal.

ABX1147E **Invalid hex character found-*character***

Explanation: In processing a hexadecimal literal, an invalid (non-hex) value was specified.

User Response: Correct this literal.

ABX1149E Excessive length of data at unpack call

Explanation: When attempting to unpack a value, the length was excessive - beyond the supported length of packed data.

User Response: Correct this field specification.

ABX1150E Length of 0 at unpack call is not valid

Explanation: When processing packed data, the resultant length is too short. This can happen on a short record specified as length '*'.

User Response: Correct this field, or use a type definition to correctly process this record type.

ABX1151E Invalid packed data

Explanation: A field was specified as packed, yet the data was not valid packed data.

User Response: Correct this field definition, or use a type definition for alternative processing.

ABX1152E Length cannot be less that 1 at pack

Explanation: At run time, the length of a packed field is too short. Possibly the record is too short.

User Response: Correct this field definition, or use a TYPE statement to invoke alternative processing.

ABX1153E Word 'literal' must be followed by a left parenthesis

Explanation: In parsing a named literal, the second field is the literal's name and this field begins with a left parenthesis.

User Response: Correct the LITERAL statement.

ABX1154E Expected a literal name

Explanation: In parsing a named literal, the area where the name of the literal is specified (after the first left parenthesis) is not present or has an invalid name.

User Response: Correct this literal definition.

ABX1155E Expected a right parenthesis

Explanation: In processing a literal, the name of the literal is de-marked with parentheses before and after the name.

User Response: Correct this literal definition.

ABX1156E Syntax requires a right parenthesis after the literal name

Explanation: In processing a named literal, the right parenthesis following the chosen name is missing.

User Response: Correct this statement.

ABX1157E Expected an equal sign or word 'value'

Explanation: In defining a literal, the text after the chosen name (and the closing parenthesis) must be an equal sign or the word 'value'.

User Response: Correct this literal definition.

ABX1158E Syntax requires 'value' or an equal sign after literal name

Explanation: The text following 'LITERAL (chosen-name)... ' must be either the equal sign or the word VALUE.

User Response: Correct this definition.

ABX1159E Expected a literal after 'value'

Explanation: In defining a literal, the text after the equal sign or the word 'value' should define the content. The found text does not define a valid literal.

User Response: Correct the literal definition.

ABX1161E Only a left parenthesis may follow the literal

Explanation: In defining a literal, the area after defining the content defines the explicit data attributes - length and data type. This is not the case in the found text.

User Response: Correct this literal definition.

ABX1162E Expected a length field

Explanation: In defining a literal, the text following the definition of the values of the literal defines the attributes - length and data type. This length field is missing.

User Response: Correct the literal definition.

ABX1163E Length field is not numeric (*found-text*)

Explanation: In defining a literal, the length specification of the constant is not valid.

User Response: Correct the literal definition.

ABX1164E **A literal length of *found-text* is not valid**

Explanation: In defining a literal, the length as specified is not valid text.

User Response: Correct this literal definition.

ABX1166E **Expected right parenthesis or comma after the explicit length**

Explanation: In defining a literal, the explicit length must be followed by either the data type (if ambiguous) or a closing parenthesis.

User Response: Correct the literal definition.

ABX1168E **Expected a data type after the comma**

Explanation: In defining a literal, the explicit length was specified and the required following comma. However the next word is not recognized as a valid data type such as INT or SHORT or CHAR.

User Response: Correct this literal definition.

ABX1169E **INT requires a length of 4**

Explanation: When specifying a literal with a data type of INT, only the length of four is supported.

User Response: Correct this literal definition.

ABX1170E **SHORT requires a length of 2**

Explanation: When specifying a literal with a data type od SHORT, the only supported data length is two.

User Response: Correct this literal definition.

ABX1171E **FLOAT requires a length of 4**

Explanation: When specifying a literal with a data type of FLOAT, the length must be 4.

User Response: Correct this literal definition.

ABX1172E **DOUBLE requires a length of 8**

Explanation: When specifying a literal of data type DOUBLE, only a length of eight is valid.

User Response: Correct this literal definition.

ABX1173E **INTEGER requires a length of 4**

Explanation: When defining an integer literal, the only allowed length is four.

User Response: Correct this definition.

ABX1174E **Unknown data type**

Explanation: When defining a literal or a variable, the data type expressed in the declarative is not known.

User Response: Correct this statement.

ABX1175E **A type of *found-word* is invalid**

Explanation: In processing a literal or a variable, the expressed data type is not recognized.

User Response: Correct this statement.

ABX1176E **Missing final right parenthesis**

Explanation: In defining a literal or a variable, the last right parenthesis is missing.

User Response: Correct this statement.

ABX1177E **Final character is not a right parenthesis**

Explanation: The ultimate 'word' in the literal or variable statement is not the expected right parenthesis.

User Response: Correct this statement.

ABX1178E **Extraneous word(s) after definition - *found-word***

Explanation: The declaration of the variable to literal has text beyond the expected syntax.

User Response: Correct this statement.

ABX1179I **Char: *found-character***

Explanation: This is a debugging message and is of little value to the user. It is emitted in 'debug' mode.

User Response: Ignore this message, but send the produces print to your support team if OPTIONS(... DEBUG...) was requested of you.

ABX1180I **Int: *an integer value***

Explanation: This is a debugging message and is of little value to the user. It is emitted in 'debug' mode.

User Response: Ignore this message, but send the produces print to your support team if OPTIONS(... DEBUG...) was requested of you.

ABX1181I **Small int (SHORT): *a short integer value***

Explanation: This is a debugging message and is of little value to the user. It is emitted in 'debug' mode.

User Response: Ignore this message, but send the produces print to your support team if OPTIONS(... DEBUG...) was requested of you.

ABX1182I **Float:** ***a real number***

Explanation: This is a debugging message and is of little value to the user. It is emitted in 'debug' mode.

User Response: Ignore this message, but send the produces print to your support team if OPTIONS(... DEBUG...) was requested of you.

ABX1183I **Double:** ***a long real number***

Explanation: This is a debugging message and is of little value to the user. It is emitted in 'debug' mode.

User Response: Ignore this message, but send the produces print to your support team if OPTIONS(... DEBUG...) was requested of you.

ABX1184I **Packed:** ***a packed value***

Explanation: This is a debugging message and is of little value to the user. It is emitted in 'debug' mode.

User Response: Ignore this message, but send the produces print to your support team if OPTIONS(... DEBUG...) was requested of you.

ABX1185I **Hex:** ***a packed value***

Explanation: This is a debugging message and is of little value to the user. It is emitted in 'debug' mode.

User Response: Ignore this message, but send the produces print to your support team if OPTIONS(... DEBUG...) was requested of you.

ABX1186E **Bad literal type**

Explanation: In parsing a literal or a variable, an unrecognized literal type was encountered.

User Response: Correct this statement.

ABX1187E **The name *found-literal-or-variable-name* is already defined**

Explanation: This name is redundantly used. It must be unique within its scope.

User Response: Chose a new name for this literal or variable. Or for the declaration for which it collides.

ABX1188E **Table put failed with return code *nn***

Explanation: This may be accompanied with an 'out of memory' message. If memory is exhausted, increase the region size. Else possible memory corruption has occurred.

User Response: If not a memory issue, report this incident.

ABX1189E **The field *field-name* is already defined**

Explanation: In defining a field, the chosen name is redundant within its scope.

User Response: Name this field something else, or rename the field with which it collides.

ABX1190E **Table put failed with return code *nnn***

Explanation: An error has occurred in the table handler. If this message is preceded by an 'out of memory' message, the problem is low memory.

User Response: If there is a memory shortage, increase the region size. If not accompanied with an 'out of memory' message, memory corruption has probably occurred.

ABX1191E **At literal capture, value is not marked as a literal**

Explanation: In processing the text where a literal value should appear, the actual text is not a literal - numeric or character data.

User Response: Correct this statement.

ABX1192E **You can only initialize a CHAR or HEX field with a HEX or CHAR literal**

Explanation: In defining the literal, the initialization text is not compatible with the declared type of literal.

User Response: Correct this statement.

ABX1193E **Hex literal has invalid data (*found-text*)**

Explanation: Only zero through nine and 'a' through 'f' can initialize a hew literal.

User Response: Correct this text.

ABX1194E **Literal size causes loss of HEX data**

Explanation: The text used to define a literal exceeds the declared length of the literal.

User Response: Either change the initializing data or the explicit length of the literal.

ABX1195E **You can only initialize a PACKED field with a PACKED literal**

Explanation: When initializing a packed field only an optional preceding minus sign and the numbers zero to nine are allowed.

User Response: Correct the literal value.

ABX1196E **Data is not packed data**

Explanation: Non numeric data is in a literal identified as packed.

User Response: Correct the literal value.

ABX1197E **Cannot have a packed size greater than 16**

Explanation: The z/OS hardware only supports 31 digits when packed - or 16 bytes of packed data.

User Response: Correct the literal value.

ABX1198E **Literal size causes loss of PACKED data**

Explanation: The value has too many digits to fit in the explicitly specified length.

User Response: Correct either the value or the explicit length.

ABX1199E **A string type literal cannot be converted to another type**

Explanation: A string literal like 'Miss Smith' cannot be converted to anything but a character literal. For Packed, the format is P'dddd', for hex it is x'hhhh'.

User Response: Correct either the value or the data type as expressed.

ABX1200E **Explicitly specified length will truncate the literal supplied**

Explanation: The data will not fit in the explicit length expressed.

User Response: Correct either the value (data) or the explicit length.

ABX1201E ***data specified; length-specified vs. length-required***

Explanation: The literal data specified (and its required length) versus what was specified in the explicit length. This message accompanies ABX1200E above.

User Response: Correct the literal as specified under message ABE0200E above.

ABX1202I **At Lit Cap, number =** ***captured-number***

Explanation: This is a debugging message and only appears in OPTIONS (... DEBUG...) mode. It is of little meaning to the user.

User Response: Ignore this message.

ABX1203E **For a integer type of literal, the length (*length- specified*) is invalid**

Explanation: For integer literals, only an explicit length of four is allowed.

User Response: Correct this syntax.

ABX1204E **For a specification of INTEGER, the length of (*length-specified*) is invalid**

Explanation: The only supported length for an explicitly expressed integer literal is four.

User Response: Correct this literal.

ABX1205E **For a specification of SHORT, the length of (*length-specified*) is invalid**

Explanation: The only supported length for a short integer is two.

User Response: Correct this literal.

ABX1206E **The length of *length-specified* is invalid for FLOAT**

Explanation: Float only allows a length of four.

User Response: Correct this literal.

ABX1207E **The length of *length-specified* is invalid for DOUBLE**

Explanation: DOUBLE can only be defined with a length of eight.

User Response: Correct this literal.

ABX1208E **Invalid data type for an integer or literal**

Explanation: When defining an integer, the data can only contain an optional leading minus sign and the numbers zero to nine.

User Response: Correct this literal.

ABX1209E **The length of *length-specified* is invalid for FLOAT**

Explanation: Float always has a length of four.

User Response: Correct this literal.

ABX1210E **The length of *length-specified* is invalid for DOUBLE**

Explanation: Double always has a length of eight.

User Response: Correct this literal.

ABX1211E **Invalid length (*length-specified*) for a real number**

Explanation: Real has a length of four.

User Response: Correct this literal.

ABX1212E **Literal type not identified**

Explanation: The data type specified for a literal is not recognized.

User Response: Correct the syntax.

ABX1213E **Tried to resolve a name (*found-name*) of invalid type *(nn)***

Explanation: This is an internal error. The syntax called for a name, but the allowed type of name is invalid.

User Response: Forward to your support personnel.

ABX1214E **Table *table-name* not found**

Explanation: This is an internal error. The named internal table should have been found. There has probably been memory corruption.

User Response: Forward this output to your support personnel.

ABX1215E **Name sought-name is not known**

Explanation: The syntax referenced a name that has not (yet) been defined.

User Response: Either change the reference, or define the object.

ABX1216E **Table of the source segments not found**

Explanation: In DLI mode, the source segments are defined from the source DBD. The source DBD was not resolved.

User Response: Correct the DD that references the DBD, or the DBD name.

ABX1217E **Expected a left parenthesis after the word SEGMENT**

Explanation: The SEGMENT syntax is SEGMENT (*segment-name*)... the opening (left) parenthesis was not found.

User Response: Correct the segment definition.

ABX1218E **The definition for *segment-name* was not found in the source DBD**

Explanation: In a SEGMENT declarative, the *segment-name* is not defined in the source DBD.

User Response: Correct either the segment name, or refer to a different DBD.

ABX1219E **Expected a right parenthesis after the SEGMENT name**

Explanation: The syntax is SEGMENT (*segment-name*)... The closing (right) parenthesis after the segment name was not found.

User Response: Correct the syntax in the segment declarative.

ABX1220E **Found no use for word *found-word***

Explanation: A word that is not in the allowed syntax was found.

User Response: Either remove the word, or possibly correct the spelling.

ABX1221E **Could not open table of old (source) Segment names in memory**

Explanation: This is indicative of 'out of memory' and usually accompanies a message stating that there is insufficient memory.

User Response: Increase the region size.

ABX1222E **Could not open table of new (target) Segment names in memory**

Explanation: This is normally a short on memory problem.

User Response: Increase the region size.

ABX1223E **Table of source segment names not found**

Explanation: This message indicates a memory corruption problem.

User Response: Forward to your support personnel.

ABX1224I **From the source DBD definition we are defining field (*field-name*)**

Explanation: If in DLI mode and when OPTIONS (... NOFIELDS...) is not specified, the fields defined in the DBD definition are displayed.

User Response: If you do not want this display, specify OPTIONS (... NOFIELDS...)

ABX1225I **at scope SEGMENT (*segment-name*) as position = starting-position, length = length, data type = data-type**

Explanation: The field as defined in the source DBD.

User Response: You can suppress this display with OPTIONS (... NOFIELDS...)

ABX1226E **The field *field-name* is already defined**

Explanation: An attempt has been made to define a field at the same scope as an already-defined field.

User Response: Either discard this definition or chose another name for this field.

ABX1227E **Table put failed with return code *nn***

Explanation: This is usually indicative of lack of memory, if accompanied with an out of memory message. Else it may be a case of corrupted memory.

User Response: If out of memory, increase the region. Else, pass this to your support personnel.

ABX1228E **The segment *segment-name* is already defined**

Explanation: This occurs whenever the same segment is defined more than once in the syntax.

User Response: Correct (remove) the definition.

ABX1230I **Could not find segment named *segment-name* in the source DBD**

Explanation: There is a reference to a segment that is not defined in the source DBD.

User Response: Correct or remove this segment from the syntax.

ABX1232E **Table put failed with return code *nn***

Explanation: This is usually indicative of lack of memory, if accompanied with an 'out of memory' message. Else it may be a case of corrupted memory.

User Response: If out of memory, increase the region. Else, pass this to your support personnel.

ABX1233I **Supplied 'print enabling' word is incorrect. Ignoring it**

Explanation: This message should not appear. If it does, there is a logic error in the program.

User Response: If this message appears, ignore it.

ABX1234I **Data content will not be printed due to the lack of the 'print enabling' word**

Explanation: Either an error in the data was found and the system would have printed the data, or the user requested record printing, but the print enabling word was not supplied.

User Response: Either ignore this message, or supply the print enabling word.

ABX1235I **Data content suppressed due to no 'print enabling' word**

Explanation: Either an error in the data was found and the system would have printed the data, or the user requested record printing, but the print enabling word was not supplied.

User Response: Either ignore this message, or supply the print enabling word.

ABX1236I **ABXUTIL return code = nn**

Explanation: The run is complete. This is the final return code of this execution.

User Response: If the return code is not zero, you should examine the printed output for messages and take corrective action.

ABX1237W **CAUTION! Changing segment *segment-name* from VARIABLE to FIXED. Be sure to verify that resulting alignment and slack bytes are correct**

Explanation: In DLI mode, a segment in the source DBD is defined as variable in length. In the target DBD it is defined as fixed. This results in deleting the first two bytes of data (the length) from the source record. For any fields that need to be on a word or double word boundary, this 'left shift' of two bytes may cause misalignment in the produced record. An explicit formatting SELECT can eliminate this potential problem.

User Response: Examine the produced record to see if any slack bytes should have been inserted.

ABX1238W **CAUTION! Changing segment *segment-name* from FIXED to VARIABLE. Be sure to verify that resulting alignment and slack bytes are correct**

Explanation: The input segment record is fixed length. The output is variable length. This causes two additional bytes to precede the output record and this addition may cause misalignment in the produced record.

User Response: Examine the produced record to see if all the fields are on acceptable alignments.

ABX1239E **The Hex offset in this FIELD is invalid data**

Explanation: In this field, the specified offset is represented as hex, but contains non-hex data.

User Response: Correct the definition.

ABX1240E **The Hex offset in this FIELD is too large**

Explanation: In defining a field, the offset places the data outside of the bounds of the record.

User Response: Correct the field definition.

ABX1241I **This Field (*field-name*) is defined to be at offset *nnn***

Explanation: This message tells the user where the field lies. When defining fields using the NEXT approach, this is useful to assure the proper positioning.

User Response: Check to see if this is the expected result.

ABX1242I **This Field (*field-name*) is defined to be at position *nnn***

Explanation: This is just like ABX1242I above, but is emitted when in 'position' mode, not 'offset' mode.

User Response: Assure that the expected result is as desired.

ABX1250E **Found no use for the word *found-word***

Explanation: An extraneous word was appended to valid text. This may be the misspelling of a valid optional word.

User Response: Correct the statement.

ABX1251E **In select list, word = *found-word***

Explanation: In parsing a list of fields in a select list, an invalid word was found.

User Response: correct the select list.

ABX1252E **The word (*found-word*) is not a literal, field, collect or content name**

Explanation: In parsing a SELECT, a word that should have been one of the above is not defined as such.

User Response: Correct the occurrence of this word, either by defining it or by removing it.

ABX1253E **Data type *(nn)* in a select list is invalid**

Explanation: In processing a SELECT list, a word that has no discernible data type -- usually a word looking like a literal-- could not be processed.

User Response: Correct the word.

ABX1254E **Ran out of text in an select list**

Explanation: A SELECT list never terminates a statement. It is normally followed by the word FROM not preceded by a comma.

User Response: Correct the select list.

ABX1255E **Cannot use NOTHING with other things in a column list**

Explanation: SELECT NOTHING is specified as SELECT NOTHING FROM... It can not have other named components such as literals or field names.

User Response: Correct the syntax by either deleting the word 'nothing' or by deleting the other words in the SELECT list.

ABX1256E **There can only be one select marked final in GROUP mode(DLI)**

Explanation: In DLI GROUP mode, there is a SELECT that when met (is true) establishes that all accumulated and selected records for this group (selected segments from this root down) are to be emitted. There can only be one of these marked SELECTs.

User Response: Correct the syntax - either this SELECT or the prior SELECT marked FINAL.

ABX1257E **There can only be one 'select * scrambling...' in a select**

Explanation: For scrambling, one names each field to be scrambled with the exception of the text SELECT * scrambling column-list. There can only be one occurrence of this phrase within this SELECT.

User Response: Correct the syntax

ABX1258E **Select list did not end with the word 'FROM' - *found-word***

Explanation: A SELECT list is a list of fields separated by commas. The first word appended to a SELECT list (no preceding comma) must be the word FROM.

User Response: Either a comma is missing between named fields and literals, or the word FROM is misspelled.

ABX1259E **Invalid column type in select list (*found-word*);**

Explanation: In a SELECT list, a word is not defined in a manner that allows it to be used in a SELECT list. Either the word is misspelled or not defined at all or is a non-valid word (such as containing non-valid characters).

User Response: Correct the syntax.

ABX1260I **This produced record will be *nnn* bytes long including *mmm* slack bytes**

Explanation: This informational message describes the record being produced from the SELECT list.

User Response: This length should be what was expected from the described SELECT list.

ABX1261I **plus the length of the source record**

Explanation: This message is attached to ABX1260I above and indicated that an asterisk (*) was one part of the SELECT list.

User Response: This is informational only.

ABX1262I **and any needed post record slack bytes**

Explanation: In addition to the above message, there may be slack bytes preceding the asterisk (whole record) inserted between what leads or follows the asterisk indication in the actual record. This assures that the asterisk portion starts on a double word.

User Response: If this is not intended, use a field definition of position one, length *, data type CHAR or HEX, which requires no alignment.

ABX1263I **and the resolved length of filed name**

Explanation: Associated with the above, there is a field defined in the SELECT list that uses the length indicator of asterisk. This is resolved at run time and can be variable for variable length records.

User Response: This is informational only.

ABX1265I **Select table name is table-name**

Explanation: In the SELECT, the word used to specify the 'table' is listed. This must be the source segment name in DLI mode. In VSAM or sequential, the name is a required field but has no specific definition.

User Response: This is informational unless followed by an ABX message stating that the name is not the associated segment name required.

ABX1266E **The word *found-word* is too long for a table name**

Explanation: In DLI mode, the 'table name' must be the same as the source segment name, which is itself limited to eight characters.

User Response: Correct the 'table' name.

ABX1267I **The minimum source record length for executing the select list and predicates is *nnn* bytes**

Explanation: At run time, any record shorter that this specification cannot have the predicates properly evaluated, and/or cannot supply the full text requested for output. Any record being too short for this need, will be skipped - with a message emitted for the first occurrence of this event.

User Response: Examine the run time messages to see if any short records were skipped.

ABX1268E **Minimum (*nnn*) exceeds Maximum (*mmm*)**

Explanation: Both a minimum length (*nnn*) and a maximum length (*mmm*) were specified. As specified, the minimum is greater than the maximum and is therefore invalid.

User Response: Correct either the minimum or the maximum.

ABX1269I **Printing the select list for statement *nnn***

Explanation: The definition for statement *nnn* requested printing of the content (for the first *mmm* records). This record content follows.

User Response: This explains the data that prints after this message.

ABX1270E **In formatting a record, FIELD *field-name* extends beyond the input record**

Explanation: A SELECT list is being formatted from an input record, but the actual record is too short to contain all of the content for the named field. This can happen whenever a record is not of the 'type' thought.

User Response: Perhaps TYPE definitions need to be specified.

ABX1271E **Attempting to use a COLLECT (*collect-name*) that has not been set**

Explanation: In formatting a SELECT list, one of the specified 'fields' is actually a COLLECT. This particular collect has yet to be populated. Probably the associated TYPE of record has not been received as of yet.

User Response: See if the COLLECT is truly called for, and that the requisite record is present in the data stream.

ABX1272E **Invalid column type in select list (*found-word*)**

Explanation: What should be a field, collect, named literal, explicit literal or content expression is not.

User Response: Correct the statement.

ABX1273E **Can only scramble a field**

Explanation: You cannot scramble a literal.

User Response: Correct the syntax.

ABX1274E **In SCRAMBLING a field, FIELD *field-name* extends beyond the input record**

Explanation: In the process of scrambling a field, the content extends beyond the read record.

User Response: Perhaps TYPE statements are needed to process shorter records in a different manner.

ABX1275E **Produced record (length-produced) greater than size specified in 'fixed (specified-fixed length)'**

Explanation: In formatting a record, the produced length exceeds the specified fixed length.

User Response: Either respecify the fixed length, or respecify the content.

ABX1276I **Statement number *nnn***

Explanation: This precedes the following message (ABX1277I).

User Response: No response needed.

ABX1277I **Produced record:**

Explanation: A produced record will be printed due to a print request.

User Response: No action needed.

ABX1279I **End of printing for statement *nnn***

Explanation: This bounds the text printed for statement *nnn*.

User Response: No action required.

ABX1280E **Produced record *nn* greater than size specified in MAXIMUM (*mmm*)**

Explanation: In formatting a record, its developed length exceeded the dictated maximum.

User Response: Change the maximum, or change the content of the produced record.

ABX1281I **Statement number *nnn***

Explanation: The following text is for a formatted record as requested in statement *nnn*.

User Response: No response needed.

ABX1282I **Produced record:**

Explanation: The actual produced text.

User Response: No action required.

ABX1283I **In column list, word = word**

Explanation: IN displaying the produced record, this is one 'field'

User Response: No action required.

ABX1284E **Cannot have a literal in a UNIQUE list**

Explanation: UNIQUE was requested, but there is no need to check a literal for uniqueness because it is a constant.

User Response: Correct the UNIQUE ON... request.

ABX1285E **Cannot have a literal in a SCRAMBLING list**

Explanation: It makes no sense to scramble a literal - which is always a constant.

User Response: Correct the statement.

ABX1286E **Cannot have a SPACE control in a UNIQUE list**

Explanation: Space control is for formatting an output record, not for building the concatenated unique string.

User Response: Remove the spacing directive from the unique instructions.

ABX1287E **The word (*found-word*) is not a literal, field, content or collect name**

Explanation: The shown *found-word* is not recognized.

User Response: Either correct the word or define it.

ABX1288E **Cannot have a literal name in a UNIQUE list**

Explanation: It makes no sense (and wastes storage) to have a constant literal in a unique array.

User Response: Correct this directive.

ABX1289E **Cannot use a literal name in a SCRAMBLE list**

Explanation: It makes no sense to scramble a literal as it is static and constant.

User Response: Correct this statement.

ABX1290E **Cannot have a collect in a UNIQUE list**

Explanation: This feature is not supported as a collect comes from a prior record, not this current record.

User Response: Correct this statement.

ABX1291E **Cannot use a collect name in an INSERT statement (can in a THEN INSERT)**

Explanation: Inserts are for LOAD mode where there is no source record from which to collect data.

User Response: Correct this statement.

ABX1292E **Cannot use a field name in an INSERT statement (can in a THEN INSERT)**

Explanation: Insert is for LOAD mode, where there are no input record and therefore no fields.

User Response: Correct this statement.

ABX1293E **Data type (*nnn*) in a column list is invalid**

Explanation: An invalid word appears on a list of columns or fields.

User Response: Correct this syntax.

ABX1294E **List does not start with a left parenthesis**

Explanation: In parsing a column list, the list requires commencement with a left parenthesis.

User Response: Correct this syntax.

ABX1295E **Ran out of text in a column list**

Explanation: The statement ended while still in a list.

User Response: Correct this statement.

ABX1296E **Column list did not end with a right parenthesis**

Explanation: A list starting with a left (open) parenthesis, ends with a right (close) parenthesis.

User Response: Correct this syntax.

ABX1297E **Expected the word INSERT**

Explanation: The syntax requires the word INSERT following the word 'THEN'.

User Response: Correct this statement.

ABX1298E **Expected the word INTO**

Explanation: When parsing an INSERT or THEN INSERT, the word INTO was not found where the syntax dictates.

User Response: Correct this statement.

ABX1300E **Expected the name of the insert table**

Explanation: For an INSERT or THEN INSERT, the next word after INTO is the table (by syntax). For DB2, this name must be the target segment; for VSAM and sequential, any name will work. For insertion into a file, the syntax is INTO dd:ddname.

User Response: Correct this statement.

ABX1301E **The name of the insert into table is excessive in length (*nnn*)**

Explanation: When processing an INSERT or THEN INSERT, the word declaring the 'table' is excessive in length. For DLI, this maximum length is eight - the limit on a segment name.

User Response: Correct this statement.

ABX1302E **Expected the word VALUES**

Explanation: In parsing an INSERT or THEN INSERT, the text is INTO TABLE_NAME VALUES. The word VALUES was not found.

User Response: Correct this statement.

ABX1303E **Expected the VALUES list**

Explanation: When parsing an INSERT or THEN INSERT, the text following the word VALUES is the content to insert (literals, *field-name*s, etc). This is in the form of a list in parentheses.

User Response: Correct the statement.

ABX1304E **Minimum (*nnn*) exceeds Maximum (*mmm*)**

Explanation: When specifying both a minimum and a maximum, the minimum must not be larger than the maximum.

User Response: Correct this statement.

ABX1305E **Invalid column type in insert list (*found-word*);**

Explanation: In parsing a list, an invalid word was found.

User Response: Correct this word.

ABX1306I **This produced record will be *nnn* bytes long including *mmm* slack bytes**

Explanation: This is an informative message describing the produced record.

User Response: No action is required if this portrays the expected result.

ABX1307I **plus the length of the source record**

Explanation: This message accompanies ABX1306I above if an asterisk was expressed in the formatting description.

User Response: No action is required.

ABX1308I **plus the resolved length of *field-name***

Explanation: This message accompanies ABX1306I above whenever there is a field with an asterisk type of length dictated.

User Response: No action is required.

ABX1309I **The minimum source record length for executing this insert is *nnn* bytes**

Explanation: The insert list's needs and the predicate needs for this input record require a minimum source record for evaluation. Shorter records (of this type) will cause run time messages.

User Response: No action is required unless run time messages associated with short records appear. This case may cause the user to need to TYPE source records.

ABX1310I **Printing the insert list for statement *nnn***

Explanation: The user requested printing of this record. The record's content follows.

User Response: No user action is required.

ABX1311E **In formatting a record, FIELD *field-name* extends beyond the input record**

Explanation: A record too short to produce all the fields requested from this record was read.

User Response: One or more possible TYPEs are needed. Else the formatting requested needs to change.

ABX1312E **Attempting to use a COLLECT (*collect-name*) that has not been set**

Explanation: The content of a collect statement is set when it is executed in association with a select with predicates that are true. Until it is at least set once, it cannot participate in formatting an output record.

User Response: Correct the fact that the collect has not been instantiated.

ABX1313E **Invalid column type in select list *found-data)***

Explanation: In formatting a select list, the data represented in the record is invalid.

User Response: Correct this data, or the definition of the data.

ABX1314E **Produced record (*nnn*) greater than size specified in 'fixed (*mmm*)'**

Explanation: When building a record from the select list, the accumulated data exceeds the expressed maximum for this record.

User Response: Either change the maximum or correct the content that accumulates for producing this record.

ABX1315I **Statement number *nnn***

Explanation: This message accompanies ABX1314E above and portrays the statement in the syntax stream that is being effected.

User Response: Either change the maximum or correct the content that accumulates for producing this record.

ABX1316I **Produced record:**

Explanation: This message accompanies messages ABX1314E and ABX1315E above.

User Response: Either change the maximum or correct the content that accumulates for producing this record.

ABX1318I **End of printing for statement *nnn***

Explanation: This message accompanies message ABX1314E above and identifies the end of the print for the produced record.

User Response: Either change the maximum or correct the content that accumulates for producing this record.

ABX1319E **Produced record (*nnn*) greater than size specified in MAXIMUM (*mmm*)**

Explanation: In formatting an output record, the produced data exceeded the specified maximum length.

User Response: Change the formatting instructions or specify a greater maximum.

ABX1320I **Statement number *nnn***

Explanation: This is the statement number of the formatting instructions associated with message ABX1319E above.

User Response: Change the formatting instructions or specify a greater maximum.

ABX1321I **Produced record:**

Explanation: If printing is permitted (the print enabling word was supplied) this is the record that was produced.

User Response: Change the formatting instructions or specify a greater maximum.

ABX1322E **Expected a left parenthesis**

Explanation: In parsing a COLLECT statement, the first valid syntax after the word COLLECT must be a left parenthesis.

User Response: Correct this statement.

ABX1323E **Expected a COLLECT NAME**

Explanation: In parsing a COLLECT statement the syntax is COLLECT (*collect-name*)... and this name is not present.

User Response: Correct the statement.

ABX1324E **Expected a right parenthesis**

Explanation: In a COLLECT statement, the right (closing) parenthesis after the name is not found.

User Response: Correct this statement.

ABX1325E **Syntax requires a right parenthesis after the collect name**

Explanation: In defining a COLLECT statement, the *collect-name* must be followed with a right parenthesis.

User Response: Correct this statement.

ABX1326E **Expected the word FROM**

Explanation: In parsing a COLLECT statement, the syntax required is COLLECT (*collect-name*) FROM... This word, from, was not found.

User Response: Correct the statement.

ABX1327E **Syntax requires the word FROM**

Explanation: The word FROM was not found to follow the *collect-name* in processing a COLLECT statement.

User Response: Correct this statement.

ABX1328E **Expected a FIELD NAME after FROM**

Explanation: The syntax for a COLLECT statement is COLLECT (*collect-name*) FROM *field-name*. This field name was not found.

User Response: Correct the statement.

ABX1329E **The FIELD or COLLECT name (*found-word*) is not known**

Explanation: The source of the COLLECT is not an identified FIELD or COLLECT.

User Response: Correct this statement.

ABX1330E **The name (*found-word*) is not a FIELD name**

Explanation: The source of the data for this COLLECT is not a defined FIELD.

User Response: Correct this statement.

ABX1331I **Inheriting scramble from the source field**

Explanation: In a COLLECT, the source FIELD is scrambled, so this field is also to be scrambled.

User Response: No action is required.

ABX1332E **Don't support a length of '*' in a FIELD for COLLECT**

Explanation: The 'until the end of the record' asterisk length indication is not supported for a source in a COLLECT.

User Response: Correct this statement.

ABX1333E **Don't support collecting from a LITERAL or a COLLECT**

Explanation: The source of the data in a collect must be a FIELD.

User Response: Correct this statement.

ABX1334E **The name *collect-name* is already defined**

Explanation: This name is not unique at the required scope.

User Response: Use a different name for this COLLECT.

ABX1335E **Table put failed with return code *nnn***

Explanation: On saving the information for this COLLECT, the routines to add it to a table failed. If this message accompanies an 'out of memory' message, increase the region.

User Response: If there is not an accompanying 'out of memory' message, there is probable memory corruption. In that case, forward this output to your support personnel.

ABX1336E **This COLLECT name *collect-name* is already defined**

Explanation: Collect names must be unique within their global scope.

User Response: Choose another name for this COLLECT.

ABX1337E **Table put failed with return code *nnn***

Explanation: On saving the information for this COLLECT, the routines to add it to a table failed. If this message accompanies an 'out of memory' message, increase the region

User Response: If there is not an accompanying 'out of memory' message, there is probable memory corruption. In that case, forward this output to your support personnel.

ABX1338E **Word (*found-word*) has no use**

Explanation: In processing a COLLECT, an extraneous word was found.

User Response: Either remove this word, or possibly correct the spelling.

ABX1339E **Word (*found-word*) has no use**

Explanation: In parsing a COLLECT, an extraneous word was found.

User Response: Remove this word, correct the spelling, or possibly the ending semi-colon is misplaced.

ABX1340E **Cannot reference a field (*field-name*) with a length of ***

Explanation: The 'until end of record' length indicator is not supported in a field used in a COLLECT expression.

User Response: Correct this statement.

ABX1341E **Expected 'ON' after the word 'UNIQUE'**

Explanation: The syntax for UNIQUE must name the fields constituting the combination that must be unique. The required syntax is UNIQUE ON...

User Response: Correct this statement.

ABX1342E **Can only use field names in a UNIQUE list**

Explanation: The data content of fields make up the uniqueness of the record. Literals and the like are not supported.

User Response: Correct the syntax.

ABX1343E **Cannot use a length type of * in a UNIQUE list**

Explanation: In a definition of uniqueness, the variable length field definition of asterisk is not supported.

User Response: Correct the statement.

ABX1344E **No fields in the UNIQUE list**

Explanation: A UNIQUE statement exists with no fields referenced for developing the unique constraint.

User Response: Either correct the statement, or remove it.

ABX1345E **In executing a UNIQUE request, FIELD *field-name* extends beyond the input record**

Explanation: The definition of a UNIQUE constraint uses a field that extends beyond an actual record. Possibly TYPE definitions are required.

User Response: Either use TYPE definitions or remove the offending source record.

ABX1346I **LITERAL *(literal-name)*CHAR or HEX:**

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1347I **LITERAL ((*literal-name)*INT: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1348I **LITERAL ((*literal-name)*SMALL INT: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1349I **LITERAL *(literal-name)* FLOAT: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1350I **LITERAL (literal-name) DOUBLE: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1351I **LITERAL *(literal-name)*PACKED: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1352I **COLLECT (*collect-name*) CHAR or HEX:**

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1353I **COLLECT (*collect-name*) INT: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1354I **COLLECT (*collect-name*) SMALL INT: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1355I **COLLECT (*collect-name*) FLOAT: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1356I **COLLECT (*collect-name*) DOUBLE: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1357I **COLLECT (*collect-name*) PACKED: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1358I **FIELD (*field-name*) CHAR or HEX: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1359I **FIELD (*field-name*) INT: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1360I **FIELD (*field-name*) SMALL INT: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1361I **FIELD (*field-name*) FLOAT: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1362I **FIELD (*field-name*) DOUBLE: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1363I **FIELD (*field-name*) PACKED: *value***

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1364I **For the field '*'**

Explanation: This is a debugging message emitted in processing a print request and may be of value to the user.

User Response: Ignore this message if the data is as expected.

ABX1365E **Improper format of the PRINT directive**

Explanation: The print directive has two formats: PRINT or PRINT (*nnn*).

User Response: Correct the syntax.

ABX1367E **Improper format of the MINIMUM directive**

Explanation: The minimum directive has the format MINIMUM(*nnn*).

User Response: Correct the statement.

ABX1369E **The specified MINIMUM(*nnn*) exceeds the specified FIXED(*nnn*)**

Explanation: Fixed was specified, but the minimum as specified exceeds that declared fixed size.

User Response: Correct either the minimum value or the fixed value.

ABX1370E **MINIMUM is of the format MINIMUM(*nn*)**

Explanation: An improper format was specified for MINIMUM.

User Response: Correct the statement.

ABX1371E **Improper format of the MAXIMUM directive**

Explanation: Maximum is specified as MAXIMUM(*nnn*).

User Response: Correct the syntax.

ABX1373E **The specified MAXIMUM(*nnn*) exceeds the specified FIXED(*nnn*)**

Explanation: Fixed was specified, but the maximum as declared is in contrast to the fixed declaration.

User Response: Correct either the FIXED or the MAXIMUM.

ABX1374E **MAXIMUM is of the format MAXIMUM(*nnn*)**

Explanation: The word MAXIMUM was found, but the rest of the declaration is not the correct format.

User Response: Correct the statement.

ABX1375E **Improper format of the SPUFI directive**

Explanation: SPUFI (for process modes sequential or VSAM or DLI) is specified as SPUFI (*nnn*) or just SPUFI with a default of one.

User Response: Correct the declaration.

ABX1376E **Improper format of the SPUFI directive**

Explanation: SPUFI is specified as either SPUFI or SPUFI(*nnn*).

User Response: Correct the format.

ABX1377E **Improper format of the TABLE directive**

Explanation: In parsing a THEN INSERT statement or a SELECT, the syntax is incorrect. This message is emitted when the required left (opening) parenthesis does not follow the word TABLE.

User Response: Correct this statement.

ABX1379E **Invalid name in the TABLE*(name)*directive**

Explanation: The name as specified in the INSERT or SELECT for the target SPUFI formatting is not a valid DB2 name.

User Response: Correct the name.

ABX1380E **Improper format of the TABLE directive**

Explanation: The TABLE expression as used in the SELECT or THEN INSERT statement has the format TABLE(table-name).

User Response: Correct this declarative.

ABX1382E **Invalid name in the QUALIFIER*(name)* directive**

Explanation: For the use of SPUFI on a SELECT or THEN INSERT, the required verbiage is QUALIFIER (qualifier-name).

User Response: Correct the statement.

ABX1383E **Do not support a field length of '*' in SPUFI - ignoring *field-name***

Explanation: SPUFI output does not support a variable length input field.

User Response: Correct the field definition.

ABX1384E **Found no named FIELDS or LITERALS or COLLECTS to build SPUFI**

Explanation: An invalid identifier is in the text.

User Response: Either define this identifier or remove the reference.

ABX1385I **This produced record will**

Explanation: When processing a SELECT statement, the developed output record will be so many bytes long, as portrayed in the following messages.

User Response: No action is required if the produced text is as described. Else correct the 'column list' in the SELECT.

ABX1386I **(Note: has a two byte length field in the input record)**

Explanation: The produced segment (DLI mode) will have a two byte length field as the TARGETDBD definition for this segment calls for variable length. This message is logically a continuation of ABX1385I above.

User Response: Informational only, no action is required if the intent is for a variable length segment.

ABX1387I **(Note: has a four byte length field in the input record)**

Explanation: The output records to be written into ABXOUT will have a four byte length field as RECFM V or VB was specified. This message is logically a continuation of ABX1385I above.

User Response: No action is required is variable length output records are desired.

ABX1388I **contain the source record's data**

Explanation: One of the 'fields' defined for formatting the record is the asterisk (*) - meaning the whole input record (less and LL or LLZZ fields). This is logically a continuation of message ABX1385I above.

User Response: None is required.

ABX1389I **have any necessary slack bytes after the '*' data to set the Origin**

Explanation: Whenever an asterisk (the whole record) is selected, there may be slack bytes preceding it (if there are fields or collects or literals preceding it) and there may be slack bytes following it if there are fields or literals or collects following the asterisk. This message is logically a continuation of message ABX1385I above.

User Response: This message is informational only.

ABX1390I **have *nnn* bytes of selected/inserted data including *mmm* slack bytes**

Explanation: This is the sum of the explicit items selected in the list. This message is logically a continuation of ABX1365I above.

User Response: No action required.

ABX1391E **The ddname must be 1 to 8 characters, found *found-word* characters**

Explanation: In the Z/OS operating system, ddnames are from 1 to 8 bytes of text starting with a character.

User Response: Correct the ddname.

ABX1392E **The ddname *found-ddname* is invalid as it is one of the DD names used by ABXUTIL**

Explanation: ABXUTIL reserves the ddnames ABXCTL, ABXPRINT, ABXIN, ABXOUT, and SYSSPUFI.

User Response: Choose another name for your DD name.

ABX1393E **The ddname *found-ddname* is already defined as an input DD**

Explanation: A RELATIONSHIP has already been defined with this ddname.

User Response: Choose another DDNAME for this THEN INSERT.

ABX1401I **OPTION DUMMY specified - nothing written to ABXOUT**

Explanation: This is an informational message. Due to the election of OPTIONS(... DUMMY...), nothing will be written to ABXOUT.

User Response: No user action is required.

ABX1402E **Could not get input buffer**

Explanation: During initialization for running, a suitable input buffer could not be acquired.

User Response: Increase the region.

ABX1403I **Could not open SYSSPUFI - turning off this option**

Explanation: The DD card for SYSSPUFI is likely missing. The election of SPUFI is negated because this file could not be opened.

User Response: Supply a DD named SYSSPUFI, or suppress activating SPUFI.

ABX1405E **Table table-name not found**

Explanation: A required memory resident table was not found during execution. This is probably indicative of memory corruption.

User Response: Forward this print to your support personnel.

ABX1406E **Could not get output buffer**

Explanation: Insufficient free memory was available to acquire the output buffer.

User Response: Increase the region size.

ABX1407I **Formatting data in DLI format**

Explanation: The system is running in DLI mode of operation.

User Response: No user response is required.

ABX1408I **Formatting data in Sequential format**

Explanation: The system is running in sequential mode.

User Response: No user response is required.

ABX1409E **Not an identified Segment in source DBD: *segment-name***

Explanation: A segment was read from the input (ABXIN) that is not identified in the SOURCEDBD.

User Response: Check that the correct file is referenced by ABXIN.

ABX1410E **Failure in writing to ABXOUT**

Explanation: The program was unable to write to ABXOUT. This is most commonly associated with an 'out of space' condition and will often have ab E35 type of message in the SYSOUT.

User Response: Allocate more space for ABXOUT.

ABX1411E **A record without a select exceeded the specified fixed length**

Explanation: A record that was just to be copied was too long.

User Response: Perhaps declarations of TYPE should be specified, or the data is in error.

ABX1412E **Could not open DD named *ddname***

Explanation: A THEN INSERT was to be executed, but the file could not be opened. The probable explanation is that the DD card is missing.

User Response: Add the appropriate DD card to the JCL stream.

ABX1413I **For statement *statement-number* the record read (length of *length-read*) is below the minimum length of *minimum-length* required to evaluate**

predicates, this message will not repeat on further occurrences

Explanation: To process predicates, or to format a record from this record as a source of data, the record must contain data at least through its required length. The read record is too short. The use of TYPES is probably required.

User Response: Most likely TYPE definitions are required, or the record itself may be in error.

ABX1414E **Trouble in evaluating the select predicate...**

Explanation: A predicate could not be evaluated. This message usually accompanies other messages portraying the base cause, such as the data found was not packed.

User Response: The record is probably mistyped or is an invalid record.

ABX1415I **Due to UNIQUE requested in statement *statement-number*, bypassing record *nnn***

Explanation: The record just processed was redundant in content to a previous record as far as the unique constraints specified by the user. The record is skipped.

User Response: Probably no response is required. However, if this occurrence is not expected, the message may require some action.

ABX1416E **Trouble in evaluating a UNIQUE for statement *nnn***

Explanation: The record just read could not be processed for the UNIQUE tests. This is typically a short record. TYPES might be employed to properly process different record types.

User Response: Decide if TYPES are applicable to the data set, and proper processing.

ABX1417E **Trouble in formatting the record**

Explanation: A record could not be formatted from the content of the source record. This is possibly a record that is too short to contain all the needed data. Examine the data to see if perhaps TYPES should be used.

User Response: Examine the source data to see if it is properly described.

ABX1418E **Failure in writing to ABXOUT**

Explanation: The processing failed when trying to write to the primary output stream (ABXOUT). This is typically a full data set issue.

User Response: If there is indication that the output data set is full, allocate more space.

ABX1419E **Trouble in formatting the record**

Explanation: The output record could not be formatted from the input record. This message usually accompanies other messages explaining the problem.

User Response: Correct the definition of the data, or the data itself, or employ TYPING.

ABX1420I **For DBD *identifier*, *nnn* segments meeting select criteria were discarded due to GROUP specification**

Explanation: In the DLI mode of selection, in GROUP mode (where the segments from the root are buffered until the FINAL select is successful), *nnn* records (segment images) were discarded due to the FINAL select failing the predicates.

User Response: This is an end of run informational message and generally needs no action.

ABX1421I **For DBD *identifier*, processed *nnn* data records, selected *mmm*, inserted *xxx*, bypassed *yyy*, then one of *zzz***

Explanation: At the end of processing, these statistics are produced. *nnn* is the number of records read from the input, *mmm* are those selected because either the predicates evaluated as true, or there were no predicates for this record type. *xxx* is the number of records inserted into ABXOUT due to THEN INSERT INTO table-name statements. *yyy* is the count of records skipped over due to predicate evaluations. That is, a bypassed record is one that is skipped (explicitly bypassed) as a result of your definition for the SKIP, LIMIT, THEN ONE OF control syntax declaratives. Then one of *zzz* indicates that after other processing, of the remaining records, 1 in every *zzz* records is also selected.

User Response: This is an informational message and generally requires no action.

ABX1422I **For DBD *identifier*, eliminated *nnn* record(s) because of UNIQUE requests**

Explanation: This is a summary message ate the end of processing. It recaps the count of records in the primary input that were eliminated due to the specified constraints for uniqueness.

User Response: This is an informational message and generally requires no user action.

ABX1423E **Could not get output buffer**

Explanation: During the initialization for execution, memory was not available for the necessary output buffer.

User Response: Increase the region size.

ABX1424I **Creating load mode data in DLI format**

Explanation: The user has requested processing mode DLI and LOAD mode.

User Response: No action is required if the user desires this mode of processing.

ABX1425I **Creating load mode data in Sequential format**

Explanation: The user has requested sequential processing in load mode.

User Response: No action is required unless this was not the intent.

ABX1426E **Trouble in formatting the record**

Explanation: This message accompanies other message(s) whenever the formatting of an output record fails.

User Response: Correct the base cause of the failure as described in the accompanying message(s).

ABX1427E **Failure in writing to ABXOUT**

Explanation: When placing a record onto the primary output file (ABXOUT), an attempt to write failed. This is typically an 'out of space' situation.

User Response: Increase the space allocation of ABXOUT.

ABX1428I **Inserted *nnn* records**

Explanation: At the end of processing, the number of records added to ABXOUT by THEN INSERT INTO target statements is displayed.

User Response: This message is informational and requires no action.

ABX1429I **Input record number *nnn* (SEGMENT *segment-name*), contents:**

Explanation: If print was requested (and allowed by supplying the print-enabling word) this input segment is printed. *nnn* is the sequential number of the record in ABXIN.

User Response: No user response is required.

ABX1430I **Input record number *nnn*, contents:**

Explanation: Print was requested (and allowed because the print-enabling word was supplied). The record prints.

User Response: No user response is required.

ABX1431E **A record exceeded the specified maximum length**

Explanation: A maximum length was requested, but a produced record exceeded this maximum.

User Response: Either change the formatting description or the maximum length allowed.

ABX1432I **Padding record(s) to the specified minimum length (*nnn*)**

Explanation: The produced record is below either the minimum or the fixed specification. The record will be padded.

User Response: No user response is required.

ABX1433I **Formatting data in VSAM format**

Explanation: The user specified OPTIONS (... VSAM...) and the produced records for ABXOUT will be in this format.

User Response: No user response is required.

ABX1434I **Creating load mode data in VSAM format**

Explanation: The user specified OPTIONS (... VSAM... LOAD...) and the produced records will be in this format.

User Response: No user response is required.

ABX1435E **On a DLI load, the target segment (*segment-name*) is not defined in the target DBD**

Explanation: The specified segment is not valid, as it is not defined to the target DBD.

User Response: Correct the specification.

ABX1436I **On a non-DLI processing mode, the target limit (*nnn*) is met - quitting early**

Explanation: The user requested a limit on the output. This has been met and processing will terminate.

User Response: No user response is required.

ABX1437I **Processing ended with the end of data**

Explanation: This is the normal termination of processing when no edit or run-time errors occurred.

User Response: No user response is required.

ABX1438I **On DLI processing mode, the target limit (*nnn*) in the root is met - quitting early**

Explanation: The ROOT limit as specified by the user has been met. The program will terminate.

User Response: No user response is required.

ABX1439W **CAUTION! When going from FIXED to VARIABLE records assure that the resulting field alignment is correct and that appropriate slack bytes are inserted**

Explanation: Going from Fixed to Variable causes the record length to be added to the front of the record. This will change the alignment of fields. Assure that the resulting record is as desired.

User Response: Assure that the output format is as expected.

ABX1440W **CAUTION! When going from VARIABLE to FIXED records assure that the resulting field alignment is correct and that appropriate slack bytes are inserted**

Explanation: Going from Variable to Fixed omits the length indicator from the input record. This may cause field realignment. The user should carefully check the produced records to see that fields are aligned as necessary.

User Response: Assure that the output format is as expected.

ABX1441E **For segment *segment-name* file record *nnn* there is a required conversion from variable to fixed without a formatting select**

Explanation: For the named segment, there is no explicit select. The input record is converted from variable (on the input side) to fixed on the output side. This results in all the data being left shifted by two bytes. This may produce fields that are not aligned to their natural boundaries. This will affect all fields requiring any alignment greater than a half word.

User Response: Supply a formatting select. The system will accept a SELECT *, but the results may be unacceptable to the user.

ABX1442E **For segment *segment-name* file record *nnn* there is a required conversion from fixed to variable without a formatting select**

Explanation: For the named segment, there is no explicit select. The input record is converted from fixed (on the input side) to variable on the output side. This results in all the data being right shifted by two bytes. This may well produce fields that are not aligned to their natural boundaries. This will affect all fields requiring any alignment greater than a half word.

User Response: Supply a formatting select. The system will accept a SELECT *, but the results may be unacceptable to the user.

ABX1443E **For file record *nnn* there is a required conversion from fixed to variable without a formatting select**

Explanation: In sequential processing a record was read that is to be converted from fixed length to variable length. There is no explicit SELECT that specified the fields, so the will be relocated in a block to live behind the length (LLZZ). This may well cause any fields requiring double byte alignment to be placed on a word boundary. The user should have fields defined for this data (and possibly a TYPE) so that alignment can be produced.

User Response: To assure alignment beyond word alignment, specify a specific select column-list (or field-list).

ABX1444E **For file record *nnn* there is a required conversion from variable to fixed without a formatting select**

Explanation: This is an attempt to change from variable to fixed. The first four bytes are eliminated in the source record and the remainder is placed in the output record. Any fields requiring alignment of double will lose their alignment.

User Response: A specific select list should be used specify the data types of the fields so that the required output alignment for doubles can be assured.

ABX1445E **Could not write to DD named ddname**

Explanation: A THEN INSERT into DD:DDNAME could not be performed. The usual problem is space in the output file. This should have a Z/OS type of message in the SYSOUT.

User Response: Allow more space in the output data set.

ABX1446E **Cannot change record size on DD ddname, prior was *nnn*, this is *mmm***

Explanation: Then insert only supports fixed length output records. The situation was found whereby records of a different length we being sent to the same file.

User Response: For different record types and lengths, use different receiving filed (different DD names).

ABX1447I **For DD named ddname there were *nnn* data record(s) of *mmm* bytes each written**

Explanation: At the end of processing, the system reports how many records were written by ddname.

User Response: No response is required.

ABX1450E **Expected end of text**

Explanation: In parsing a predicate, something was found beyond a predicate that did not start another predicate nor did it start a phrase like PRINT or SPUFI or QUALIFIER.

User Response: Correct the text. There is either an unrecognized word following a predicate or possibly a misspelling.

ABX1451E **Expected predicate start**

Explanation: In parsing, a commencement of a predicate was expected, but was not found.

User Response: Correct the text.

ABX1452E **Invalid predicate format**

Explanation: In parsing a predicate, invalid text was found. Such as a non-logical operator when expected.

User Response: Correct the text.

ABX1453E **Expected more text**

Explanation: The statement ran out of text before completing a predicate.

User Response: Perhaps the ending semi-colon was misplaced. Correct the syntax.

ABX1454E **Length of '*' for field (*field-name*) is not supported in a predicate**

Explanation: A predicate used a field with a length specified by the asterisk (meaning the rest of the record). This is not supported.

User Response: Fields used in predicates must have a fixed length (except for DB2 columns).

ABX1455E **For a FIELD name (*field-name*), we have the wrong type of data**

Explanation: In parsing a predicate, the system found a field with one type of data and a second operand with a different data type. In the case of literals, be sure to specify the correct data type - string literal versus integer or float.

User Response: Correct the syntax.

ABX1456E **Expected FIELD name**

Explanation: In parsing a predicate, the syntax calls for a field name as the first operand, but this was not the found case.

User Response: Correct the statement.

ABX1457E **Is not a logical operator**

Explanation: In parsing a predicate, a logical operator (<, >, = LIKE, IN, etc) was expected by syntax, but was not found.

User Response: Correct the syntax.

ABX1458E **Illegal in-list in a predicate**

Explanation: An 'IN' logical operator was used, but the second operand is not either an in-list of the consistent data type, or not a SOURCE of the correct data type.

User Response: Correct the statement.

ABX1459E **Can only do a LIKE on CHAR or HEX**

Explanation: A predicate using a LIKE referenced a FIELD that is not character or HEX data. LIKE does not apply to things like INTEGER, FLOAT, etc.

User Response: Correct the syntax.

ABX1462E **Length of '*' for field (*field-name*) is not supported in a predicate**

Explanation: Fields used in a predicate must have a fixed length (except for DB2).

User Response: Correct either the field definition or the predicate.

ABX1463E **For a FIELD name(*field-name*), we have the wrong type of data**

Explanation: The data to be compared in a predicate does not match the data type of the first field.

User Response: Correct the predicate.

ABX1464E **Mismatched data types in a predicate**

Explanation: The request was made to compare data of two different data types. This is not allowed.

User Response: Correct either the predicate or the data definition of the field(s).

ABX1465E **Can only use the 'ANDED' operator on a one byte field**

Explanation: The logical operator ANDED is used to test the content of a single byte.

User Response: Perhaps a set of AND's are called for: FIELD one ANDED 3 AND FIELD2 ANDED x'3f'

ABX1466E **Length of '*' for field (*field-name*) is not supported in a predicate**

Explanation: Predicates only support fixed length fields.

User Response: Correct either the predicate or the field definition.

ABX1470E **Mismatched data types in a predicate**

Explanation: The data types in the fields or literals in a predicate must be compatible. As an example, you cannot compare a hex field to an integer.

User Response: Correct either the FIELD definition or the predicate.

ABX1471E **Could not open a table (table-name) to capture in-list**

Explanation: There is in-sufficient memory for capturing the contents of an in-list.

User Response: Increase the region size.

ABX1472E **Could not initialize the locate block for table-name**

Explanation: This is probably a condition of memory corruption.

User Response: Forward this output to your support personnel.

ABX1473E **An in list must start with a left parenthesis**

Explanation: The SQL syntax for an in-list begins with a left (opening) parenthesis.

User Response: Correct the in-list.

ABX1474I **Informational - this is a duplication: content**

Explanation: An in-list contains redundant information, that is, the same literal value was previously found in the in-list.

User Response: No action is required unless this entry was perhaps misspelled or meant to have different content.

ABX1475E **Failure in addition to the table table-name, re = *nnn***

Explanation: When populating a memory table for an in-list, an entry could not be added. This is insufficient memory if this message is accompanied with an 'out-of-memory' message, or memory corruption if not.

User Response: If an accompanying 'out of memory' message is found, increase the region. Else forward this output to your support personnel.

ABX1476E **In an in-list, expected a literal, found *found-text***

Explanation: Either a literal is not properly presented (no quotes surrounding a character literal) or is bad text like 123.456f99

User Response: Correct the literal.

ABX1477E **Unexpected word in an in-list**

Explanation: Some content of an in-list, such as a logical operator, is invalid.

User Response: Correct the text.

ABX1478E **In list must end with a right parenthesis**

Explanation: The SQL syntax for an in-list has the list surrounded with parentheses.

User Response: Correct the in-list.

ABX1479E **The first operand in statement *nnn* predicate *predicate text***

Explanation: At run time, the position of the data in the record for the FIELD in this record could not be ascertained.

User Response: Correct the field definition.

ABX1480E **Attempted to use a collect (*collect-name*) that has not been set**

Explanation: In evaluating a predicate, a collect was used. This collect had yet to be populated with data by the time the predicate was evaluated.

User Response: The record type that would have populated the COLLECT was not a precedent to evaluating this predicate. Either a TYPE is wrong, or the file is not in the sequence that the user thinks. Perhaps the file should be sorted.

ABX1483E **Attempted to use a collect (*collect-name*) that has not been set**

Explanation: Same as ABX1480E above.

User Response: See ABX1480E above.

ABX1485E **The second operand in statement *nnn* predicate *mmm***

Explanation: For the statement *nnn*, the address of the second operand in the *mmm* predicate is beyond the length of the read record.

User Response: Either a field is not correctly described, or perhaps TYPE definitions should be applied.

ABX1488E **The third operand in statement *nnn* predicate *mmm***

Explanation: In evaluating a BETWEEN, the third operand has a data address that is beyond the length of the read record.

User Response: Correct the field definition or the predicate or employ TYPE definitions.

ABX1491E **Invalid data type in predicate evaluation stmt *nnn* pred *mmm* type column-data-type**

Explanation: During processing, the data found in this predicate is invalid as described in the field definition.

User Response: Correct either the field definition or the predicate.

ABX1492E **For left arg in statement *nnn*, predicate *mmm*, the data is not valid packed data**

Explanation: The definition for this data in this predicate was packed data, but the content of the record was not packed data.

User Response: Correct either the field definition or the record or the TYPE.

ABX1493E **For right argument in statement *nnn*, predicate *mmm*, the data is not valid packed data**

Explanation: The definition for this data in this predicate was packed data, but the content of the record was not packed data.

User Response: Correct either the field definition or the record, or possibly the TYPE.

ABX1494E **SQL Syntax requires a table name in a select (we require one even in sequential processing)**

Explanation: In parsing a SELECT, the '*table-name*' was not in the required position.

User Response: Correct the statement. A table name is required in all forms of processing.

ABX1501E **Syntax error - Position = *nnn phrases-as-specified-above***

Explanation: This is the general SYNTAX ERROR message. *nnn* is the position where the error starts. *phrases-as-specified-above* is a phrase that is documented in the following set of unnumbered messages that follow:

A word exceeds 256 bytes

Explanation: In parsing the input stream a 'word' which is usually a quoted or DBCS literal continued on beyond 256 characters (bytes) before ending.

User Response: Correct the 'WORD' by shortening it.

A name exceeds one part (is qualified)

Explanation: Regular SQL supports qualified names like 'OWNER.TABLE'. This system does not, outside of the processing of DB2 text.

User Response: Change the name to an unqualified name. Drop the owner or the correlation.

Right parenthesis without preceding left parenthesis

Explanation: Within the SQL syntax, there never is a right (closing parenthesis) that is not preceded by a matching left (opening) parenthesis.

User Response: Correct the syntax.

Invalid word format, starts with a number, but not numeric

Explanation: Names can start with letters, nut not numbers. Names like 001-USER are not supported.

User Response: Chose a different name.

Excessive length for a string with a " or ' - Greater than 256

Explanation: A string constant exceeds the maximum length.

User Response: Break it into more than one string, or correct the text if the terminator was omitted.

Excessive length in a dbcs string

Explanation: The limit of 256 characters applies to a DBCS (graphic) string.

User Response: Correct the text - perhaps by breaking this string into two or more strings.

Missing an ending " or ' within statement

Explanation: The statement ended before 'closing' a string literal.

User Response: Correct the text

Missing an ending SHIFT_IN within a statement

Explanation: In a DBCS string, the statement ended before the shift-out (x'f').

User Response: Correct the text

ABX1502E **Text = *found-text***

Explanation: This is the text associated with the lexical problem.

User Response: Correct the statement.

ABX1503E **Parsing failed - Insufficient memory**

Explanation: There was a failure in acquiring working memory.

User Response: Increase the region size.

ABX1504E **Out of working memory**

Explanation: There was a problem in acquiring working persistent memory.

User Response: Increase the region size.

ABX1505E **Syntax Error - Parentheses don't balance**

Explanation: On parsing a statement, the left parentheses (opening parentheses) were not in balance with the right (closing) parentheses.

ABX1506E **Insufficient memory**

Explanation: While processing a statement, memory was not available.

User Response: Increase the region size.

ABX1507E **Statement Error - Could not classify statement type**

Explanation: The first word in a statement dictates the statement type. This word was not known to the parser.

User Response: Correct the statement. Perhaps the first word is misspelled.

ABX1750E **Maximum number of records exceeded # *records***
- Record type *type*

Explanation: More records of the described type were supplied as input to program ABXD2LEX than could be accommodated

User Response: Review the input and recreate source to fall within the appropriate guidelines.

System Action: DB2 LOAD process terminates.

ABX1831E **Expected text of DD:DDNAME**

Explanation: When parsing a RELATIONSHIP the expected text 'DD:DDNAME' was not found where expected.

User Response: Correct the RELATIONSHIP statement.

ABX1832E **Expected text of DD:DDNAME**

Explanation: When processing a RELATIONSHIP statement, the DD:DDNAME text was expected, but not found.

User Response: Correct the RELATIONSHIP statement.

ABX1833E **A DD name must be from 1 to 8 characters**

Explanation: In processing a RELATIONSHIP statement, the text DD: was not juxtaposed to a valid DDNAME (by size).

User Response: Correct the RELATIONSHIP statement.

ABX1834E **Final comma is invalid**

Explanation: A RELATIONSHIP statement does not end with a comma.

User Response: Correct the relationship statement.

ABX1835E **SOURCE name is excessive in length (*nnn*)**

Explanation: The length of a source name is limited to 32 characters.

User Response: Correct the source name in the RELATIONSHIP statement.

ABX1836E **SOURCE name found-*source-name* is not unique**

Explanation: The source name identified has already been used.

User Response: Choose a different (unique) source name.

ABX1837E **Syntax requires an equal sign after SOURCE name**

Explanation: In processing a RELATIONSHIP statement, the required syntax is *chosen-ource-name* = (the position, length, and data type). This equals sign is not present.

User Response: Correct the RELATIONSHIP statement.

ABX1838E **The attributes in a SOURCE begin with a left parenthesis**

Explanation: The syntax in a RELATIONSHIP statement is chosen-*source-name* = (position, length, data-type). This left parenthesis is missing.

User Response: Correct the RELATIONSHIP statement.

ABX1839E **A length of * is not supported in defining a SOURCE**

Explanation: The asterisk (*) length is only for defining 'the rest of the record' in normal fields. Not in sources. Sources must be fixed length.

User Response: Specify the correct length in the source.

ABX1840E **Expected source data type**

Explanation: In processing a SOURCE within a relationship statement, the *source-name*, starting position and length were found, but the data type (which follows the length) was not found.

User Response: Correct this SOURCE definition.

ABX1841I **The minimum record length in DDNAME must be *nnn***

Explanation: To satisfy the SOURCE and CONTENT definitions for this RELATIONSHIP, the input records for the dd named ddname must be at least *nnn* bytes long.

User Response: No user response is required.

ABX1842I **There are no SOURCES or CONTENTS referenced in DD:DDNAME= - Ignoring it**

Explanation: The SOURCE(S) and CONTENT(s) defined in this DDNAME are never referenced. Consequently this file will be ignored - that is it will not be read and tabularized.

User Response: No user response is required, but there may be an error since this data is not used.

ABX1843E **Problem in reading DD: DDNAME**

Explanation: The file referenced in this RELATIONSHIP could not be read.

User Response: Check that the data set has valid data, and a supported data type that is not VSAM linear or RECFM=U. Also check that the minimum length of the records matches the value displayed in the associated ABX1841I message.

ABX1844E **This file (DD = DDNAME) is not a data RELATIONSHIP file**

Explanation: The file associated with the cited ddname is not valid. Check that the DSORG is supported (not RECFM-U and not an unsupported form of VSAM).

User Response: Correct the file specification.

ABX1845E **This file (dd = DDNAME) is a length of *nnn*, minimum needed = *mmm***

Explanation: The associated ABX1841I for this RELATIONSHIP specifies the minimum required record length needed to extract data. A read record was below this minimum.

User Response: Correct either the RELATIONSHIP statement, or the content of the data set.

ABX1846E **Problem in reading dd: DDNAME**

Explanation: There was an error in reading the cited DDNAME. Check the SYSOUT for any error messages pertinent to this data set. Correct these errors.

User Response: Examine the SYSOUT listing and correct any errors found that relate to this file. Probably the record(s) are too short.

ABX1847E **Problem in reading dd: DDNAME**

Explanation: There was an error in reading the cited DDNAME. Check the SYSOUT for any error messages pertinent to this data set. Correct these errors.

User Response: Examine the SYSOUT listing and correct any errors found that relate to this file. Probably the record(s) are too short.

ABX1848I **As there are no references to SOURCE named *source-name* - bypassing it**

Explanation: A SOURCE was defined but not referenced. Consequently this SOURCE data is not needed and will not be read and tabularized.

User Response: No user response is required if there was no intent to use this data. Otherwise, there is an error elsewhere in referencing data as you intended to use this SOURCE.

ABX1849E **Could not PUT in table for SOURCE *source-name* in memory**

Explanation: The system could not acquire the needed memory to retain the data from this source.

User Response: Increase the region size.

ABX1850I **For SOURCE name *source-name* added *nnn* values (len = *mmm*) to the 'in-list'**

Explanation: A source was defined and referenced. This data was brought into memory (to be used in an SQL in list) and *nnn* unique items of length *mmm* each were tabularized.

User Response: No user response is required.

ABX1851I **For SOURCE name *source-name*, *nnn* duplicates were found**

Explanation: When reading and tabularizing this SOURCE, duplicates (count *nnn*) were found in the source file.

User Response: This is informational only and would only constitute an error if the entries were thought to be unique.

ABX1852E **Could not open table for SOURCE *source-name* in memory**

Explanation: In reading and populating a SOURCE, there was insufficient memory to open the target table.

User Response: Increase the region size.

ABX1861E **On a SOURCE in-list, the lengths don't match *nnn* vs *mmm***

Explanation: For a predicate that references a source, the length of the argument field and the data from the source are incompatible. For instance, a short integer versus an integer *nnn* is the field length, *mmm* the source length.

User Response: Redefine either the source or the field.

ABX1862E **The SOURCE data type does not match the predicate data type;**

Explanation: In a predicate reference to a SOURCE, the data type of the FIELD and the SOURCE are incompatible for comparisons.

User Response: Correct the definition of either the SOURCE or the FIELD.

ABX1870I **Sequence field *field-name* in segment *segment-name* has a changed length**

Explanation: In comparing the SOURCEDBD and TARGETDBD, the length of the sequence field in the named segment is different.

User Response: If the target segment sequence field is not correctly formatted, use a constant (a LITERAL) or VARIABLE to populate the extension, or if contracting, assure the correctness.

ABX1871I **Be advised that you may have problems here**

Explanation: This message is to catch your attention so that you will look to see if the produced data loads correctly. This message accompanies ABX1870I.

User Response: Assure that the desired output data is produced.

ABX1872I **Because Sequence field *sequence-field-name* in Segment *segment-name* is relocated, you will need an explicit select for this segment**

Explanation: The sequence field changes position from the input definition to the output definition. To accomplish this reformatting, you will need an explicit SELECT describing the output record.

User Response: Supply an explicit SELECT that allocates the sequence field to the proper position.

ABX1873I **Sequence field *field-name* in segment *segment-name* has a changed data type**

Explanation: The SOURCEDBD and TARGETDBD definitions for the sequence field have different data type. You are responsible to supply the correct data in the correct data type.

User Response: Ensure that the produced record has correct data in the sequence field.

ABX1874I **Be advised that you may have problems here**

Explanation: This message follows ABX1873I.

User Response: Ensure that the produced record has correct data in the sequence field.

ABX1875I **For Segment *segment-name* a new Sequence field (*field-name*) has been added**

Explanation: In comparing the source and target DBDs, it was found that a sequence field was added for the named segment.

User Response: Ensure that this is as desired and that the sequence field is properly populated.

ABX1876I **Be sure to populate it correctly**

Explanation: This message accompanies ABX1875I above.

User Response: Ensure that this is as desired and that the sequence field is properly populated.

ABX1877I **For segment *segment-name*, the sequence field matched completely in input and output definitions**

Explanation: There is a match in the definition of the sequence field from the source and target DBDs.

User Response: No user response is required.

ABX1880E **Input record *nnn* (for segment *segment-name*) has a key change from the source to the target - no automatic copy will be effected (needs a formatting select)**

Explanation: While processing, it was discovered that a segment has a key change (length or data type) and no explicit formatting SELECT was supplied.

User Response: An explicit SELECT is required to assure population of the key.

ABX1881I **At VSAM write one, length = *nnn***

Explanation: This is a debug message and may have no value to the user.

User Response: No user response is required.

ABX1882I **At VSAM write two, len = *nnn* rec_min = *mmm***

Explanation: This is a debug message and may have no value to the user.

User Response: No user response is required.

ABX1885E **For the segment *segment-name*, the Sequence field (*field-name*) in the produced record for statement statement-number is not the same as the input Sequence field**

Explanation: During execution it was detected that the contents of a sequence field were changed by the (specified) formatting SELECT.

User Response: Modify the 'column-list; of the SELECT to preserve the sequence field's content.

ABX1886W **Forcing Sequence field *field-name* to be the same**

Explanation: Option FORCE is in effect and the sequence field for the output segment is being forced to match the input record's sequence field.

User Response: No user response is required if FORCE is desired.

ABX1887E **Setting the return code to 12. Will only print five of these messages (per select)**

Explanation: The discovered alteration to the sequence field will force a minimum return code of 12.

User Response: Either accept the action of the FORCE option, or change the formatting directives to get the proper results.

Chapter 4. Messages ABX2101--ABX2841

All messages generated by IBM File Export for z/OS have a severity code printed as the last character of the message ID. The severity codes are described in the following table:

Table 3. Error message severity codes

Severity Code	Description
I	Information only. No user action required.
W	Warning message. Results may not be as expected.
E	Error message. Some may be user-correctable, read the User Response to determine the course of action.
S	Severe error -- note the information in the message and contact your support personnel.

ABX2101E Load can be specified as LOAD *(nn)*

Explanation: In parsing the OPTIONS statement, an error was found in the syntax of the LOAD directive.

User Response: Correct the LOAD parameter of OPTIONS statement.

ABX2102E Improper LOAD format

Explanation: The LOAD parameter of the OPTIONS statement is not correctly formatted.

User Response: Correct the LOAD specification of the OPTIONS statement.

ABX2103I Ignoring RESET when not in DLI mode

Explanation: RESET (as a parameter on a VARIABLE) is applicable only to DLI (IMS) processing. Since this is not the chosen process mode, it will be ignored.

User Response: No user response is required.

ABX2104I Ignoring USED when not in DLI mode

Explanation: USED, as a parameter on a VARIABLE, only applies to DLI processing. As the specified mode of processing is NOT DLI (IMS), this parameter is ignored.

User Response: No user response is required.

ABX2105I Ignoring RESET when not in DLI mode

Explanation: RESET as a parameter on a VARIABLE is only effected in DLI or IMS mode. In other forms of operation, it is not applicable. In DB2 mode of operation, variables are reset on the commencement of processing for each TABLE.

User Response: No user response is required.

ABX2106E Increment value is not numeric (found-content)

Explanation: On a variable, only numeric increments are valid. For HEX and CHAR, no increment is allowed as the system always progresses to the next supported value.

User Response: Either remove or correct the increment value.

ABX2107E An increment of found-value is not valid

Explanation: Increments, where allowed (numbers), must be numeric.

User Response: Either remove or correct the increment.

ABX2108E An increment of *nnn* is not supported - 9999 is the max

Explanation: For an integer, the increment is so large that the value may well wrap too soon. To avoid, A lesser value should be used.

User Response: Correct the value.

ABX2109E **In a Variable, 'BY' must be followed with an increment**

Explanation: When parsing a VARIABLE, the syntax following the word BY is the increment. None was found.

User Response: Correct this variable definition.

ABX2110E **Increment value is not numeric (found-value)**

Explanation: Increments where allowed (numbers) must be numeric.

User Response: Correct the increment for this VARIABLE.

ABX2111E **An increment of found-value is not valid**

Explanation: The specified increment (where allowed) is not numeric.

User Response: Correct this increment in the variable statement.

ABX2112I **An increment of found-value is not supported - 1000 is the max - ignoring this, using 1000**

Explanation: The specified increment will make the variable wrap in short order. To prevent this, the specified maximum is used.

User Response: No user response is required.

ABX2113E **In a Variable, 'BY' must be followed with an increment**

Explanation: The syntax of a VARIABLE required an increment if the word BY is specified.

User Response: Correct the declaration.

ABX2114I **Ignoring RESET when not in DLI mode**

Explanation: DLI has the facility for the user to specify RESET explicitly. Variables are reset automatically when traversing down to their first reference. In DB2 mode, reset occurs on the commencement of processing each TABLE. Outside of DLI/IMS processing, user specified reset is not applicable.

User Response: No user response is required.

ABX2115I **Ignoring USED when not in DLI mode**

Explanation: USED as an action of incrementing a VARIABLE is supported only in DLI/IMS mode. Elsewhere, it is ignored.

User Response: No user response is required.

ABX2116I **Ignoring RESET when not in DLI mode**

Explanation: RESET is an option on a variable in DLI mode. In other processing modes, it is not applicable.

User Response: No user response is required.

ABX2117I **Only support an increment of one for a CHAR or HEX LITERAL - ignoring this**

Explanation: HEX and CHAR variables can not have directed increments. They are advanced to the next supported value when required by the system.

User Response: No user response is required.

ABX2118I **Found a character in this CHAR VARIABLE that is not alpha-numeric or blank - setting to blank**

Explanation: In parsing the initial value in a CHAR variable, an unsupported character was found and replaced with a blank.

User Response: No user response is required.

ABX2119I **In this VARIABLE, the first disallowed source value is found-value**

Explanation: In parsing for a VARIABLE, an unsupported value for a character was found and replaced with the minimum value for this data type.

User Response: Either correct the initial value, or accept the system default.

ABX2120I **Maximum unique fields/records using this about *nnn***

Explanation: Given the increment, the maximum number of unique contents before wrapping this value is reflected by *nnn*.

User Response: No user response is required.

ABX2121I **Maximum unique fields/records using this is *nnn***

Explanation: Given the increment, the maximum number of unique contents before wrapping this value is reflected by *nnn*.

User Response: No user response is required.

ABX2122I **Only support an increment of one for a PACKED LITERAL - ignoring this**

Explanation: The system does not support explicit increments for packed data. The system implements plus one internally.

User Response: No user response is required.

ABX2123E **Bad data type in a VARIABLE**

Explanation: The initial value for this VARIABLE does not fit this class of data.

User Response: No user response is required.

ABX2124E **Table 'literals' not found**

Explanation: This message indicates memory corruption.

User Response: Forward the output to your support personnel.

ABX2125E **Failure in the table handler looking up a literal name**

Explanation: The table handler failed to retrieve a value. This is indicative of corrupted memory.

User Response: Forward your output to your support personnel.

ABX2126E **In DLI load mode, must have a TARGETDBD specification**

Explanation: DLI processing always requires a TARGET DBD specification.

User Response: Include a definition of the TARGET DBD in the control syntax.

ABX2127E **In DLI mode, must have a SOURCEDBD specification**

Explanation: In DLI mode of processing a SOURCE DBD must be supplied.

User Response: Include a SOURCE DBD definition in the control syntax.

ABX2128I **As no TARGETDBD was specified, using source = target**

Explanation: As no target DBD was specified (at least not before it was required by other syntax such as a SEGMENT statement), the system is assuming that the TARGET DBD is the same as the SOURCE DBD.

User Response: If this assumption is not correct, supply a TARGETDBD definition.

ABX2129E **TARGETDBD specified too late - already initialized**

Explanation: The TARGETDBD statement was found in the input stream, but after it was required by other statements (such as a FIELD or SEGMENT statement). When it was needed, but not yet supplied, it was assumed that the target and source were identical. Then the TARGETDBD was found (after needed) and the system stops editing and forfeits execution.

User Response: Place the TARGETDBD definition earlier in the syntax stream.

ABX2135W **Warning CHARACTER VARIABLE variable-name has been exhausted, if this is a (part of a) key you may have non-uniques**

Explanation: A CHARACTER variable has wrapped - that is it has reached its maximum allowed value. It will now be set to its lowest allowed value (blanks). Because of this wrapping, there may be duplicate values in the produced output. Were these values to be part of a field requiring uniqueness, a problem may exist in the produced records.

User Response: If this could be a problem, choose a different starting value (blanks) to allow for the maximum combinations.

ABX2136W **Warning HEX VARIABLE variable-name has been exhausted, if this is a (part of a) key you may have non-uniques**

Explanation: This is like the message ABX2135W above, except the variable has hex content. For hex data, the wrap is to binary zeroes.

User Response: If this could be a problem, choose a different starting value (blanks) to allow for the maximum combinations.

ABX2137W **Warning PACKED VARIABLE variable-name has been exhausted if this is a (part of a) key you may have non-uniques**

Explanation: A Packed variable has wrapped - that is it has reached its maximum allowed value. It will now be set to its lowest allowed value (zeroes). Because of this

wrapping, there may be duplicate values in the produced output. Were these values to be part of a field requiring uniqueness, a problem may exist in the produced records.

User Response: If this could be a problem, choose a different starting value (zero or a negative number) to allow for the maximum combinations.

ABX2138W **Warning the VARIABLE variable-name exceeded its maximum of 2147483647, if this is a key you will have non-uniques**

Explanation: An INTEGER variable has wrapped. That is, it has reached its maximum allowed value. It will now be set to its lowest allowed value (zero). Because of this wrapping, there may be duplicate values in the produced output. Were these values to be part of a field requiring uniqueness, a problem may exist in the produced records.

User Response: If this could be a problem, choose a different starting value (zero or a negative number) to allow for the maximum combinations.

ABX2139W **Warning the variable variable-name exceeded its maximum of 32767, if this is a key you will have non-uniques**

Explanation: A SHORT INTEGER variable has wrapped - that is it has reached its maximum allowed value. It will now be set to its lowest allowed value (zero). Because of this wrapping, there may be duplicate values in the produced output. Were these values to be part of a field requiring uniqueness, a problem may exist in the produced records

User Response: If this could be a problem, choose a different starting value (zero or a negative number) to allow for the maximum combinations.

ABX2150E **Word 'date' must be followed by a left parenthesis**

Explanation: In parsing a DATE statement, the text is DATE(*date-name*). This opening (left) parenthesis was not found.

User Response: Correct the syntax of the statement.

ABX2151E **Expected a date name**

Explanation: In parsing a date statement, the expected syntax is DATE (*date-name*) = ... The date-name was not found after the opening (left) parenthesis.

User Response: Correct the syntax of the statement.

ABX2152E **Due to the syntax of FIELD , SCRAMBLE cannot be a date name**

Explanation: An attempt was made to use the word SCRAMBLE as a date-name. This name is not allowed as it would confuse parsing other statements.

User Response: Correct the syntax of the statement. Use another name for this date.

ABX2153E **Syntax requires a right parenthesis after the date name**

Explanation: In parsing a date statement, the expected syntax is DATE (*date-name*) = ... The right (closing) parenthesis after the date name was not found.

User Response: Correct the syntax of the statement.

ABX2154E **Syntax requires an equal sign after date name**

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = ... The equal sign was not found.

User Response: Correct the syntax of the statement.

ABX2155E **Expected the word PACKED or CHAR or CHARACTER**

Explanation: In parsing a date statement, the expected syntax is: DATE (date-name) = data-type 'mask' deltas... The data-type must be PACKED or CHAR (or CHARACTER). This was not the case.

User Response: Correct the syntax of the statement.

ABX2156E **Expected a character literal describing YY MM DD**

Explanation: In parsing a date statement, the expected syntax is: DATE (date-name) = data-type 'mask' deltas... The mask is composed of the characters YY or YYYY, DD (or DDD for Julian dates) and MM with an optional edit character like a dash or slash. The Ys and Ms and Ds can be in any order. This mask must be in a quoted string. This mask was not found.

User Response: Correct the syntax of the statement.

ABX2157E **Had the year designated more than once in the date mask**

Explanation: In a date mask, you can only have the year (YY or YYYY) once.

User Response: Correct the syntax of the statement.

ABX2158E **Had the month designated more than once in the date mask**

Explanation: In a date mask, you can only specify the month (MM) once.

User Response: Correct the syntax of the statement.

ABX2159E **Had the day designated more than once in the date mask**

Explanation: In a date mask, you can only have the day (DD or DDD for Julian) specified once.

User Response: Correct the syntax of the statement.

ABX2160E **Can only have 'z' if packed**

Explanation: The 'z' character, for the sign, is only applicable to a mask describing a packed date field.

User Response: Correct the syntax of the statement.

ABX2161E **Edit character first-edit-character not the same as found character Succeeding-found-character**

Explanation: In a date mask description, any edit character may be used (dash, slash, blank) but you must be consistent. The first and all succeeding edit characters must be specified identically. This edit character need not match the actual data in records - it is a place holder only.

User Response: Correct the syntax of the statement.

ABX2162E **The maximum length for YYYYeMMeDD (any order) is 10**

Explanation: The date mask is excessive in length.

User Response: Correct the syntax of the statement.

ABX2163E **Expected the word DELTAS**

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = data-type 'mask' deltas...The word 'deltas' was not found.

User Response: Correct the syntax of the statement.

ABX2164E **Expected a left parenthesis after the word DELTAS**

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = data-type 'mask' deltas (year sign amount, month sign amount, day sign amount). The left parenthesis after the word 'deltas' was mot found.

User Response: Correct the syntax of the statement.

ABX2165E **Expected YEAR or MONTH or DAY, found *found-word***

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = data-type 'mask' deltas (year sign amount, month sign amount, day sign amount). Within the deltas, MONTH or DAY or YEAR may be in any order (with their sign and amount). However the next word was expected to be one of these.

User Response: Correct the statement.

ABX2166E **Expected a plus sign or a minus sign**

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = data-type 'mask' deltas (year sign amount, month sign amount, day sign amount). Within the deltas, MONTH or DAY or YEAR may be in any order (with their sign and amount). The correct word was found, DAY, MONTH or YEAR , but the expected sign (+ or -) was not found.

User Response: Correct the syntax of the statement.

ABX2167E **Expected a plus sign or a minus sign**

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = data-type 'mask' deltas (year sign amount, month sign amount, day sign amount). Within the deltas, MONTH or DAY or YEAR may be in any order (with their sign and amount). The expected sign (+ or -) was not found.

User Response: Correct the syntax of the statement.

ABX2168E **Number is excessive**

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = data-type 'mask' deltas (year sign amount, month sign amount, day sign amount). Within the deltas, MONTH or DAY or YEAR may be in any order (with their sign and amount). The value of the amount is excessive. For instance a five digit increment is excessive.

User Response: Correct the syntax of the statement.

ABX2169E **Expected an integer value**

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = data-type 'mask' deltas (year sign amount, month sign amount, day sign amount). Within the deltas, MONTH or DAY or YEAR may be in any order (with their sign and amount). The 'amount' is an integer.

User Response: Correct the syntax of the statement.

ABX2170E **Expected an integer value**

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = data-type 'mask' deltas (year sign amount, month sign amount, day sign

amount). Within the deltas, MONTH or DAY or YEAR may be in any order (with their sign and amount). The specified 'amount' is not an integer.

User Response: Correct the syntax of the statement.

ABX2171E **No alterations for MONTH or DAY or YEAR was specified**

Explanation: In parsing a date statement, the expected syntax is DATE (date-name) = data-type 'mask' deltas (year sign amount, month sign amount, day sign amount). Within the deltas, MONTH or DAY or YEAR may be in any order (with their sign and amount). Even though the syntax was acceptable, the expressions did not alter any data, No day or year is to be altered by an amount other than zero.

User Response: Correct the syntax of the statement, or remove this statement.

ABX2172E **The month field must be two characters long**

Explanation: In the date mask of a DATE statement, the MONTH is always two characters (MM).

User Response: Correct the syntax of this statement.

ABX2173E **The day field must be two characters long**

Explanation: In the day mask of a DATE statement, the DAY is always two characters (DD) unless no month field is specified - a Julian date format which can have a three day value (DDD).

User Response: Correct the syntax of the statement.

ABX2174E **The year field must be two or four characters**

Explanation: In the year mask of a DATE statement, the YEAR is always two characters (YY) or four (YYYY).

User Response: Correct the syntax of the statement.

ABX2175E **Date named *date-name*was already defined in statement number *nnn***

Explanation: This is a redundant use of a date-name.

User Response: Either eliminate this statement, or chose a different name for this or the other (colliding) DATE statement.

ABX2176E **Invalid character in the year data : *found-character***

Explanation: At run time, a character other than zero through nine was found in a position described as a YEAR field.

User Response: Either the FIELD position or the date mask is in error, or the file contains invalid date information. Correct the erroneous definition or data.

ABX2177E **Invalid date data, year = *found-data***

Explanation: At run time, data described as containing a year does not. The found content is shown.

User Response: Either the FIELD position or the date mask is in error, or the file contains invalid date information. Correct the erroneous definition or data.

ABX2178E **Invalid character in the day data : *found-data***

Explanation: At run time, a character other than zero through nine was found in a position described as a DAY field.

User Response: Either the FIELD position or the date mask is in error, or the file contains invalid date information. Correct the erroneous definition or data.

ABX2179E **Invalid date data, day = *found-data***

Explanation: At run time, data described as containing a day does not. The found content is shown.

User Response: Either the FIELD position or the date mask is in error, or the file contains invalid date information. Correct the erroneous definition or data.

ABX2180E **Invalid date data, day = *found-data***

Explanation: At run time, data described as containing a day does not. The found content is shown.

User Response: Either the FIELD position or the date mask is in error, or the file contains invalid date information. Correct the erroneous definition or data.

ABX2181E **Invalid character in the month :*found-character***

Explanation: At run time, a character other than zero through nine was found in a position described as a MONTH field.

User Response: Either the FIELD position or the date mask is in error, or the file contains invalid date information. Correct the erroneous definition or data.

ABX2182E **Invalid date data, month = *found-data***

Explanation: At run time, data described as containing a month does not. The found content is shown.

User Response: Either the FIELD position or the date mask is in error, or the file contains invalid date information. Correct the erroneous definition or data.

ABX2183E **Invalid date data, day = *found-data***

Explanation: At run time, data described as containing a day does not. The found content is shown.

User Response: Either the FIELD position or the date mask is in error, or the file contains invalid date information. Correct the erroneous definition or data.

ABX2200I **The date definition *date-name* defined in statement *nnn* is never referenced**

Explanation: A date definition was defined, but never referenced.

User Response: This may indicate an error, if it was intended to reference this definition.

ABX2201E **Incompatible data types between this field and the associated data in the date definition of *date-name***

Explanation: A date was referenced that was described as having a different data type than the definition for this field.

User Response: Either the field definition is wrong, or the referenced date-name is the wrong date-name. Correct the erroneous field or date.

ABX2202E **Incompatible data types between this field and the associated data in the date definition of *date-name***

Explanation: A date was referenced that was described as having a different data type than the definition for this field.

User Response: Either the field definition is wrong, or the referenced date-name is the wrong date-name. Correct the erroneous field or date.

ABX2230E **The date type named (*date-name*) is not defined**

Explanation: A reference was made to date-name which is not defined.

User Response: Correct this reference. Perhaps it is misspelled.

ABX2231E **A statement exceeded 32k, quitting**

Explanation: The maximum length for any control statement is limited to 32K characters comprised of the data between column one and column 72 of each 'card' image.

User Response: Shorten the statement by removing unnecessary trailing or leading blanks on the 'card' images.

ABX2240E **Expected a CONTENT name**

Explanation: In parsing a RELATIONSHIP statement, a content name was expected in the text ... CONTENT (content-name) ...

User Response: Correct the syntax in this statement.

ABX2241E **The CONTENT name is excessive in length**

Explanation: In parsing a RELATIONSHIP statement, a content name was expected in the text ... CONTENT (content-name) ...

User Response: Correct the syntax in this statement.

ABX2242E **The CONTENT name (content-name) is already defined in statement *nnn***

Explanation: This is a redundant use of this content name.

User Response: Correct the syntax in this statement. Chose a unique name for all content names.

ABX2243E **Expected the word 'KEY' to follow the CONTENT name**

Explanation: The syntax of the content portion of the RELATIONSHIP statement is: ... CONTENT content-name KEY = (POSITION, LENGTH, data-type) DATA = (POSITION, LENGTH, data-type) ... The word KEY is expected after the content-name.

User Response: Correct the syntax in this statement.

ABX2244E **Expected the word 'KEY' to follow the CONTENT name**

Explanation: The syntax of the content portion of the RELATIONSHIP statement is: ... CONTENT content-name KEY = (POSITION, LENGTH, data-type) DATA = (POSITION, LENGTH, data-type) ... The word KEY is expected after the content-name.

User Response: Correct the syntax in this statement.

ABX2245E **Expected an equal sign (=) to follow the word 'KEY'**

Explanation: The syntax of the content portion of the RELATIONSHIP statement is: ... CONTENT content-name KEY = (POSITION, LENGTH, data-type) DATA = (POSITION, LENGTH, data-type) ... The equal sign is expected after the word 'KEY'.

User Response: Correct the syntax in this statement.

ABX2246E **Expected an equal sign (=) to follow the word 'KEY'**

Explanation: The syntax of the content portion of the RELATIONSHIP statement is: ... CONTENT content-name KEY = (POSITION, LENGTH, data-type) DATA = (POSITION, LENGTH, data-type) ... The equal sign is expected after the word 'KEY'.

User Response: Correct the syntax in this statement.

ABX2247E **Expected the word 'DATA' after the key attributes**

Explanation: The syntax of the content portion of the RELATIONSHIP statement is: ... CONTENT content-name KEY = (POSITION, LENGTH, data-type) DATA = (POSITION, LENGTH, data-type) ... The word DATA is expected after the key attributes.

User Response: Correct the syntax in this statement.

ABX2248E **Expected the word 'DATA' after the key attributes**

Explanation: The syntax of the content portion of the RELATIONSHIP statement is: ... CONTENT content-name KEY = (POSITION, LENGTH, data-type) DATA = (POSITION, LENGTH, data-type) ... The word DATA is expected after the key attributes.

User Response: Correct the syntax in this statement.

ABX2249E **Expected an equal sign (=) to follow the word 'DATA'**

Explanation: The syntax of the content portion of the RELATIONSHIP statement is: ... CONTENT content-name KEY = (POSITION, LENGTH, data-type) DATA = (POSITION, LENGTH, data-type) ... The equal sign is expected after the word DATA.

User Response: Correct the syntax in this statement.

ABX2250E **Expected an equal sign (=) to follow the word 'DATA'**

Explanation: The syntax of the content portion of the RELATIONSHIP statement is: ... CONTENT content-name KEY = (POSITION, LENGTH, data-type) DATA = (POSITION, LENGTH, data-type) ... The equal sign is expected after the word DATA.

User Response: Correct the syntax in this statement.

ABX2251E **Expected an equal sign (=) after the word DATE**

Explanation: In parsing a RELATIONSHIP statement, a DATE declarative followed the CONTENT definition. This date reference has the syntax of DATE = date-name.

User Response: Correct the syntax in this statement.

ABX2252E **Expected an equal sign (=) after the word DATE:**

Explanation: In parsing a RELATIONSHIP statement, a DATE declarative followed the CONTENT definition. This date reference has the syntax of DATE = date-name.

User Response: Correct the syntax in this statement.

ABX2253E **Expected a date name to follow 'DATE = '**

Explanation: In parsing a RELATIONSHIP statement, a DATE declarative followed the CONTENT definition. This date reference has the syntax of DATE = date-name. This date name was not found.

User Response: Correct the syntax in this statement.

ABX2254E **Incompatible data types between this CONTENT and the associated data in the date definition of date-name**

Explanation: In parsing a RELATIONSHIP statement, a DATE declarative followed the CONTENT definition. This date reference has the syntax of DATE = date-name. The date-name is valid, but the data type described in the CONTENT DATA parameter is not the same as the data type described in the DATE statement referenced.

User Response: Correct the syntax in this statement.

ABX2255E **The date type named (*referenced-date-name*) is not defined**

Explanation: In parsing a RELATIONSHIP statement, a DATE declarative followed the CONTENT definition. This date reference has the syntax of DATE = date-name. The date-name is not defined.

User Response: Correct the syntax in this statement.

ABX2256E **Must have a SOURCE name or the word 'CONTENT'**

Explanation: In parsing a RELATIONSHIP statement, the required syntax is (another) *source-name* or a content definition starting with the word CONTENT.

User Response: Correct the syntax of this statement.

ABX2257I **There are no SOURCE(s) or CONTENT(s) referenced in DD:ddname - Ignoring it**

Explanation: There is a RELATIONSHIP statement using the dd named ddname, but as there are no references to either CONTENT or SOURCE names in that definition, it will not be read or tabularized.

User Response: This may constitute an error. Was the process intended to use this file?

ABX2258I **As there are no references to CONTENT named *content-name*; bypassing it**

Explanation: Although a CONTENT was defined, it is it not referenced. Consequently it will not be tabularized in memory.

User Response: If this is an error, correct the syntax stream. If not, ignore this message.

ABX2259I **Put content into table-name len = *nnn***

Explanation: This is a debugging message and may have no value to the user.

User Response: Ignore this message.

ABX2260I **Put content into table-name len = *nnn***

Explanation: This is a debugging message and may have no value to the user.

User Response: Ignore this message.

ABX2261E **Could not PUT in table for CONTENT *content-name* in memory**

Explanation: There is insufficient memory to tabularize this content.

User Response: Increase the region size.

ABX2262I **For CONTENT name *content-name* added *nnn* values (len = *mmm*) to the keys and data table**

Explanation: The data for a CONTENT was successfully tabularized

User Response: No response is required.

ABX2263I **For CONTENT name *content-name*, *nnn* duplicates were found**

Explanation: In populating the named CONTENT, *nnn* duplicate key values were processed. The first of the redundant entries is kept (by key), the rest are discarded.

User Response: If there is reason to believe that the keys are unique, check that the definition (position and length) is correct.

ABX2264E **Could not open table for CONTENT*content-name* in memory**

Explanation: There is likely no memory available. This message normally follows an 'out of memory' message.

User Response: Increase the region size.

ABX2265E **Could not initialize the CONTENT table for *content-name***

Explanation: This is most likely a memory availability problem.

User Response: Increase the region size. If this does not cure the problem, forward your SYSOUT to your support personnel.

ABX2270E **In CONTENT *content-name*, the data type is not known**

Explanation: The data type is not defined.

User Response: Correct this statement.

ABX2271E **In CONTENT *content-name*, the data type is not known**

Explanation: The data type of the key or data portion of the content is not known.

User Response: Correct this statement.

ABX2272I **Found content based on value of 'key' *key-value***

Explanation: This is a debugging message and may have no value to the user.

User Response: Ignore this message.

ABX2273E **In CONTENT *content-name*, the data type is not known**

Explanation: The data type for a CONTENT expression is not a known type. This can be either the KEY or the DATA portion of the expression.

User Response: Correct this statement.

ABX2274E **Cannot use a length of star (field *field-name*) in a CONTENT expression**

Explanation: The FIELD to use as the KEY is defined with a length indicator of asterisk, meaning the rest of the record. Keys with which to find CONTENT must be fixed length.

User Response: Correct the CONTENT expression.

ABX2275E **Cannot use a 'FROM END' (field *field-name*) position in CONTENT expression**

Explanation: Because the location varies with variable length records, the data to match a KEY in a content expression must have a fixed location.

User Response: Alter this content expression.

ABX2276E **The data type of the FIELD (*field-name*) does not match the data type of the key in the CONTENT (*content-name*)**

Explanation: The content expression is using as an argument for matching the key, a different data-type than the content key. An example would be using an integer versus character data.

User Response: Correct this expression if you can. Matches are only supported on like data-types.

ABX2277E **The length of the FIELD *field-name* (*nnn*) does not match the length of the key (*mmm*) in the CONTENT (*content-name*)**

Explanation: There is a size mismatch in using a content lookup.

User Response: Be sure the data lengths are compatible.

ABX2290E **Source db2 name is excessive in length - max is four characters**

Explanation: The name of the DB2 subsystem is limited to four characters.

User Response: Correct the name of the subsystem that is the DB2 to which we will communicate.

ABX2291E **Syntax error - format is SOURCEDB2(*subsystem-name*)**

Explanation: The format of the SOURCEDB2 syntax is as above.

User Response: Correct the SOURCEDB2 statement.

ABX2292E **TARGETDB2 name is excessive in length - max is four characters**

Explanation: The subsystem name is only supported with a maximum length of four.

User Response: Correct the SOURCEDB2 statement.

ABX2293E **Syntax error - format is TARGETDB2(*subsystem-name*)**

Explanation: The required syntax is as above.

User Response: Correct the TARGETDB2 statement.

ABX2294E **Expected text to follow the word SORT**

Explanation: The format of the SORT phrase is: SORT (column, ASC/DESC {, column ASC/DESC}). This statement ended with the word SORT.

User Response: Correct the syntax of the statement.

ABX2295E **Expected a left parenthesis after the word SORT**

Explanation: The format of the SORT phrase is: SORT (column, ASC/DESC {, column ASC/DESC}). There is a required left parenthesis after the word SORT.

User Response: Correct the syntax of the statement.

ABX2296E **Expected something to follow the left parenthesis in the sort clause**

Explanation: The format of the SORT phrase is: SORT (column, ASC/DESC {, column ASC/DESC}). The statement ended with the left parenthesis.

User Response: Correct the syntax of the statement.

ABX2297E **Sort column name is too long**

Explanation: In a SORT phrase, the column name is excessive in length. Column names are limited to 30 characters at this time.

User Response: Correct the syntax of the statement.

ABX2298E **Expected a right parenthesis to end the sort declaration**

Explanation: The format of the SORT phrase is: SORT (column, ASC/DESC {, column ASC/DESC}). The right parenthesis was not found where required,

User Response: Correct the syntax of the statement.

ABX2299E **Bad SORT syntax in the column list**

Explanation: The format of the SORT phrase is: SORT (column, ASC/DESC {, column ASC/DESC}). The word SORT appeared, but there is an error in the phrase.

User Response: Correct the syntax of the statement.

ABX2300E **Expected the word 'CHILD' (following the word 'AND')**

Explanation: In parsing a table expression a CHILD phrase was processed and the next text was AND. The only allowed use of AND at this point is AND CHILD OF ...

User Response: Correct the syntax of the statement.

ABX2301E **Expected the word 'OF' following the word 'CHILD')**

Explanation: The syntax is CHILD OF *parent-reference-id*. The word OF therefore must follow the word CHILD.

User Response: Correct the syntax of the statement.

ABX2302E **Expected the name of the parent following the word 'OF'**

Explanation: The syntax of the CHILD phrase is CHILD OF *parent-reference-id*. Therefore the required syntax must specify the parent.

User Response: Correct the syntax of the statement.

ABX2303E **The reference-id of the parent table is excessive in length**

Explanation: The short name or reference identified of the parent is limited to 32 characters.

User Response: Correct the syntax of the statement.

ABX2304E **The reference-id of the parent (*parent-reference-id*) is not defined**

Explanation: The referred-to parent is not defined. This is either a sequence of definition problem or a spelling issue.

User Response: Correct the syntax of the statement.

ABX2305E **Expected the word 'WHERE' after the parent name**

Explanation: In the CHILD OF declarative, the relationship must be specified. That is the matching column(s) in the child and parent. The syntax is CHILD OF *parent-reference-id* WHERE *parent-reference-id*.column = *parent-reference-id*.column. Or WHERE *this-reference-id*.column,column,... = *parent-reference-id*.column,...

User Response: Correct the syntax of the statement.

ABX2306E **Expected text like *this-reference-id*.column = ...**

Explanation: See the above message (ABX2305E) for the allowed syntax.

User Response: Correct the syntax of the statement.

ABX2307E **Bad format of *this reference-id*.column**

Explanation: See message ABX2305E for the allowed syntax. The found text was not *this-reference-id*.column.

User Response: Correct the syntax of the statement.

ABX2308E **Expected a period after *this-reference-id***

Explanation: See message ABX2305E for the allowed syntax. The period that is required after *this-reference-id* is missing.

User Response: Correct the syntax of the statement.

ABX2309E **Expected a period after *this-reference-id***

Explanation: See message ABX2305E for the required syntax. The required period is missing.

User Response: Correct the syntax of the statement.

ABX2310E **Column name exceeds maximum length of 30**

Explanation: The Column name found in the parent-child relationship is excessive in length.

User Response: Correct the syntax of the statement.

ABX2311E **Expected the column name**

Explanation: In the parent-child relationship, an expected columns name was expected, but was not present.

User Response: Correct the syntax of the statement.

ABX2312E **The expected text is *this-reference-id*.column - *found-word* is not this table's reference-id**

Explanation: To distinguish the child-parent reference, the qualified for the left argument must be the child's reference-id. The right argument must be qualified by the parent's reference-id.

User Response: Correct the syntax of the statement.

ABX2313E **Expected an '=' or the characters 'EQ'**

Explanation: In the CHILD OF declarative, the relationship must be specified. That is the matching column(s) in the child and parent. The syntax is CHILD OF parent-reference-id WHERE *this-reference-id*.column = *parent-reference-id*.column. Or WHERE *this-reference-id*.column,column,... = *parent-reference-id*.column,column,... The equal mark was not found.

User Response: Correct the syntax of the statement.

ABX2314E **Expected an equal sign (=) or the characters 'EQ' after the *this-reference-id*.column**

Explanation: In the CHILD OF declarative, the relationship must be specified. That is the matching column(s) in the child and parent. The syntax is CHILD OF *parent-reference-id*WHERE *this-reference-id*.column = *parent-reference-id*.column. Or WHERE *this-reference-id*.column,column,... = *parent-reference-id*.column,column,... This required equal sign was not found.

User Response: Correct the syntax of the statement.

ABX2315E **Expected the text parent.column after '=' or 'EQ'**

Explanation: In the CHILD OF declarative, the relationship must be specified. That is the matching column(s) in the child and parent. The syntax is CHILD OF *parent-reference-id*WHERE *this-reference-id*.column = *parent-reference-id*.column. Or WHERE *this-reference-id*.column,column,... = *parent-reference-id*.column,column,... The right argument was not as described.

User Response: Correct the syntax of the statement.

ABX2316E **Bad format of parent.column**

Explanation: In the CHILD OF declarative, the relationship must be specified. That is the matching column(s) in the child and parent. The syntax is CHILD OF *parent-reference-id*WHERE *this-reference-id*.column = *parent-reference-id*.column. Or WHERE *this-reference-id*.column,column,... = *parent-reference-id*.column,column,... The *parent-reference-id*.column is not in the required format.

User Response: Correct the syntax of the statement.

ABX2317E **Expected a period after the parent table name**

Explanation: In the CHILD OF declarative, the relationship must be specified. That is the matching column(s) in the child and parent. The syntax is CHILD OF *parent-reference-id*WHERE *this-reference-id*.column = *parent-reference-id*.column. Or WHERE *this-reference-id*.column,column,... = *parent-reference-id*.column,column,... The period expected in qualifying the column in the parent was not found.

User Response: Correct the syntax of the statement.

ABX2318E **Expected a period after the source table owner**

Explanation: In parsing the SOURCE phrase in the TABLE statement, the owner.table format is missing the required period.

User Response: Correct the syntax of the statement.

ABX2319E **Parent column name exceeds maximum length of 30**

Explanation: In the current implementation, no column exceeding 30 characters in the name is supported in that the limit in an SQLDA is currently 30.

User Response: Correct the syntax of the statement.

ABX2320E **Expected the PARENT table column name**

Explanation: : In the CHILD OF declarative, the relationship must be specified. That is the matching column(s) in the child and parent. The syntax is CHILD OF *parent-reference-id*WHERE *this-reference-id*.column = *parent-reference-id*.column. Or WHERE *this-reference-id*.column,column,... = *parent-reference-id*.column,column,... The second argument (*parent-reference-id*.column) is incorrect.

User Response: Correct the syntax of the statement.

ABX2321E **The parent in parent.column text is not the named parent**

Explanation: In the parent-child relationship WHERE phrase, the entered *parent-reference-id* is not the named parent.

User Response: Correct the syntax of the statement.

ABX2322E **Duplicate definition of parent and column relationship**

Explanation: In the parent-child relationship, there is a redundant definition of the column(s) in the relationship.

User Response: Correct the syntax of the statement.

ABX2323E **Expected the word 'WHEN' to follow the word 'ONLY'**

Explanation: Then syntax for matching or pairing records is ONLY WHEN column {NOT} IN (source) {AND column {NOT} IN (source)}. Consequently the word ONLY must be followed by the word WHEN.

User Response: Correct the syntax of the statement.

ABX2324E **Expected a column name after 'ONLY WHEN' or 'NOT WHEN' or 'AND'**

Explanation: Then syntax for matching or pairing records is ONLY WHEN column {NOT} IN (source) {AND column {NOT} IN (source)}. The found word did not meet this syntax.

User Response: Correct the syntax of the statement.

ABX2325E **The column name is excessive in length**

Explanation: Then syntax for matching or pairing records is ONLY WHEN column {NOT} IN (source) {AND column {NOT} IN (source)}. The maximum length of the name of the column is thirty characters.

User Response: Correct the syntax of the statement.

ABX2326E **Expected the word 'IN' in ONLY WHEN COL IN (SOURCE)**

Explanation: Then syntax for matching or pairing records is ONLY WHEN column {NOT} IN (source) {AND column {NOT} IN (source)}. The word IN was not found.

User Response: Correct the syntax of the statement.

ABX2327E **Expected a left parenthesis after 'IN'**

Explanation: Then syntax for matching or pairing records is ONLY WHEN column {NOT} IN (source) {AND column {NOT} IN (source)}. The left parenthesis required after the word IN was not found.

User Response: Correct the syntax of the statement.

ABX2328E **Expected a SOURCE name - ran out of statement**

Explanation: Then syntax for matching or pairing records is ONLY WHEN column {NOT} IN (source) {AND column {NOT} IN (source)}. The statement ended with a left parenthesis.

User Response: Correct the syntax of the statement.

ABX2329E **Source *source-name* is not defined**

Explanation: Then syntax for matching or pairing records is ONLY WHEN column {NOT} IN (source) {AND column {NOT} IN (source)}. The cited source name is not defined in a RELATIONSHIP statement.

User Response: Correct the syntax of the statement.

ABX2330E **This has a redundant SOURCE**

Explanation: The syntax for matching or pairing records is ONLY WHEN column {NOT} IN (source) {AND column {NOT} IN (source)}. The same SOURCE name pairing was found in the text.

User Response: Correct the syntax of the statement.

ABX2331E **Expected a left parenthesis after the word TABLE**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ...The Left (opening) parenthesis preceding the reference-id was not found.

User Response: Correct the syntax of the statement.

ABX2332E **REFERENCE-ID for the table exceeds 32 bytes**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The reference identity is excessive in length.

User Response: Correct the syntax of the statement.

ABX2333E **Expected a reference-id - not found**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ...The reference identity is missing.

User Response: Correct the syntax of the statement.

ABX2334E **Expected Right parenthesis after the SHORT-NAME or REFERENCE-ID**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE(OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ...The right (closing) parenthesis after the reference identity is missing.

User Response: Correct the syntax of the statement.

ABX2335E **Expected the word SOURCE**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ...The word SOURCE is required in the syntax.

User Response: Correct the syntax of the statement.

ABX2336E **Expected the word SOURCE**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The word SOURCE is required.

User Response: Correct the syntax of the statement.

ABX2337E **Expected a left parenthesis after the word SOURCE**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The required left parenthesis is missing, or misplaced.

User Response: Correct the syntax of the statement.

ABX2338E **Source table owner name exceeds maximum length of 128**

Explanation: In DB2 prior to version 8, the limit is eight.

User Response: Correct the syntax of the statement.

ABX2339E **Expected the SOURCE table owner**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The OWNER.TABLE format is incorrect.

User Response: Correct the syntax of the statement.

ABX2340E **Bad format of owner.table**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The OWNER.TABLE format is incorrect.

User Response: Correct the syntax of the statement.

ABX2341E **Expected a period after the source table owner**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The period following the OWNER is missing.

User Response: Correct the syntax of the statement.

ABX2342E **Expected a period after the target table owner**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The period following the OWNER is missing.

User Response: Correct the syntax of the statement.

ABX2343E **Source table name exceeds maximum length of 128**

Explanation: Prior to DB2 version 8, the limit is eighteen characters.

User Response: Correct the syntax of the statement.

ABX2344E **Expected the SOURCE table name**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... Expected the OWNER.TABLE combination of the DB2 source table(s),

User Response: Correct the syntax of the statement.

ABX2345E **Expected a right parenthesis after the source table name**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The right parenthesis following the SOURCE (OWNER.TABLE ...) was not found.

User Response: Correct the syntax of the statement.

ABX2346E **Expected the word SOURCE**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The word TARGET was not found where it is required.

User Response: Correct the syntax of the statement.

ABX2347E **Expected the word TARGET**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The word TARGET was not found where it was expected.

User Response: Correct the syntax of the statement.

ABX2348E **Expected a left parenthesis after the word TARGET**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The left (opening) parenthesis following TARGET is missing or misplaced.

User Response: Correct the syntax of the statement.

ABX2349E **Target table owner name exceeds maximum length of 128**

Explanation: Prior to DB2 Version 8, the length limit is eight.

User Response: Correct the syntax of the statement.

ABX2350E **Expected the TARGET table owner**

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The OWNER of the TARGET table was not found.

User Response: Correct the syntax of the statement.

ABX2351E Bad format of owner.table

Explanation: OWNER.TABLE can be expressed as OWNER.TABLE or OWNER . TABLE.

User Response: Correct the syntax of the statement.

ABX2352E Expected a period after the target table owner

Explanation: The format of the TARGET (OWNER.TABLE) is missing the period.

User Response: Correct the syntax of the statement.

ABX2353E Expected a period after the target table owner

Explanation: The period in TARGET (OWNER.TABLE) is missing.

User Response: Correct the syntax of the statement.

ABX2354E Target table name exceeds maximum length of 128

Explanation: Prior to DB2 version 8, the name is limited to 18 characters.

User Response: Correct the syntax of the statement.

ABX2355E Expected the Target table name

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The TARGET (OWNER.TABLE) is missing the TABLE.

User Response: Correct the syntax of the statement.

ABX2356E Expected a right parenthesis after the target table name

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The right parenthesis following the TABLE in TARGET (OWNER.TABLE) is missing.

User Response: Correct the syntax of the statement.

ABX2357E Expected the text of 'DD:ddname'

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The DD:DDNAME text was not found.

User Response: Correct the syntax of the statement.

ABX2358E Expected the text of 'DD:ddname'

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ...The DD:DDNAME text was not found.

User Response: Correct the syntax of the statement.

ABX2359E Expected the text of 'DD:ddname'

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... The DD:DDNAME text was not found.

User Response: Correct the syntax of the statement.

ABX2360E Maximum length of a DDNAME is eight characters

Explanation: In expressing a table statement, the syntax is: TABLE (reference-id) SOURCE (OWNER.TABLE {, OWNER.TABLE ...}) TARGET (OWNER.TABLE) DD:DDNAME ... In the DD:DDNAME text, the length of the data definition name is limited to eight characters.

User Response: Correct the syntax of the statement.

ABX2361E Unexpected text (*found-text*)

Explanation: In parsing a TABLE statement, unexpected text was found.

User Response: Correct the syntax of the statement.

ABX2362E The table reference-id reference-identity is already defined in statement number *nnn*

Explanation: In this table statement, a reference identity was specified that had already been used in a previous TABLE statement. Reference identities must be unique.

User Response: Change this reference identity or the reference identity of the prior redundant definition.

ABX2363E The table defined in statement #*nnn* already uses dd:ddname

Explanation: The ddname chosen to contain the exported table data has already been used for a prior table statement. These data definitions must be unique for each table.

User Response: Choose a different DDNAME.

ABX2364E **Error reading input file from DD ddname - Record read is *nnn* bytes, minimum is *mmm***

Explanation: In reading a file involved in a RELATIONSHIP the actual record(s) is too short for the use of the SOURCE(s) and CONTENT(s) defined on that file.

User Response: Either correct the SOURCE definitions or the CONTENT definitions in this relationship, or perhaps you are referencing the wrong data set.

ABX2375E **You have already set the run type**

Explanation: In processing an OPTIONS statement, a run type declarative was found. This declarative is redundant as the run type was already set.

User Response: Correct the syntax of this statement.

ABX2381E **LOAD is not supported in DB2 processing mode**

Explanation: The OPTIONS (... LOAD ...) was specified, but the run type is DB2. The system does not support LOAD in the DB2 processing mode.

User Response: Correct the syntax of this statement.

ABX2382E **This SKIP must be preceded by a TABLE declarative (in DB2 processing)**

Explanation: An orphan SKIP statement was found. SKIP is a sub-statement of a TABLE statement in DB2 processing mode and must be preceded by a TABLE statement.

User Response: Correct the syntax of this statement.

ABX2383E **This LIMIT must be preceded by a TABLE declarative (in DB2 processing)**

Explanation: An orphan LIMIT statement was found. LIMIT is a sub-statement of a TABLE statement in DB2 processing mode and must be preceded by a TABLE statement.

User Response: Correct the syntax of this statement.

ABX2384I **This statement type does not apply to DB2 processing - ignoring it**

Explanation: The above statement (such as a SEGMENT statement) was found in the syntax input stream. This statement type has no meaning in DB2 mode of operation and will be ignored.

User Response: If you have the correct processing mode declared, ignore this message.

ABX2385I **This statement type is only applicable to DB2 processing - ignoring it**

Explanation: In a non-DB2 mode of processing a DB2 statement (like TABLE or COLUMN) was found. It has no value in the declared mode of processing.

User Response: If have declared the correct processing mode, ignore this message.

ABX2386E **This SELECT must be preceded by a TABLE declarative (in DB2 processing)**

Explanation: In DB2 processing mode, the SELECT is logically a child of a TABLE statement. A SELECT was found before a valid TABLE statement and as such is an orphan.

User Response: Correct the location of this statement, or correct the prior TABLE statement.

ABX2387E **There is already a defined select for this table - you may chose to define this table more than once, but you will need two separate table reference-ids**

Explanation: In DB2 processing there is but one SELECT per TABLE statement. If a SELECT is absent, the default is SELECT *.

User Response: Remove this or the prior SELECT. If necessary, you may define an additional TABLE.

ABX2388E **There must be a TABLE statement preceding this THEN INSERT INTO DD:ddname statement (in DB2 mode)**

Explanation: In DB2 processing a TABLE statement must precede a THEN INSERT INTO DD:DDNAME statement as this statement is a child of the TABLE statement and is effected on each row fetched from the 'owning' table. This THEN INSERT is an orphan statement.

User Response: Remove or relocate this statement. Possibly correct an erroneous prior TABLE statement.

ABX2389E **THEN INSERT ... must immediately follow a select or another THEN INSERT ...or a TABLE**

Explanation: A found THEN INSERT is out of position.

User Response: Remove or relocate this statement. Possibly repair a prior TABLE statement.

ABX2400E **A column definition must be preceded with a TABLE declarative**

Explanation: A COLUMN statement is an augmentation to a TABLE statement. As such, it must follow a valid prior TABLE statement.

User Response: Remove or relocate this COLUMN statement. Possibly a prior TABLE statement is in need of correction.

ABX2401E **Expected a left parenthesis after the word 'COLUMN'**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The left parenthesis after the COLUMN is missing.

User Response: Correct the syntax of this statement.

ABX2402E **Expected the column name after COLUMN (, found end of text**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The statement ended after COLUMN (.

User Response: Correct the syntax of this statement.

ABX2403E **The column name is excessive in length : only 30 bytes currently allowed due to restrictions in the SQLDA**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. This column name exceeds 30 bytes.

User Response: Correct the syntax of this statement.

ABX2404E **Expected a right parenthesis after the column name**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The right parenthesis following the column name is missing.

User Response: Correct the syntax of this statement.

ABX2405E **Expected a left parenthesis after the word ORDINAL**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The optional ORDINAL (signifying the nth column in the select column list) must be followed by a left (opening) parenthesis. This was not found.

User Response: Correct the syntax of this statement.

ABX2406E **Expected an integer value after the ORDINAL (**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The right (closing) parenthesis is missing.

User Response: Correct the syntax of this statement.

ABX2407E **The ordinal value is greater than 5 in length**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The ordinal number is excessive in scale.

User Response: Correct the syntax of this statement.

ABX2408E **Invalid value in the ORDINAL integer**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. This ordinal value is invalid.

User Response: Correct the syntax of this statement.

ABX2409E **Expected a right parenthesis after the word ORDINAL number**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The closing (right) parenthesis following the ordinal vales was not found.

User Response: Correct the syntax of this statement.

ABX2410E **Don't know the word *found-word* (first 32 bytes)**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. In parsing this COLUMN statement, an unrecognized word was found.

User Response: Correct the syntax of this statement.

ABX2411E **Expected more text, such as NEW, OMIT, SCRAMBLE, SOURCE or DATE-NAME found statement end**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new

{NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The statement ended prematurely.

User Response: Correct the syntax of this statement.

ABX2412E **Expected an equal sign (=) after the word SOURCE**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. If SOURCE (for adding or replacing data) is specified it must be followed by an equal sign.

User Response: Correct the syntax of this statement.

ABX2413E **Expected a content-name after the expression 'source = '**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The equal sign following the word SOURCE was not found.

User Response: Correct the syntax of this statement.

ABX2414E **The content name is not known (content-name-word)**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. This content-name is not defined.

User Response: Correct the syntax of this statement or define the CONTENT in a RELATIONSHIP statement.

ABX2415E **Expected a left parenthesis after the content-name**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The SOURCE = CONTENT-NAME was parsed, but the left parenthesis to introduce the key-match column value is missing.

User Response: Correct the syntax of this statement.

ABX2416E **The based-on-column name exceeds maximum length of 128**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. The name of a column to constitute the matching for CONTENT retrieval is excessive in length. The SQLDA definition restricts this name to thirty characters.

User Response: Correct the syntax of this statement.

ABX2417E **Expected the based-on column name**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. This column name for matching against a CONTENT entry is missing.

User Response: Correct the syntax of this statement.

ABX2418E **Expected a right parenthesis after the content-name**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = *content-name*(based-on-*column-name*) { omit }{scramble } {date-name}. The right (closing) parenthesis following the column which is to be matched for content is missing.

User Response: Correct the syntax of this statement.

ABX2419E **Syntax error, word *found-word* not identified**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. A word was found that did not match this syntax.

User Response: Correct the syntax of this statement.

ABX2420E **NEW and OMIT conflict**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. Defining a NEW column and then omitting it seems to be an error.

User Response: Correct the syntax of this statement.

ABX2421E **NEW requires a SOURCE**

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. To specify a NEW column (not coming from DB2), the only possible source for the data is from a CONTENT match. Therefore a SOURCE is required if NEW is specified.

User Response: Correct the syntax of this statement.

ABX2422E **Found no use for the word *found-word***

Explanation: The syntax of the COLUMN statement is: COLUMN (*column-name*) {ordinal(nn) {new {NULLABLE } } {source = content-name (based-on-*column-name*) { omit }{scramble } {date-name}. After the required part of the statement, an apparently optional word was found. There is no definition of what to do with this word.

User Response: Correct the syntax of this statement.

ABX2423E **This same COLUMN name was used in statement #*nnn***

Explanation: Within the scope of this same table, there is a COLUMN statement for the same column. This is not allowed as all options for a column must be specified on one COLUMN statement. Either this *column-name* is incorrect, or the statement is redundant.

User Response: Correct the syntax of this statement.

ABX2430I **The field to match in the CONTENT expression (*word-within-parentheses*) is not known - treating as non-content expression**

Explanation: A phrase like word-one (word-two) which could have been a content(key) format was parsed but no matching content was found.

User Response: Correct the syntax of this statement.

ABX2450E **Either no tables were declared or the SOURCEDB2 was not declared or the connection to DB2 failed**

Explanation: In parsing the syntactical input in OPTIONS (... DB2 ...) mode, no correct TABLE statements were found. With no valid TABLE statements, processing cannot continue.

User Response: Correct the syntax, or resolve why no connection to DB2 was permitted.

ABX2460I **Formatting data in DEDB format**

Explanation: The processing is in IMS mode with the type of data being DEDB data.

User Response: No response is required.

ABX2470E **Connecting to DB2 *(subsystem-id)*failed**

Explanation: The program could not connect to the SOURCEDB2. The failure reason will precede this message.

User Response: Resolve the connection issue shown in a prior message.

ABX2471I **The SOURCEDB2 *(subsystem-id)* is version version-identifier**

Explanation: The connection to the SOURCEDB2 was successful. The release identifier is shown.

User Response: No action is required if this is the correct SOURCEDB2.

ABX2472E **Expected the word NULLS following the word ACCEPT**

Explanation: In a column matching, if the SOURCE within the TABLE statement's phrase ONLY WHEN IN with which to match is NULL, there is no way to match. ACCEPT NULLS says to consider the row matched if the column on which to match is NULL.

User Response: Correct this syntax.

ABX2473E **Must have a SOURCEDB2 statement preceding the first table statement - and the connect must be successful**

Explanation: The sequence of statement affects the parsing. The SOURCEDB2 statement must occur prior to the first TABLE statement in processing mode DB2.

User Response: Supply the required SOURCEDB2 statement, or sequence the statements in the correct order.

ABX2474E **The select defined in statement *nnn* for the table defined in statement *mmm* is in error**

Explanation: The select statement following its owning table statement would not prepare (in DB2).

User Response: Correct this SELECT statement.

ABX2475E **The internally generated SELECT * from source-table for the table defined in statement *nnn* failed to prepare**

Explanation: For a given TABLE statement, no select was supplied and the system generated a SELECT * for this table(s). This generated statement was invalid and would not prepare. Most likely one of the named tables or named owner is unknown, or the user is not authorized to read these tables. See the messages emitted by the PREPARE.

User Response: Correct the OWNER or the TABLE names in the associated TABLE statement.

ABX2476S **The data type for column *column-name* in the table defined in statement *nnn* is a LARGE OBJECT, which is not**

supported in this release - it will be omitted in the output

Explanation: This release does not support large objects (LOB's). The system will bypass exporting this column.

User Response: This utility will not fit the need at this time.

ABX2477E **The produced record for the table in statement *nnn* is too long at *mmm* bytes**

Explanation: In exporting a row and possibly replacing or adding CONTENT, the produced record exceeds 32,767 bytes and is ineligible to be loaded by the load utility.

User Response: See if the definitions are correct on what is to be exported. If the need is for a row and the null indicators to exceed 32K, perhaps the target table definitions should change.

ABX2478E **The COLUMN (*column-name*) used in the sort declarative for the table statement (number *nnn*) is not in the select**

Explanation: To be able to sort on a column, it must be included in the produced record.

User Response: Correct either this SORT declarative or include the column in the produced export record.

ABX2479E **The COLUMN (*column-name*) used in the CHILD declarative for the table statement number *nnn* is not in the select**

Explanation: To effect the parent-child matching, the column to match on must be fetched from DB2. It is not necessary to pass this column on to the target table as an OMIT parameter of the COLUMN statement can be used.

User Response: Correct the SELECT statement.

ABX2480E **For statement *nnn* the data type of the parent's child-parent relationship column (*column-name*) is different than the child's column data type (*column-name*)**

Explanation: The match of parent-child columns depends on each column being the same data type as the system will not convert data to execute a match.

User Response: See if the definitions are correct. If so, this utility cannot be used to maintain the RI need.

ABX2481E **For statement *nnn* the data length of the parent's child-parent relationship column (*column-name*) is different than the child's column data length (*column-name*)**

Explanation: To match parents and children, the data type of the associated columns must be the same, and except for character data (CHAR and VARCHAR), the lengths must match.

User Response: See that the definitions are correct and match. If so, you may have to remove this constraint in order to run.

ABX2482E **The COLUMN (*column-name*) used by a CHILD declarative for the table statement number *nnn* is not in the select list for this parent**

Explanation: In a parent-child relationship it is necessary to extract the parental columns and child columns from DB2 for matching. If these extracted columns are not wanted in the TARGET table, they may be omitted using the COLUMN statement.

User Response: Correct the SELECT associated with the parent TABLE definition.

ABX2483E **The COLUMN (*column-name*) used by an IN predicate declarative for the table statement number *nnn* is not in the select list for this table**

Explanation: The TABLE definition includes an ONLY WHEN *column-name* IN (*source-name*). This *column-name* is not in the SELECT for this table (SELECT ... this-*column-name* ... FROM). It must be fetched from DB2 in order to effect the matching. It need not be included in the produced TARGET table as it can be excluded from the LOAD with the OMIT option of the COLUMN statement.

User Response: Include this column in the SELECT statement for this TABLE. Effect any needed omission as necessary.

ABX2484E **OPTIONS (DB2, ...) was specified, but no tables statements were declared**

Explanation: In DB2 processing, at least one valid TABLE statement must be included. Perhaps there is an error in included TABLE statement(s), or they were not included in the control stream, or the processing mode is wrong.

User Response: Correct the control syntax stream.

ABX2490E **DB2 subsystem is shutting down**

Explanation: The SOURCED2 subsystem to which we are connected is requesting a shut down. The utility will terminate processing.

User Response: Resubmit the run whenever the subsystem is gain available. You may have to delete any DISP=(NEW,CATLG) data sets created in this run.

ABX2491E DB2 is requesting a forced shutdown

Explanation: The utility was notified that the DB2 is quitting. The utility terminates.

User Response: Resubmit this run when the DB2 is again available. You may need to delete any DISP=(NEW,CATLG) data sets created by this execution.

ABX2492E DB2 CAF interface failed unexpectedly for function : *function-being-performed*

Explanation: This message will be associated with additional messages created by DB2.

User Response: Correct whatever DB2 indicates to be the problem.

ABX2493E CAF release level mismatch

Explanation: The loaded CAF (call attach facility) is a mismatch for the DB2 subsystem named in the SOURCEDB2. The STEPLIB does not reference the correct DSN library.

User Response: Correct the STEPLIB statement. Be sure that the correct DSN library is referenced.

ABX2494E CAF wants us to retry

Explanation: The CAF signaled that we need to retry. This is taken as an error and processing terminates.

User Response: Rerun the utility after deleting the output data sets.

ABX2495E CAF failed, unknown reason

Explanation: The CAF returned an unknown (at least not-provided-for) status.

User Response: Analyze other messages and see if a retry is called for. If you retry, delete the output data sets (DISP=(NEW,CATL)).

ABX2496E DB2 status information

Explanation: DB2 output as formatted by DSNTIAR is presented here.

User Response: Examine the return code, reason code and messages from DB2 to establish what can be done. Effect what is required.

ABX2497E DB2 is down

Explanation: The SOURCEDB2 specified subsystem is known to the operating system, but it is currently down.

User Response: Rerun this step after the DB2 subsystem returns to normal operation. Delete any output data sets created in this step (DISP=(NEW,CATLG)).

ABX2498E ABXUTIL plan not known to DB2

Explanation: The DB2 plan for the utility (ABXUTIL) is not bound on the subsystem referred to in the SOURCEDB2 statement.

User Response: Bind the plan and rerun. Delete the output data sets (DISP=(NEW,CATLG)).

ABX2499E DB2 is down

Explanation: See message ABX2497E above.

User Response: See message ABX2497E above.

ABX2500E CAF returned:

Explanation: The CAF returned a status that was not programmed for. The run terminates. See the message following this message in the SYSOUT for the CAF status.

User Response: Examine the CAF response and make appropriate corrections.

ABX2501E CAF response

Explanation: This is the status as reported by the CAF.

User Response: Take action as appropriate to the reported CAF status.

ABX2502E DB2 is shutting down

Explanation: The DB2 subsystem referenced in SOURCEDB2 statement is terminating. The ABXUTIL utility will also shut down.

User Response: When the DB2 subsystem returns to normal operation, delete the ABXUTIL output data sets (DISP=(NEW,CATLG)) and rerun the utility.

ABX2503E DB2 requesting forced termination

Explanation: The DB2 subsystem is effecting a forced termination. This request is honored by also terminating the utility.

User Response: When the DB2 subsystem returns to normal operation, delete the ABXUTIL output data sets (DISP=(NEW,CATLG)) and rerun the utility.

ABX2504E DB2 found :

Explanation: This message displays the messages and status as returned by DB2.

User Response: Examine the DB2 messages and RC and reason(s) and take appropriate action.

ABX2505E DB2 messages

Explanation: This is the information reported by the DSNTIAR formatting routine.

User Response: Take action as indicated in the DB2 messages. You will likely have to delete the ABXUTIL output data sets before rerunning this step.

ABX2506E SQL call in effect: what-the-call-was-for-the-above-errors

Explanation: This is what the application is executing. Like OPEN CURSOR, FETCH, PREPARE, etc.

User Response: This message is in conjunction with the reported DB2 messages.

ABX2507E Invalid cursor open call

Explanation: An OPEN CURSOR failed. This message will be in conjunction with DB2 status and messages.

User Response: Take the corrective action suggested by the DB2 status.

ABX2508E Prepare of select failed

Explanation: The SELECT for the current TABLE processing would not PREPARE. The TEXT is incorrect or the table is not known, or there may be another reason. The DB2 messages associated with this message will show the cause.

User Response: Examine the DB2 messages emitted in association with this message and correct the cause. The ABXUTIL output data sets may have to be deleted.

ABX2509E Describe failed

Explanation: In effecting a DESCRIBE against a prepared SELECT, the DESCRIBE failed.

User Response: Examine the DB2 messages emitted in association with this message and correct the cause. The ABXUTIL output data sets may have to be deleted.

ABX2510E Describe failed

Explanation: In effecting a DESCRIBE against a prepared SELECT, the DESCRIBE failed.

User Response: Examine the DB2 messages emitted in association with this message and correct the cause. The ABXUTIL output data sets may have to be deleted.

ABX2511E Either you are not authorized to this DB2 or to the plan (ABXUTIL) or the plan has not been bound on subsystem id

Explanation: When opening the ABXUTIL plan, DB2 returned a status indicating the above status.

User Response: Bind the plan or grant use on the plan or add the user with the appropriate authorizations. When you rerun this step, the ABXUTIL output data sets may have to be deleted.

ABX2512E DSNALI could not be loaded, DD STEPLIB does not include an appropriate reference to 'qualifier.SDSNLOAD'

Explanation: The STEPLIB does not include a concatenation to the appropriate SDSNLOAD for the pertinent DB2 subsystem.

User Response: Correct the STEPLIB in the JCL. Rerun the step after deleting the output data sets.

ABX2513E Cannot continue parsing as DB2 is not available

Explanation: When the processing mode is DB2, calls are made to DB2 to PREPARE and DESCRIBE the SELECT statements. This is necessary to assure the correctness of the SELECT, and to assure the needed COLUMNS are present in the SELECT. Without access to DB2, the parsing cannot be complete.

User Response: Wait until the required DB2 subsystem is available or bring the subsystem up.

ABX2514E Bad parameter list - no table pointer (internal logic error)

Explanation: This is indicative of memory corruption.

User Response: Forward your SYSOUT to your support personnel.

ABX2520I Loading source data for SOURCE *source-name*

Explanation: During DB2 processing, the TABLE statement to be executed now requires the named SOURCE (part of a RELATIONSHIP statement) in memory. This SOURCE will be read and tabularized.

User Response: No user action is required.

ABX2521I Loading content for CONTENT named *content-name*

Explanation: During DB2 processing, the TABLE statement to be executed now requires the named CONTENT (part of a RELATIONSHIP statement) in memory. This CONTENT will be read and tabularized.

User Response: No user action is required.

ABX2522I **Discarding information in SOURCE *source-name* as it is no longer needed**

Explanation: The last TABLE statement that refers to this SOURCE has been serviced and the SOURCE data is no longer needed. It will be deleted from memory.

User Response: No user action is required.

ABX2523I **Discarding information in CONTENT *content-name* as it is no longer needed**

Explanation: The last TABLE statement that refers to this CONTENT has been serviced and the CONTENT data is no longer needed. It will be deleted from memory.

User Response: No user action is required.

ABX2530E **The source (len = *nnn*) used in statement number *mmm* (*source-name*) does not match the data length (lll) for column *column-name***

Explanation: There is a mismatch in the SOURCE data length and the COLUMN data length (excluding CHAR and VARCHAR) that will prevent matching the data.

User Response: Correct the incorrect definition, either the SOURCE or the matching COLUMN.

ABX2531E **The source used in statement number *nnn* (*source-name*) does not match the data type for column *column-name***

Explanation: There is a mismatch in the SOURCE data type and the COLUMN data type (excluding CHAR and VARCHAR) that will prevent matching the data.

User Response: Correct the incorrect definition, either the SOURCE or the related COLUMN.

ABX2532I **Processing table reference id**

Explanation: This is only a progress message and states where ABXUTIL is in executing the control syntax.

User Response: No user action is required.

ABX2533I **Reducing prerequisite count (currently *nnn*) for reference-id**

Explanation: In parent-child processing, the parents must be processed first. After processing a referred-to parent, the dependencies of the children are reduced. When these dependencies are reduced to zero, the child table processing can be effected.

User Response: No user action is required.

ABX2534I **Finished processing table reference id**

Explanation: This is a progress message and states where the processing is now located.

User Response: No user action is required for this message.

ABX2550E **The ddname ddname is already used in statement number *nnn***

Explanation: In DB2 processing there is more that one request to place data on the same file. Each output file should have its unique data.

User Response: Choose a different (or unique) DD for each file.

ABX2551E **Invalid PRINT parameter**

Explanation: The PRINT parameter of the TABLE statement has invalid syntax. The allowed SYNTAX is PRINT or PRINT(*nnn*).

User Response: Correct the PRINT declarative.

ABX2552I **Adding column named *column-name***

Explanation: This is a debugging message and may have little value to the user.

User Response: Ignore this message.

ABX2553W **The column named *column-name* is duplicated in the select list. You may have to edit the control cards to change the name. Giving it a name of 'Duplicate*nnn*' . Any references to this name (as in a SORT or INSERT) should be examined.**

Explanation: The SELECT list has the redundant occurrence of a column name. This is probably a mistake, except when the SELECT is doing a JOIN, where it is sometimes unavoidable. If the name must repeat, rename the column using the COLUMN (new-name) ... ORDINAL(*nnn*) facility.

User Response: Correct the SELECT or rename the COLUMN or edit the produced LOAD and SORT cards to obtain the desired result.

ABX2554E **The added column (*column-name*) is redundant with a column named in the select - rename the added column**

Explanation: In a COLUMN statement: COLUMN ... NEW, the chosen name collides with the name of a column already in the select list.

User Response: Choose a unique name for this added column.

ABX2555E **Error when adding to the NAMES table**

Explanation: This is normally accompanied with an 'out of memory' message.

User Response: Increase the region size.

ABX2556E **The added column (*column-name*) is redundant with a column named in the select - rename the added column**

Explanation: In a COLUMN statement, COLUMN ... NEW, the chosen name collides with the name of a column already in the SELECT list.

User Response: Choose a unique name for this added column.

ABX2557E **Error when adding to the FIELDS/COLUMNS table**

Explanation: If this is accompanied with an 'out of memory' message, the region size must be increased. If there is not an 'out of memory' message, memory has probably been corrupted..

User Response: If out of memory, increase the region size. Otherwise forward this SYSOUT to your support personnel.

ABX2558I **The following messages apply to the THEN INSERT specified in statement number *nnn***

Explanation: In DB2 process editing, the THEN INSERT INTO DD:DDNAME statements are processed in a separate pass, not synchronous with reading the statement. Errors such as an invalid column name will not be caught at initial parsing time.

User Response: Correct the error(s) that are indicated in the messages following this message.

ABX2559E **The only form of THEN INSERT supported in DB2 processing is the THEN INSERT INTO DD:ddname format (statement number *nnn*)**

Explanation: A THEN INSERT INTO target type of statement was found, which is only supported in sequential or VSAM or DLI processing.

User Response: Correct the THEN INSERT statement.

ABX2560E **Internal error, COLUMN entry does not have a FIELD pointer or is not a LITERAL**

Explanation: This is an internal error.

User Response: Forward your SYSOUT to your support personnel.

ABX2561I **The above error(s) were found in processing the THEN INSERT for statement *nnn***

Explanation: Some errors related to THEN INSERT in DB2 mode are not caught on the initial parsing of the statement.

User Response: Correct the THEN INSERT statement numbered *nnn*.

ABX2562I **The above error(s) are from analyzing the THEN INSERT in statement *nnn***

Explanation: Some errors related to THEN INSERT in DB2 mode are not caught on the initial parsing of the statement.

User Response: Correct the THEN INSERT statement numbered *nnn*.

ABX2563E **Could not initialize an RBLOCK for table FIELDS**

Explanation: This message is usually indicative of memory corruption.

User Response: Forward your SYSOUT to your support personnel.

ABX2564E **Could not find information for column *column-name***

Explanation: The named column is not defined in the select list.

User Response: Either correct the SELECT column list, or correct this reference.

ABX2565E **Could not find a SQLVAR entry for *column-name* (statement *nnn*)**

Explanation: A reference to the named column could not be resolved as it was not in the list of selected column.

User Response: Either correct the SELECT or the reference.

ABX2566E **Logical error - an output descriptor without a field pointer**

Explanation: There is a problem with the ABXUTIL program.

User Response: Forward this SYSOUT to your support personnel.

ABX2567E **Logical error - an output descriptor without a field pointer**

Explanation: This should not occur. If it does, there is a problem with the ABXUTIL program.

User Response: Forward this SYSOUT to your support personnel.

ABX2568E **The produced record is excessive in length at *nnn* bytes**

Explanation: The produced record exceeds the limit of 32K.

User Response: Change the formatting directives to create a shorter record.

ABX2569I **Table reference-id has no filtering (LIMIT, explicit SELECT, SKIP, SOURCE limitations or filtered parents) - all rows will be selected.**

Explanation: A TABLE statement has no inhibition to the rows to select. The whole table (all the rows) will be exported.

User Response: No user action is required.

ABX2570E **Could not initialize access to the FIELDS table**

Explanation: This is an internal error and is probably associated with memory corruption.

User Response: Forward your SYSOUT to your support personnel.

ABX2571E **Error in opening the DB2 cursor**

Explanation: DB2 did not allow opening the cursor associated with the SELECT pertinent to the current TABLE being processed.

User Response: Resolve the problem as indicated in the accompanying DB2 messages.

ABX2572E **The column *column-name* is not defined**

Explanation: There is a reference to the named column that is not defined.

User Response: Correct the reference to the appropriate column.

ABX2573E **Could not find the based on column for CONTENT *content-name***

Explanation: In a COLUMN or THEN INSERT statement, there is a reference to the named CONTENT, which is not defined.

User Response: Correct either the associated RELATIONSHIP statement or this reference.

ABX2574E **Could not find the based on column for CONTENT *column-name***

Explanation: There is a reference to a CONTENT where the reference uses an undefined column as the key with which to match.

User Response: Correct this reference.

ABX2575E **The column *column-name* is not defined**

Explanation: The cited column is not defined.

User Response: Correct this reference.

ABX2576E **Could not open the table for capturing column *column-name* in table reference-id for matching the children rc =*nn***

Explanation: This is an insufficient memory problem.

User Response: Increase the region size.

ABX2577E **Could not do a table initialization for *internal_table-name* , rc =*nnn***

Explanation: This is most likely a memory corruption problem.

User Response: Forward your SYSOUT to your support personnel.

ABX2578E **Could not find column *column-name* in table reference-id**

Explanation: The named column is not part of the named table, or in its SELECT column list.

User Response: Correct the SELECT column list or change this reference.

ABX2579E **Could not initialize access to source named *source-name***

Explanation: This is most likely a memory corruption problem.

User Response: Forward your SYSOUT to your support personnel.

ABX2580E **Could not find the column (*column-name*) for source *source-name***

Explanation: In an 'ONLY WHEN *column-name* {NOT} IN (*source-name*)' the *column-name* was not found. It is probably not in the SELECT list associated with the TABLE statement.

User Response: Either correct the *column-name*, or alter the select column list. If this column is not desired in the

TARGET table, use a COLUMN (*column-name*) ... OMIT declarative to suppress including it in the output records.

ABX2581E **Error in creating the load control cards for table table-name**

Explanation: There should be other message(s) preceding this message explaining the true problem. If there are no such messages, this is a problem with ABXUTIL itself.

User Response: Correct the errors in the preceding message(s).

ABX2582E **Failed saving a parental key (content) from parent Reference-id, rc = *nnn***

Explanation: If the RC is 2, this is a duplicate key. Otherwise there is a memory shortage.

User Response: Increase the region size if there is a memory shortage. Else forward your SYSOUT to your support personnel.

ABX2583I **Error in closing the cursor**

Explanation: DB2 reported some problem when ABXUTIL attempted to close a cursor. This message will accompany messages emitted by DB2.

User Response: Correct the problem referred to by DB2.

ABX2584E **Could not open the output file with DDNAME ddname for table Reference-id, LRECL of *nnn***

Explanation: The DDNAME as stated in the TABLE declarative DD:DDNAME could not be opened.

User Response: See that the correct DD JCL statement is included in the step's JCL. If the LRECL conflicts, remove the LRECL in the JCL.

ABX2585E **Error in formatting an output record for table reference-id**

Explanation: This message will accompany other message(s) indicating the problem.

User Response: Correct the indicated error.

ABX2586E **Error effecting insert when processing table reference-id**

Explanation: This message will accompany other message(s) indicating the problem.

User Response: Correct the indicated problem.

ABX2587I **At the limit of *nnn* for table reference-id - bypassing the remaining rows**

Explanation: A LIMIT for exporting rows in this table was specified and met. Processing for this TABLE will end and the next existent TABLE statement will commence processing.

User Response: None required.

ABX2588E **For the data in column *column-name*, the length (*nnn*) is incorrect**

Explanation: For a VARCHAR column the length indicated is less then zero or greater than the catalog indicates. This is probably a memory corruption issue.

User Response: Forward your SYSOUT to your support personnel.

ABX2589E **For the data in column *column-name*, the length (*nnn*) is incorrect**

Explanation: The length indicated in a VARCHAR column is outside the range expected as indicated in the catalog.

User Response: This is probably a memory corruption error. Forward your SYSOUT to your support personnel.

ABX2590E **Error in writing to the DDNAME ddname for table reference-id**

Explanation: This is usually an insufficient space issue.

User Response: Increase the SPACE allocated for the output data set.

ABX2591I **For the Table reference-id, there were *nnn* rows processed and *mmm* rows selected**

Explanation: This is an informational message and states the results for processing this TABLE statement.

User Response: No user response is required.

ABX2592I ***nnn* rows were bypassed for heritage, and *mmm* rows for IN SOURCE processing**

Explanation: This message accompanies message ABX2592I above. This informational message states how many rows were cast aside due to the parent-child relationship constraint(s) (*nnn*) and for any ONLY WHEN IN ... constraint(s) (*mmm*).

User Response: No user response is required.

ABX2593I ***nnn*** **rows were skipped due to a SKIP request**

Explanation: This message accompanies message ABX2592I above. This indicates that a SKIP of *nnn* rows was effected.

User Response: No user response is required.

ABX2594I **LIMIT *lll*THEN ONE OF *nnn* caused *mmm* rows to be bypassed**

Explanation: This message accompanies message ABX2592I above. The LIMIT (*lll*) was met, then only one of each *nnn* rows was selected causing a total of *mmm* records to be cast out.

User Response: No user response is required.

ABX2595I **For the parent->child from table reference-id *nnn* parental columns (*column-name*) were null**

Explanation: This message accompanies message ABX2592I above. In the parent, there were *nnn* rows that has NULL indicated for the named column used in a parent child relationship.

User Response: No response is required.

ABX2596I **For the parental relationship to table parent-reference-id the child column *column-name* was matched *nnn* times, not matched *mmm* times. There were *zzz* child columns that were null**

Explanation: This message accompanies message ABX2592I above. This is the status of the parent-child key matching specified.

User Response: No user action is required.

ABX2597I **For the THEN INSERT (statement *nnn*), there were *mmm* record(s) emitted**

Explanation: This message accompanies message ABX2592I above. This is the count of the records inserted due to statement *nnn*.

User Response: No user action is required.

ABX2598E **You cannot use a DYNAMIC FIELD (END - *nn*) in DB2 mode**

Explanation: In DB2 mode, there is no definition of a record, just a collection of columns. So a reference to the 'END' is meaningless.

User Response: Correct this definition.

ABX2599E **Error in formatting the THEN INSERT for statement *nnn* processing table reference-id**

Explanation: There was an attempt to format a record for the stated THAN INSERT. This formatting failed. The reason is in the prior message(s).

User Response: Correct the problem indicated in the prior message(s).

ABX2600E **Failed on writing for table reference-id, THEN INSERT statement *nnn***

Explanation: The write for the formatted record failed. The usual reason is insufficient space in the output data set.

User Response: Increase the SPACE parameter for the output data set.

ABX2601I **Encountered a null statement - statement *nnn***

Explanation: Statement *nnn* contains no control syntax - only a statement terminator, the semi-colon (;).

User Response: Ignore this message as it is harmless.

ABX2610E **A zero length hex literal is not supported**

Explanation: A HEX literal of zero length (like x'') was found. This is invalid.

User Response: Correct the literal.

ABX2611E **A zero length character literal is not supported**

Explanation: A character literal (like '') was found. This is not supported.

User Response: Correct the literal.

ABX2615E **The length of the child column name exceeds 30 characters**

Explanation: Due to the limits of an SQLDA, the limit on a column name size is thirty characters.

User Response: Shorten the name as the length prior to DB2 version 8 is 18 characters.

ABX2616E **The length of the child column name exceeds 30 characters**

Explanation: The name as specified cannot be in a SQLDA.

User Response: Shorten the name.

ABX2617E **The length of the child column name exceeds 30 characters**

Explanation: The name specified exceeds the constraint of an SQLDA.

User Response: Shorten the name.

ABX2618E **In defining a compound key, there should be a column name following the comma**

Explanation: In the parent-child relationship expression, the use of compound keys is indicated with *reference-id*.column,column ... The second and succeeding columns are preceded by commas. Consequently there must be a column name after the comma , but this is not the case.

User Response: Correct the parent-child declarative in this TABLE statement.

ABX2619E **The length of the child column name exceeds 30 characters**

Explanation: The SQLDA constraint limits the size of the *column-name*.

User Response: Shorten the column name.

ABX2620E **This parent child ADR does not match in column count, for the parent *nnn* columns were used, for the child *nnn***

Explanation: In the use of compound keys, the column count in the parental key should be the same as the column count in the child.

User Response: Correct this parent-key definition.

ABX2621E **Could not open the table for capturing parent-child column(s) from table reference-id for matching the children rc = *nnn***

Explanation: This probably follows an 'out of memory' message.

User Response: If there is insufficient memory, increase the region size, else forward your SYSOUT to your support personnel.

ABX2622E **Failed saving a parental key from parent *reference-id*, rc = *nnn***

Explanation: A parental row was fetched and an unsuccessful attempt to save the parent's key was made. This is probably accompanied with an 'out of memory' message, or has a return code of two, which indicates a duplicate key. If there is a duplicate key, the constraint is probably incorrectly defined.

User Response: If there is a shortage of memory (accompanying message), increase the region size. If the key is a duplicate, check the relationship definition. Otherwise memory corruption has occurred and the SYSOUT should be forwarded to your support personnel.

ABX2623E **This return code (2) indicates a redundant key**

Explanation: In the case of the above message (ABX2622E) preceding this, the recommended action is printed.

User Response: Increase the region size.

ABX2624I **For the parent->child from table reference-id *nnn*= parental keys contained at least one null column**

Explanation: This is an informational message, but affects processing as the current row will be cast aside with the ACCEPT NULLS declaration.

User Response: See that the desired result was achieved.

ABX2625I **For the parental relationship to table *reference-id*, the child column(s) were matched *nnn* times, not matched *mmm* times. There were *zzz* child columns that were null**

Explanation: After processing a TABLE, these statistics are published to reflect the status of the parent-child definitions.

User Response: No user response is required.

ABX2630I **OPTION 'DUMMY' was specified, writing will only occur for THEN INSERTS - there will be no writing to table dd:ddname(s);**

Explanation: The parameter was specified for this given table. No load image is created, but any data for targeted THEN INSERT data sets are produced.

User Response: No response is required,

ABX2631E **Could not open ddname ABXCNTLO**

Explanation: The ABXUTIL program creates process control statements for the load executor. Load control statements, and optional Sort control statements as well as statistics for sizing sort work data sets are emitted. An execution of the load executor will fail without this control information.

User Response: If you expect to load the exported data, you must have a DD named ABXCNTLO in which to place the control information. Correct this issue. You can use a DD DUMMY if desired.

ABX2632E **Could not open the control file ABXCNTLI**

Explanation: The ISPF process places control information for SORT and LOAD into a data set that ABXUTIL merges with its additional control information. This data set is required if you intend to execute the Load Executor. If you do not wish to execute the LOAD process, this dd can be specified as DD DUMMY.

User Response: Be sure that a dd named ABXCNTLI is included in the run time JCL.

ABX2633E **Error in writing to the output control file**

Explanation: In attempting to write the SORT control cards, the ABXCNTLO data set is probably out of space.

User Response: Increase the space for the ABXCNTLO referenced data set.

ABX2634E **Error in writing to the output control file ABXCNTLO**

Explanation: The ABXCNTLO data set is probably out of space.

User Response: Increase the space for the ABXCNTLO referenced data set.

ABX2635E **Error in writing to the output control file ABXCNTLO**

Explanation: The ABXCNTLO data set is probably out of space. The system was attempting to write LOAD control cards.

User Response: Increase the space for the ABXCNTLO referenced data set.

ABX2636E **Error in writing to the output control file**

Explanation: The system was attempting to write a CREATE TABLE statement to ABXCNTLO when the error occurred. The ABXCNTLO data set is probably out of space.

User Response: Increase the space for the ABXCNTLO referenced data set.

ABX2637E **Error in writing to the output control file**

Explanation: The system was attempting to write COLUMN DEFINITION DDL statements to ABXCNTLO when the error occurred. The ABXCNTLO data set is probably out of space.

User Response: Increase the space for the ABXCNTLO referenced data set.

ABX2638E **Error in writing to the output control file**

Explanation: The system was attempting to write miscellaneous control information to ABXCNTLO when the error occurred. The ABXCNTLO data set is probably out of space.

User Response: Increase the space for the ABXCNTLO referenced data set.

ABX2639I **Produce nothing as SELECT NOTHING was specified**

Explanation: In VSAM, sequential, or DLI processing modes, a SELECT statement stated SELECT NOTHING ... Consequently, no records will be produced from this SELECT.

User Response: No user action is required.

ABX2640E **A literal exceeded 256 characters**

Explanation: The ABXUTIL program only services literals up to a size of 256 bytes. If a longer literal is needed, use two (or more) literals and modify the corresponding SELECT or INSERT or PREDICATE expressions accordingly.

User Response: Use a shorter literal and modify the control syntax as described above.

ABX2641E **Found a COLON (:) outside of a string literal**

Explanation: The colon is the comment mark when in column one (or one and two for non-printing comments). Elsewhere it can be used in the DD:DDNAME expression. Beyond that, it is only valid in a literal. It does not mark a host variable in DB2 mode of processing.

User Response: Correct the syntax of the statement.

ABX2642E **A literal spans a card image that is not marked with a continuation (-) in column 72**

Explanation: A literal was started with a quote mark (") or an apostrophe (') or a shift-out (X'E') and was not terminated on the same card. At the same time the word-continuation mark (a dash (-) in column 72) was not present indicating a 'word' continues on the next card. For processing purposes a literal (up to 256 characters) is considered a single word, even if there is a delineator such as a blank within the literal.

User Response: Correct the literal.

ABX2643E **An increment that is a real number (has an exponent or decimal point)**

cannot be applied to an integer data type

Explanation: The only supported increment for an integer type of variable (an integer or a short) is itself an integer (under 10,000 in content).

User Response: Correct the VARIABLE statement.

ABX2644E **Cannot initialize a non-real (integer) number with a real number (exponent or decimal point)**

Explanation: The initial value for an integer (or short) must be an integer - not a float or double value.

User Response: Correct the VARIABLE statement.

ABX2645E **Warning the VARIABLE *variable-name*exceeded its maximum of 2.0e+38 if this is a key you will have non-uniquess**

Explanation: This is the limit for floats and doubles supported by IBM File Export for z/OS. When this limit is reached, it wraps back to 0.0. If the variable wraps, there is a possibility of creating duplicate values that could cause a collision if this is a key or part of the key.

User Response: If this is a problem, either choose a smaller initial value or decrease the increment.

ABX2646E **Warning the VARIABLE *variable-name* exceeded its maximum of 2.0e+38 if this is a key you will have non-uniques**

Explanation: This is the same case as cited for message ABX2645E. One applies to FLOATS, the other to DOUBLES

User Response: See message ABX2646E.

ABX2647E **Could not access SYSIBM.SYSTABLES looking up table owner.table**

Explanation: In researching to create DDL for the target DB2, the SQL to research a table's attributes would not execute.

User Response: Correct this error. Perhaps a grant is required.

ABX2648E **No record found in SYSIBM.SYSTABLES for owner.table**

Explanation: The catalog had no information for the source table or view shown above.

User Response: Correct this issue, possibly with a grant or with a change in control syntax.

ABX2649E **Could not access SYSIBM.SYSCOLUMNS looking up table owner.table**

Explanation: In researching source columns for creating target DDL, the attempt to access the catalog failed.

User Response: Correct this problem, probably with a grant.

ABX2650I **Could not load DSNHDECP - using the apostrophe as the string delimiter**

Explanation: In preparing dynamic SQL to access the catalog for the purpose of research for creating DDL, an attempt to load the module DSNHDECP failed. This module is needed to ascertain the string delimiter to create appropriate site-dependent SQL text. This attempt to load DSNHDECP failed, so the default of apostrophe will be used.

User Response: DSNHDECP should be available for loading from the STEPLIB. If the default works, this message is informational only.

ABX2651E **The word LOAD must be followed by a left parenthesis**

Explanation: In the TABLE statement, the word load must be followed with a left (opening) parenthesis in the format of: LOAD (parameter, parameter).

User Response: Correct the TABLE statement.

ABX2652E **Ran out of text before completing the LOAD parameters**

Explanation: Before the load directive completed, the statement completed. The ending semi-colon may be misplaced, or the syntax was not completed.

User Response: Correct the TABLE statement.

ABX2653E **The word *found-word* is not a LOAD option like REUSE, REPLACE, RESUME**

Explanation: This is an unsupported 'word' as a load parameter.

User Response: Correct the TABLE statement.

ABX2654I **Could not open ddname ABXDDL**

Explanation: CONTROL output (separate from the ABXUTILO output) for DDL was requested. As the required DDNAME would not open, this output is suppressed.

User Response: Either supply the DD card, or negate the request for this output or ignore this message.

ABX2655W **Truncating DDL control cards from *nnn* to *nnn*, this message will appear but once. Setting return code to a minimum of 8**

Explanation: The produced DDL is excessive in length. This should not occur and indicates a problem with ABXUTIL.

User Response: Forward your SYSOUT to your support personnel.

ABX2656W **Error in writing to the DDL output file, terminating the collection of DDL data and setting the return code to 8 (minimum)**

Explanation: The probable cause is insufficient space in the ABXCNTLO file.

User Response: Increase the space of the data set referenced by the ABXCNTLO data set.

ABX2657I **Could not open ddname ABXLOADC**

Explanation: Additional output was requested for the LOAD card images (to the file pointed to by ABXLOADC). This file could not be opened.

User Response: Ignore this message, or include the required DD statement.

ABX2658E **Truncating LOAD control cards from *nnn* to *mmm*. This message will appear but once. Setting return code to a minimum of 8**

Explanation: There is a problem in formatting the load control card images.

User Response: Forward this SYSOUT to your support personnel.

ABX2659E **Error in writing to the LOAD cards output file, terminating the collection of LOAD data and setting the return code to 8 (minimum)**

Explanation: The probable cause is insufficient space in the data set referenced by the ABXCNTLO DD card.

User Response: Increase the space for the data set referenced by ABXCNTLO.

ABX2660I **Could not open ddname ABXSORTC**

Explanation: The processing requested additional output in the form of editable SORT control statement. However, the file could not be opened.

User Response: Include aa DD named ABCSORTC in the run time JCL.

ABX2661E **Truncating SORT control cards from *nnn* to 80. This message will appear but once. Setting return code to a minimum of 8**

Explanation: There is a problem in the produced SORT control cards.

User Response: Forward your SYSOUT to your support personnel.

ABX2662E **Error in writing to the LOAD cards output file, terminating the collection of LOAD data and setting the return code to 8 (minimum)**

Explanation: There is a request to emit additional images of the LOAD control cards. Very likely, there is insufficient space in the allocated data set.

User Response: Increase the space allocated to ABXLOADC.

ABX2663I **Option NOCONTROL was specified - no load executor control file will be created**

Explanation: In the control syntax OPTIONS (... NOCONTROL...) was specified. It will not be possible to execute the load executor as the required control data will not be available. This option may be useful if you are only interested in the data produced by the THEN INSERT statement(s).

User Response: Unless you were intending to load the produced row images, ignore this message. Otherwise do not specify the NOCONTROL option.

ABX2664I **Global SKIP applies to input records, while SKIP on a TYPE applies to output records**

Explanation: A SKIP statement was found, and the applicability of this statement in non-DB2 mode is portrayed.

User Response: No response is required.

ABX2665E **REPLACE is exclusive to the previously stated RESUME**

Explanation: In a TABLE statement, both the parameter REPLACE and the parameter RESUME were specified. This is an incompatible combination.

User Response: Correct the TABLE statement.

ABX2666E **RESUME is exclusive to the previously stated REPLACE**

Explanation: In the load directive of a table statement, conflicting parameters were specified.

User Response: Correct the TABLE statement.

ABX2667I **JCL is not valid on an insert (only THEN INSERT) - ignoring it**

Explanation: The request to use the external specification on a file (the LRECL) is being ignored as INSERT is placed into ABXOUT, not a user data set as is the case with THAN INSERT INTO DD:DDNAME. The output of INSERT goes to the primary output stream.

User Response: Ignore this message.

ABX2668I **VSAM is not valid on an insert (only THEN INSERT) - ignoring it**

Explanation: The specification of JCL or VSAM for an insert is ignored as this data goes to the primary output stream - ABXOUT.

User Response: Ignore this message

ABX2669I **A maximum on THEN INSERT is being ignored**

Explanation: A maximum was specified, but this is not applicable to a THEN INSERT INTO DD:DDNAME as variable length records are not supported for this type of insert.

User Response: Ignore this message unless you truly want variable length records created - which is not supported in this release of ABXUTIL for THEN INSERT processing.

ABX2670E **The column name *column-name* is already defined as a literal or variable**

Explanation: This name has already been used, and this use constitutes a collision.

User Response: Probably the colliding named literal or variable should be renamed as the names used for columns have special meaning.

ABX2671I **Output field named *field_name* defined in statement *statement_number* is never referenced**

Explanation: The output field was defined but not used.

User Response: No response is required unless you intended to use this field.

ABX2672E **Invalid use of OCCURS -- Expected a number to follow**

Explanation: The correct format is OCCURS (*nn*).

User Response: Correct the format of this statement.

ABX2673E **Expected an integer to follow OCCURS**

Explanation: The correct format is OCCURS (*nn*).

User Response: Correct the format of this statement.

ABX2674E **Can only use PRECISION on a field that is a short integer or an integer or a packed**

Explanation: The field is not a numeric field.

User Response: Either redefine the field as numeric or remove the PRECISION declarative.

ABX2675E **Invalid use of PRECISION-- Expected a number to follow**

Explanation: The correct format is PRECISION (*nn*).

User Response: Correct the format of this statement.

ABX2676E **Expected an integer to follow PRECISION**

Explanation: The correct format is PRECISION (*nn*).

User Response: Correct the format of this statement.

ABX2677E **Can only use SCALE on a field that is a short integer or an integer or a packed**

Explanation: SCALE is an attribute of numbers.

User Response: Either this field should be redefined as numeric or SCALE should be eliminated.

ABX2678E **Invalid use of SCALE-- Expected a number to follow**

Explanation: The correct format is SCALE (*nn*).

User Response: Correct the format of this statement.

ABX2679E **Expected an integer to follow SCALE**

Explanation: The correct format is SCALE (*nn*).

User Response: Correct the format of this statement.

ABX2680E **Can only use SIGNED on a field that is a short integer or an integer or a packed**

Explanation: SIGNED is a numeric attribute.

User Response: Correct the format of this statement.

ABX2681E **Expected the word ON to follow DEPENDING**

Explanation: The correct format of this statement is OCCURS DEPENDING ON.

User Response: Correct the format of this statement.

ABX2682E **Expected the depending-on object to follow ON.**

Explanation: The correct format of this statement is DEPENDING ON *object of OCCURS DEPENDING ON (ODO)*

User Response: Correct the format of this statement.

ABX2683E **Cannot use an occurring field as the object of an OCCURS DEPENDING ON**

Explanation: The definition is not valid, since the object of OCCURS DEPENDING ON itself is an OCCURRING field.

User Response: Correct either the definition of this field or the definition of the object of OCCURS DEPENDING ON.

ABX2684E **Cannot use a dynamically addressed field as the object of an OCCURS DEPENDING ON**

Explanation: The object of OCCURS DEPENDING ON must be in a static position preceding any OCCURS DEPENDING ON fields.

User Response: Correct the definition(s).

ABX2685E **The object of an OCCURS DEPENDING ON must be a short integer or an integer**

Explanation: Either the object of an OCCURS DEPENDING ON or the reference to a field as an object of an OCCURS DEPENDING ON is in error.

User Response: Correct whichever component is in error.

ABX2686E **Expected the word AS to follow the word CALC**

Explanation: The correct syntax is CALC AS followed by an expression.

User Response: Correct the syntax of this statement.

ABX2687E **Expected an integer after the words CALC AS**

Explanation: CALC AS must be followed by an expression whose first value is an integer.

User Response: Correct this expression.

ABX2688E **The named field *(fieldname)* is not the object of a DEPENDING ON**

Explanation: A CALC AS expression is composed solely from constants and OCCURS DEPENDING ON fields.

User Response: Correct this expression.

ABX2689E **Expected a multiplier (*) after the DEPENDING ON object**

Explanation: The correct syntax is CALC AS followed by an integer.

User Response: Correct this expression.

ABX2690E **Expected an integer after the words CALC AS**

Explanation: The correct syntax is CALC AS followed by an integer.

User Response: Correct this expression.

ABX2691E **Expected a plus sign after the fixed value length in a calculated address.**

Explanation: A calculation expression has the format CALC AS integer + a sum of constants times the DEPENDING-ON objects (ODOs).

User Response: Correct this expression.

ABX2692E **Extraneous word *found_word* after length fields**

Explanation: An inappropriate word was found in a field definition.

User Response: Correct the FIELD definition.

ABX2693E **Position specified as calculated but no algorithm supplied**

Explanation: The format of FIELD(field_name) = (CALC, length,data_type) was used, but the algorithm (CALC AS...) was not found.

User Response: Correct this expression.

ABX2694E **Illegal use of the back- slash (\)**

Explanation: Subscripting is not supported in this release of IBM File Export for z/OS.

User Response: Correct this statement.

ABX2695E Subscript out of range

Explanation: The value of a subscript is beyond 32k and is thus illegal.

User Response: Correct this statement.

ABX2696E Invalid use of TGLEN - expected a number to follow

Explanation: TGLEN is either a constant or an expression. In either case, it begins with an integer.

User Response: Correct this statement.

ABX2697E Repeating groups require a group length.

Explanation: This field definition requires an explicit group length.

User Response: Correct this statement.

ABX2698E Only specify a group length on a repeating group/field.

Explanation: An inappropriate use of a group length was found.

User Response: Correct the syntax of this statement.

ABX2700E The TARGET database name keyword (TDBNAME) must be followed by a left (opening) parenthesis.

Explanation: The option syntax within a TABLE definition for assigning the target DATABASE name for the CREATE TABLE DDL statement is TDBNAME*(target_database_name)*. The required left parenthesis is missing.

User Response: Correct the TABLE statement.

ABX2701E The specified TARGET database name is too long.

Explanation: In specifying the target database name for the create table, the cited name exceeds the length allowed.

User Response: Correct the statement.

ABX2702E Expected the target database name following TDBNAME and (

Explanation: The correct syntax is TDBNAME*(target_database_name)*. This statement ended with the left parenthesis.

User Response: Correct the statement.

ABX2703E Expected a right parenthesis to follow the target database name

Explanation: The correct syntax is TDBNAME*(target_database_name)*. The right parenthesis was not found.

User Response: Correct the statement.

ABX2704E The TARGET tablespace name keyword (TTSNAME) must be followed by a left (opening) parenthesis.

Explanation: The correct syntax is TTSNAME*(table_space_name)*. The required left parenthesis is missing.

User Response: Correct the statement to supply the missing left parenthesis.

ABX2705E The specified target tablespace name is too long.

Explanation: The supplied target tablespace name exceeds the length allowed.

User Response: Correct the statement.

ABX2706E Expected the target tablespace name following TSSNAME (

Explanation: The correct syntax is TSSNAME*(tablespace_name)*. This statement ended with the left parenthesis.

User Response: Correct the statement.

ABX2707E Expected a right parenthesis to follow the target tablespace name

Explanation: The correct syntax is TSSNAME(tablespace_name). The required right parenthesis was not found.

User Response: Correct the statement.

ABX2708E The use of NEXT is not supported after any depending-on field

Explanation: In defining the position of a field, no form of NEXT can specify the position of any field to the right of an OCCURS DEPENDING ON field.

User Response: Use an expression to define the position. See information about the CALC statement.

ABX2709E **Developed dynamic address is before the record**

Explanation: The supplied position algorithm evaluated to a negative value, possibly due to a negative object of an OCCURS DEPENDING ON.

User Response: Correct the algorithm.

ABX2710E **Developed dynamic position is beyond the record length**

Explanation: In evaluating the addressing expression OCCURS DEPENDING ON, the developed address was beyond (to the right of) the record. Either the expression is incorrect or the object of OCCURS DEPENDING ON is excessive.

User Response: Correct either the addressing expression or examine the ODO and correct the ODO content.

ABX2711E **Can't use END in addressing in an output field**

Explanation: The position expression END--*nn* is only valid for defining input fields. It cannot be used in output fields (OFIELDs).

User Response: Correct the OFIELD statement.

ABX2712E **Cannot use a length of asterisk (*) in an output field**

Explanation: The position expression * indicating the rest of the input record is not applicable to output fields (OFIELDs).

User Response: Correct the OFIELD statement.

ABX2713E **Developed a dynamic address before the record for field *field_name***

Explanation: When developing the position of a field to the right of an OCCURS DEPENDING ON group, the supplied addressing expression developed a negative value. Either the addressing expression is incorrect or an ODO is negative.

User Response: Correct the item in error.

ABX2714E **Call to calculate an output field position for a non-dynamic field**

Explanation: This is an internal error in ABXUTIL.

User Response: Please supply your control syntax to your support personnel.

ABX2715E **Exceeded the max of eight subscripts**

Explanation: This version of ABXUTIL can support only eight levels of subscripting.

User Response: This level of subscripting is beyond the intent of ABXUTIL.

ABX2716E **A plan name is limited to eight characters**

Explanation: OPTIONS (...PLAN(plan_name)...) was submitted in which the supplied name of the plan exceeded the 8-character length limit.

User Response: Use a valid PLAN name.

ABX2717E **Expected a right parenthesis following the plan name**

Explanation: The syntax in the OPTIONS declarative is PLAN(plan_name). The required right parenthesis was not found.

User Response: Correct the OPTIONS statement.

ABX2718E **The OPTIONS keyword PLAN must be followed by a left parenthesis**

Explanation: The syntax in the OPTIONS declarative is PLAN(plan_name). The left parenthesis was not found.

User Response: Correct the OPTIONS statement.

ABX2719I **NOECHO can only be specified in LOAD mode (and after specifying 'LOAD') -- ignoring this**

Explanation: The OPTIONS parameter NOECHO is only applicable to LOAD processing mode.

User Response: Either specify LOAD, if applicable, or remove the NOECHO parameter.

ABX2720I **Suppressing ECHO in LOAD mode**

Explanation: An error was found in LOAD mode. Consequently, NOECHO was turned off to allow printing.

User Response: No action is required.

ABX2721I **NOINFO can only be specified in LOAD mode (and after specifying 'LOAD') -- ignoring this**

Explanation: Informational messages can only be suppressed in LOAD mode to reduce printing.

User Response: No action is required.

ABX2722I **Suppressing informational messages**

Explanation: In LOAD mode, informational messages are being suppressed at the user's request.

User Response: No action is required.

ABX2723E **A Name must have at least one letter**

Explanation: A 'word' such as 001-002 was found and is invalid.

User Response: Correct the name.

ABX2724E **At each level of number * depending-on there can only be one number**

Explanation: Addressing expressions are composed of constants multiplied by ODOs. The syntax constant * ODO * is not valid as only one constant can appear in the multiplying string.

User Response: Correct the syntax.

ABX2725E **Exceeded eight levels of indexing**

Explanation: The product's limit on subscripting is 8 levels.

User Response: Correct the syntax.

ABX2730I **For the DD named *data_definition _name* there were *nnn* data records of variable length written**

Explanation: This is an informational message about the produced output.

User Response: No action is required.

ABX2731E **Improper PLAN format**

Explanation: The format of the subparameter is PLAN(plan_name)

User Response: Correct the PLAN parameter.

ABX2732E **ONEFILE is only honored in DB2 processing**

Explanation: This option applies only to DB2 processing mode.

User Response: Correct the processing mode or remove the parameter.

ABX2733E **Expected a left parenthesis after the word MAXLEN**

Explanation: The correct format of the FIELD/OFIELD MAXLEN parameter is MAXLEN*(nnn)*

User Response: Correct the syntax.

ABX2734E **Expected a number after MAXLEN**

Explanation: The correct format of the FIELD/OFIELD MAXLEN parameter is MAXLEN*(nnn)*

User Response: Correct the syntax.

ABX2735E **Expected an integer to follow MAXLEN**

Explanation: The correct format of the FIELD/OFIELD MAXLEN parameter is MAXLEN*(nnn)*. The specified *(nnn) is not an integer.*

User Response: Correct the syntax.

ABX2736E **Expected a right parenthesis after the length in MAXLEN**

Explanation: The correct format of the FIELD/OFIELD MAXLEN parameter is MAXLEN*(nnn)*. The right parenthesis is missing.

User Response: Correct the syntax.

ABX2737E **OCCURS is not supported in DB2 processing mode -- the input data is without repeating groups**

Explanation: DB2 mode of operation works on normalized data. Therefore, OCCURS DEPENDING ON is not applicable.

User Response: Remove the OFIELD declaration.

ABX2738E **Expected either a '+_' or a '*'**

Explanation: A length calculation required either a plus sign or a multiplication sign. Neither was found.

User Response: Correct the expression.

ABX2739E **A length expression always has at least one field reference -- none was found**

Explanation: Length calculations require references to ODOs. None was found.

User Response: Correct the expression.

ABX2740E **An addressing expression always has a constant -- none was found**

Explanation: Addressing expressions contain constants. None was found.

User Response: Correct the expression.

ABX2741E **An addressing expression always has at least one field reference -- none was found**

Explanation: An addressing expression contains at least one reference to an ODO. None was found.

User Response: Correct the expression.

ABX2742E **If a field has a dynamic length algorithm, it must have a MAX_LEN specified**

Explanation: The required MAXLEN was omitted.

User Response: Correct the syntax.

ABX2743E **Call to calculate a field length for a non-dynamic_length field *(field_name)***

Explanation: This is an internal error.

User Response: Please forward your control syntax to your support person.

ABX2745E **Ran out of text in an INTO list**

Explanation: An INTO list was not terminated before running out of text.

User Response: Correct the syntax.

ABX2746E **Cannot use a literal or a number in an output formatting list**

Explanation: When an output list is supplied, it can only reference OFIELDS.

User Response: Correct the syntax.

ABX2747E **Output field named *ofield_name* is not defined at this scope**

Explanation: An output list referenced an unknown name (not at this scope and not global in scope)

User Response: Correct the reference or define the output field (OFIELD).

ABX2748E **Found no output field names in the list**

Explanation: Syntactically, an output list was expected. The list has no output field (OFIELD) names.

User Response: Correct the syntax.

ABX2749E **Field *field_name* is dependent on (OCCURS DEPENDING ON) field *field_name* which does not precede it in the output field declaratives**

Explanation: The object of an OCCURS DEPENDING ON must be positioned to the left of (come before) the reference

User Response: Correct the syntax.

ABX2750E **The list of output fields has *nnn* entries while the list of input fields and literals and collects has a count of *mmm* - these must be equal**

Explanation: In pairing of the output fields and sources of data (fields/columns/collects/literals/etc.), the count of sources and targets is not matched.

User Response: Correct the error. Be sure there is a source for each target and a target for each source.

ABX2751E **Output field list did not end with the word 'FROM'--** *found_word*

Explanation: In a SELECT statement, the output field list is terminated with the word FROM but with the word inserted into the message.

User Response: Correct the syntax.

ABX2752E **Failed to resolve output address of *ofield_name***

Explanation: The supplied algorithm for calculating a position is in error. Possibly the input record is too short to contain all the ODOs.

User Response: Correct the error.

ABX2753E **The list of output fields has *nnn* entries while the list of input fields and literals and collects has a count of *mmm* - these must be equal**

Explanation: The number of source data items must equal the number of target data items.

User Response: Correct the error. Be sure there is a source for each target and a target for each source.

ABX2754E **The QSAM variable length indicator (LLZZ) can only be specified for THEN INSERT INTO DD:DDNAME**

Explanation: The length indicator (LLZZ) is only applicable to inserting into the secondary outputs.

User Response: Either make this a secondary output (DD:DDNAME) or remove the LLZZ reference.

ABX2755E **A prior statement (number *statement_number*) did not specify LLZZ for DD *ddname*. All THEN INSERTS using this DD must be compatible**

Explanation: The system found inconsistent usage of the variable length indicator on data directed to the same target ddname.

User Response: Correct the error.

ABX2756W **WARNING -- The first four bytes of the produced record will be taken for**

the QSAM LLZZ field for variable length records

Explanation: LLZZ was specified, so the system will overlay the first four bytes in the produced record. Be sure that space is allowed for this.

User Response: Remove the LLZZ reference or allow space for the four bytes.

ABX2757E **A prior statement *(statement_number)* using this same DD *(ddnname)* specified the same LLZZ. The two definitions are not compatible.**

Explanation: If LLZZ is specified for a given DDNAME, all references to this DDNAME in THEN INSERT INTO DD:DDNAME must allow for the four byte LLZZ field.

User Response: Correct any incompatible references to this same DDNAME.

ABX2758E **Failed to resolve output address of *output_field_name***

Explanation: The supplied algorithm for calculating the position of the cited output_field_name did not develop an address. Possibly, the input record is too short to have all of the ODOs present.

User Response: Correct the situation.

ABX2759E **Cannot use an asterisk (SELECT *) in combination with a formatting output definition**

Explanation: Using an asterisk (*) as the shorthand abbreviation for all the columns does not apply to output definitions.

User Response: Create this record with the use of the asterisk or do not use an output definition.

ABX2760W **WARNING- - In populating *output_field_name*, data will be truncated.**

Explanation: The length of the source data exceeds (or can exceed) the length of the target data area.

User Response: Correct the situation or ignore this message if you expected it.

ABX2761E **The data-type for *output_field_name* does not match the data type for the paired source data**

Explanation: There is a mismatch in data type in the source paired data and the target data.

User Response: Correct this statement.

ABX2762I **This produced record has one or more OCCURS DEPENDING and the length will vary -- having approximately a minimum of *nnn* to a maximum of *mmm* actual data bytes PLUS FILLER defined between output fields.**

Explanation: For records with OCCURS DEPENDING ON, this is the projection of the produced record's size(s)

User Response: No user action is required.

ABX2763I **The produced records will be *nnn* bytes.**

Explanation: This is the projected size of the produced record created with output field definitions.

User Response: No user action is required.

ABX2764I **ABXUTIL Message Summary: *iii* Informational messages, *www* Warning messages, *eee* Error messages, and *uuu* unclassified messages.**

Explanation: This is the count of messages directly produced by ABXUTIL. It does not include messages from the I/O subcomponents of the system such as user I/O exits.

User Response: No user action is required.

ABX2784S **Unable to load the IMS Global Block**

Explanation: The product is unable to load the IMS Global Block.

System Action: ABXUTIL processing terminates.

User Response: Contact your systems programmer to create an IMS Global Block or investigate the error.

ABX2785E **No IMSID statement provided**

Explanation: You have not provided a valid IMS subsystem ID.

System Action: ABXUTIL processing terminates.

User Response: Ensure that a valid IMS subsystem ID is provided with an IMSID statement prior to the first SOURCEDBD statement.

ABX2786E **Can only specify an IMS Option when processing IMS Direct IO (DLI, DBB, BMP)**

Explanation: You have specified an option other than DLI, DBB, or BMP on this IMS Option statement.

System Action: ABXUTIL processing terminates.

User Response: Specify an IMS option statement only when using direct IMS I/O.

ABX2787E Bad option syntax--does not start with '('.

Explanation: The syntax you have specified is missing a right parenthesis.

System Action: ABXUTIL processing terminates.

User Response: Correct the statement in error.

ABX2788E Can only specify option once

Explanation: You have specified an option more than once.

System Action: ABXUTIL processing terminates.

User Response: Correct the statement in error.

ABX2789E The option parameter is too long.

Explanation: You have specified an option parameter that exceeds the acceptable length.

System Action: ABXUTIL processing terminates.

User Response: Correct the statement in error.

ABX2790E Bad option syntax -- does not end with ')'

Explanation: You have omitted a required left parenthesis.

System Action: ABXUTIL processing terminates.

User Response: Correct the statement in error.

ABX2791E Option specified too late -- already initialized

Explanation: You have specified an initialization option after initialization has occurred.

System Action: ABXUTIL processing terminates.

User Response: Correct the statement in error.

ABX2792E Invalid AUDIT parameter

Explanation: You have specified an invalid value for AUDIT.

System Action: ABXUTIL processing terminates.

User Response: Correct the statement in error.

ABX2793E Invalid EXITACT parameter

Explanation: You have specified an invalid value for EXITACT.

System Action: ABXUTIL processing terminates.

User Response: Correct the statement in error.

ABX2794E Invalid EXITNAME parameter

Explanation: You have specified an invalid value for EXITNAME.

System Action: ABXUTIL processing terminates.

User Response: Correct the statement in error.

ABX2795E Invalid DYNPSB parameter

Explanation: You have specified an invalid value for DYNPSB.

System Action: ABXUTIL processing terminates.

User Response: Correct the statement in error.

ABX2796E A target DBD must not be specified when the output database is FEUNLOAD.

Explanation: You have specified a target DBD when writing output using FEUNLOAD.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2797E Error in ABXFMDBL --Unable to get DBD information.

Explanation: The DBD you used is not available.

System Action: ABXUTIL processing terminates.

User Response: Ensure you are using the correct DBD name. If correcting the DBD does not solve the problem, notify your support personnel.

ABX2798E A target DBD must not be the same as the source DBD name when processing DLI->DLI, BMP->BMP, or DBB->DBB options

Explanation: The source and output processing modes must be the same when using DLI, DBB, or BMP processing modes.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2799E **Must have a SOURCEDBD preceding first RESULTSET statement when doing IMS processing.**

Explanation: A RESULTSET statement was encountered without a preceding SOURCEDBD.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2800E **A left parenthesis is required before the RESULTSET name**

Explanation: A left parenthesis is missing in the RESULTSET name specification.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2801E **The RESULTSET name must not be longer than 32 characters.**

Explanation: A RESULTSET name exceeded the allowable length.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2802E **The RESULTSET name must be entered.**

Explanation: A RESULTSET name is missing from the input.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2803E **The RESULTSET name must be followed by a right parenthesis**

Explanation: The right parenthesis is missing following a RESULTSET name.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2804E **Spurious text exists at the end of the RESULTSET statement when output type is not FEUNLOAD**

Explanation: Extraneous text was found at the end of a RESULTSET statement.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2805E **Duplicate RESULTSET name**

Explanation: The RESULTSET name specified is a duplicate.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2806E **Expected the word CHILD following the RESULTSET name**

Explanation: The RESULTSET name was not followed by the word CHILD.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2807E **Expected the word OF following the word CHILD**

Explanation: The word OF was missing following the word CHILD.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2808E **Parent RESULTSET name not found**

Explanation: The parent RESULTSET name was not found.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2809E **Expected the word WHERE following the parent RESULTSET name**

Explanation: The word WHERE was missing following the parent RESULTSET name.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2810E **Expected text for the child resultset/fields after the word WHERE**

Explanation: Expected text not found after the word WHERE.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2811E **Expected the word EQ or = following the child resultset/fields specification**

Explanation: Expected EQ or = missing following the child resultset/fields specification.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2812E **Expected text for the parent resultset/fields after the word EQ or =**

Explanation: Expected text not found after the word EQ or =.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2813E **Error in ABXSTART - whichdb - (*database1, database2*) is not a valid database combination**

Explanation: An invalid database combination has been specified.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2814E **Error in ABXSTART - get_segment_info failed. See JOB log for more information**

Explanation: An invalid database combination has been specified.

System Action: ABXUTIL processing terminates.

User Response: Review the job log to determine the problem. Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2815E **Child resultset name incorrectly formatted - should be resultset.field**

Explanation: The format of the resultset name is incorrect.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2816E **Expected a period after this resultset name**

Explanation: A period is missing after the resultset name.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2817E **Child FIELD name too long**

Explanation: The name has exceeded the allowable length.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2818E **The RESULTSET name in resultset.field is not that of this resultset**

Explanation: An incorrect name was encountered.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2819E **The child FIELD name must not be longer than 32 characters.**

Explanation: The child FIELD name exceeded the allowable length.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2820E **Parent resultset name incorrectly formatted - should be resultset.field**

Explanation: The format of the resultset name is incorrect.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2821E **Expected a period after this resultset name**

Explanation: A period is missing after the resultset name.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2822E **Parent FIELD name too long**

Explanation: The name has exceeded the allowable length.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2823E **The resultset name in resultset.field is not that of this resultset**

Explanation: An incorrect name was encountered.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2824E **The parent FIELD name must not be longer than 32 characters.**

Explanation: The child FIELD name exceeded the allowable length.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2825E **The number of fields specified for the parent, *value*, are not equal to the number specified for the child , *value***

Explanation: There is a mismatch in the number of parent fields and the number of child fields.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2826E **Duplicate definition of parent and field relationship.**

Explanation: A duplicate definition was encountered in the control syntax.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2827E **There is already a defined select for this resultset--you may choose to define this resultset more than once, but you will need two separate resultset definitions.**

Explanation: If you define a resultset more than once, you will need separate resultset definitions for each one.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2828E **The FIELD (*field-name*), used in the CHILD declarative for the resultset statement number *value* is not in the select.**

Explanation: The field name you specified is not part of the SELECT statement.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2829E **For statement (*name*), the data type of the parent's child-parent relationship field (*field-name*) is different from the child's field data type.**

Explanation: There is a mismatch between the data type of the parent relationship field and the child relationship field.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2830E **For statement (*name*), the data length of the parent's child-parent relationship field (*field-name*) is different from the child's field data length.**

Explanation: There is a mismatch between the data length of the parent relationship field and the child relationship field.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2831E **The FIELD (*name*) used by a CHILD declarative for the resultset statement**

number *value* is not in the select list for this parent.

Explanation: The specified field used by the CHILD declarative is not in the parent's select list.

System Action: ABXUTIL processing terminates.

User Response: Correct the ABXUTIL control syntax in error and resubmit the job.

ABX2832E **Could not open the table for capturing parent-child field(s) in the select from table *table_name* for matching the children. RC=*value***

Explanation: File Export was unable to open the specified table.

System Action: ABXUTIL processing terminates.

User Response: You will need to research the problem. The region size may need to be increased. Correct the error and resubmit the job.

ABX2833E **Could not do a table initialization for *table_name*. RC=*value***

Explanation: File Export was unable to initialize the specified table.

System Action: ABXUTIL processing terminates.

User Response: You will need to research the problem. The region size may need to be increased. Correct the error and resubmit the job.

ABX2834E **Failed saving a parental key from parent *name*. RC=*value***

Explanation: File Export was unable to save the key for the specified parent.

System Action: ABXUTIL processing terminates.

User Response: You will need to research the problem. The region size may need to be increased. Correct the error and resubmit the job.

ABX2835E **The return code (2) indicates a redundant key.**

Explanation: A redundant key has been discovered.

System Action: ABXUTIL processing terminates.

User Response: Correct the error and resubmit the job.

ABX2836E **For a compound key, there should be a field name following the comma**

Explanation: The format of the RESULTSET statement is in error as described.

System Action: ABXUTIL processing terminates.

User Response: Correct the RESULTSET statement in the control language and resubmit the job.

ABX2837I **Processed *nnn* data records, selected *mmm*, inserted *xxx*, bypassed *yyy***

Explanation: At the end of processing, these statistics are produced. *nnn* is the number of records read from the input, *mmm* are those selected because either the predicates evaluated as true, or there were no predicates for this record type. *xxx* is the number of records inserted into ABXOUT due to THEN INSERT INTO table-name statements. *yyy* is the count of records skipped over due to predicate evaluations. That is, a bypassed record is one that is skipped (explicitly bypassed) as a result of your definition for the SKIP, LIMIT, THEN ONE OF control syntax declaratives.

User Response: This is an informational message and generally requires no action.

ABX2838I **Eliminated *nnn* record(s) because of UNIQUE requests**

Explanation: This is a summary message ate the end of processing. It recaps the count of records in the primary input that were eliminated due to the specified constraints for uniqueness.

User Response: This is an informational message and generally requires no user action.

ABX2839E **IMS control statements must be specified before the first SOURCEDBD or TARGETDBD statements.**

Explanation: You must ensure that all IMS control statements come before the first SOURCEDBD or TARGETDBD statement.

System Action: Processing stops.

User Response: Fix the control statements in error.

ABX2840E **A SOURCEDBD statement must precede an INPCB# statement..**

Explanation: You must ensure that the required SOURCEDBD statement is present in the control statement stream.

System Action: Processing stops.

User Response: Fix the control statements in error.

ABX2841E **A TARGETDBD statement must precede the OUTPCB# statement..**

Explanation: You must ensure that the required TARGETDBD statement is present in the control statement stream.

System Action: Processing stops.

User Response: Fix the control statements in error.

Chapter 5. Messages ABX8001--ABX9999

All messages generated by IBM File Export for z/OS have a severity code printed as the last character of the message ID. The severity codes are described in the following table:

Table 4. Error message severity codes

Severity Code	Description
I	Information only. No user action required.
W	Warning message. Results may not be as expected.
E	Error message. Some may be user-correctable, read the User Response to determine the course of action.
S	Severe error -- note the information in the message and contact your support personnel.

ABX8001E **(File *filename* line *dd*): Can't find REDEFINE symbol *name***

Explanation: A target of a REDEFINES clause cannot be found in the source file or the symbol cannot be redefined because it is not an immediate predecessor of the renaming declaration.

System Action: Processing continues with the next declaration.

User Response: Correct the error and rerun.

ABX8002E **(File *filename* line *dd*): COPY file not found *copy-file-name***

Explanation: The COPY file you have specified is not found.

System Action: The COPY file is not read. Processing continues with the next source statement.

User Response: Make sure the COPY file exists in a valid INCLUDE directory.

ABX8003E **(Line *dd*): Invalid PICTURE clause *picture***

Explanation: Because the PICTURE clause contains unexpected characters, the type for the variable cannot be determined.

System Action: Processing continues with the next declaration.

User Response: Correct the error and rerun.

ABX8004E **(File *filename* line *dd*): Bad REPLACING clause for COPY statement *copy_statement***

Explanation: The REPLACING clause cannot be understood.

System Action: The COPY file is not read. Processing continues with the next source statement.

User Response: Correct the error and rerun.

ABX8005E **(Line *dd*): elementary item *name* has type information; can't override with parent's type. *picture***

Explanation: The parent symbol's type information is not propagated to the child symbol.

System Action: The symbol's type information is not propagated.

User Response: Remove the type information from either the parent structure or the child.

ABX8006E **(File *filename* line *dd*): Invalid condition-name specification *statement***

Explanation: The condition name in the identified statement is invalid.

System Action: Processing continues with the next declaration

User Response: Correct the error and rerun.

ABX8007E **(File *filename* line *dd*): Invalid RENAMES statement *statement***

Explanation: The RENAMES in the identified statement is invalid.

System Action: Processing continues with the next declaration

User Response: Correct the error and rerun.

ABX8008E **(File *filename* line *dd*): #error preprocessor command encountered: *statement***

Explanation: A #error preprocessor command was encountered in the specified statement.

System Action: The COBOL parser stops and no data is returned.

User Response: Resolve the problem that created the error condition.

ABX8009E **(File *filename* line *dd*): *name* is not the parent of *child-name: statement***

Explanation: The level numbers of the record are inconsistent and the symbol *name* cannot be added to the current structure.

System Action: Processing continues with the next declaration

User Response: Correct the record declaration so that all level numbers are valid.

ABX8010E **(line *dd*): elementary data item *name*doesn't have a PICTURE clause**

Explanation: The data item named requires a PICTURE clause.

System Action: The type of the data item cannot be determined.

User Response: Add a PICTURE clause or a specific type to the data item.

ABX8011E **(File *filename* line *dd*): RENAMES data item *name* not found: *renames_statement***

Explanation: The target of the RENAMES cannot be found in the current structure.

System Action: Processing continues with the next declaration

User Response: Ensure that the RENAMED item is defined.

ABX8012E **(File *filename* line *dd*): RENAMES cannot be contained OCCURS group *namestatement***

Explanation: The COBOL standard prohibits this construct.

System Action: The statement is accepted but may not produce valid results.

User Response: Correct the statement to ensure valid data.

ABX8013E **(File *filename* line *dd*): Can't RENAME a level *nn* item *statement***

Explanation: The COBOL standard prohibits renaming of data items with certain level numbers.

System Action: Processing continues with the next declaration.

User Response: Correct the statement to ensure valid data.

ABX8014E **(line *dd*): RENAMES range for *name* contains OCCURS DEPENDING ON item *statement***

Explanation: The COBOL standard prohibits OCCURS DEPENDING ON clauses in the range of a RENAMES.

System Action: Processing continues with the next declaration.

User Response: Correct the statement in error.

ABX8015E **(File *filename* line *dd*): Unexpected end-of-file after the following line *statement***

Explanation: An end-of-file indicator was encountered unexpectedly. The parser expected more code but there were no more characters in the input file.

System Action: The parser stops.

User Response: Examine the statement identified and fix any errors.

ABX8016E **(File *filename* line *dd*): Unexpected level number *nn* on the following line *statement***

Explanation: The declaration shown is inconsistent with the current structure.

System Action: Processing continues with the next declaration.

User Response: Correct the statement in error.

ABX8017E **(File *filename* line *dd*): Unexpected *token*: *statement***

Explanation: The parser was expecting something other than *token*.

System Action: Processing continues with the next declaration.

User Response: Correct any errors and rerun.

ABX8018E **(line *dd*): REDEFINing object *name* cannot contain an OCCURS DEPENDING ON clause.**

Explanation: The COBOL standard prohibits redefining objects that contain an OCCURS DEPENDING ON clause.

System Action: The statement is accepted but may not be valid.

User Response: Correct the statement to ensure valid data.

ABX8019E **(line *dd*): Symbol *name* cannot be REDEFINEd because it contain an OCCURS DEPENDING ON clause.**

Explanation: The COBOL standard prohibits redefining objects that contain an OCCURS DEPENDING ON clause.

System Action: The statement is accepted but may not be valid.

User Response: Correct the statement to ensure valid data.

ABX8020W **(line *dd*): Symbol *name* is *nn* bytes larger than the object it redefines: *redefined-symbol***

Explanation: There is a mismatch in the length of the two objects.

System Action: Information only.

User Response: Ensure that the sizes of the redefined objects are correct.

ABX8021W **(File *filename* line *dd*): 01 record generated for subordinate declaration *statement***

Explanation: The first declaration in a code segment has a level number greater than 01.

System Action: An 01-level record is generated for the structure.

User Response: None.

ABX8022W **(File *filename* line *dd*): OCCURS cannot be specified on a level *nn* item: *statement***

Explanation: The COBOL standard prohibits data items with certain level numbers from having an OCCURS clause.

System Action: The declaration is accepted but may not be valid.

User Response: Correct the statement to ensure valid data.

ABX8023W **(File *filename* line *dd*): Declaration not ended by period: *statement***

Explanation: The COBOL standard requires all declarations to be terminated by a period.

System Action: Information only.

User Response: None

ABX8024W **(File *filename* line *dd*): REDEFINEd symbol cannot contain an OCCURS clause: *statement***

Explanation: The COBOL standard prohibits redefined symbols with an OCCURS clause.

System Action: The declaration is accepted but may not be valid.

User Response: Correct the statement to ensure valid data.

ABX8025S ***OPEN failed: source-file-name***

Explanation: The parser stopped due to an internal error.

System Action: The parser stops.

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

ABX8026S ***Fatal I/O error***

Explanation: The parser stopped because an I/O error occurred while reading from or writing to a file.

System Action: The parser stops.

User Response: Examine the JCL and job log and fix any reported errors.

ABX8099S ***Internal Error: Unexpected call of Expression::evaluate1()***

Explanation: The parser stopped due to an internal error.

System Action: The parser stops.

User Response: Notify your system programmer.

ABX8101E **(File *filename* line *dd*): Invalid date specification *date-specification***

Explanation: The date specification is invalid.

System Action: The specification is ignored.

User Response: Correct the date specification.

ABX8102E **(File *filename* line *dd*): Invalid #if expression *expression***

Explanation: A preprocessor expression could not be evaluated.

System Action: The preprocessor command is ignored.

User Response: Correct the error.

ABX8103E **(File *filename* line *dd*): Bad HANDLE type *handle***

Explanation: The HANDLE type is invalid.

System Action: Processing continues with the next declaration.

User Response: Correct the handle declaration

ABX8104E **(File *filename* line *dd*): file not found. *include-filename***

Explanation: A file whose name is *filename* could not be found.

System Action: The INCLUDE file is not read. Processing continues with the next statement.

User Response: Ensure that the INCLUDE file exists in a valid INCLUDE directory.

ABX8105E **(File *filename* line *dd*): Invalid string length *length-expression***

Explanation: The string length is invalid.

System Action: The string length cannot be determined.

User Response: Correct the string length specification.

ABX8106E **(File *filename* line *dd*): Invalid PICTURE specification *picture***

Explanation: The type for the data item cannot be determined.

System Action: The picture clause is ignored.

User Response: Fix the PICTURE clause or declare the variable with a valid type.

ABX8107E **(File *filename* line *dd*): can't find LIKE variable *name***

Explanation: A variable named in a LIKE clause cannot be found in the program context.

System Action: The preprocessor continues with the next declaration.

User Response: Ensure that the referenced structure is defined in a block within the current scope.

ABX8108E **(File *filename* line *dd*): DEFINED renaming not supported. *name* data can be accessed using the primary structure name *parent***

Explanation: Use of the DEFINED attribute to create an alias for a structure is not supported. Any reference to this structure by File Export must be done using the primary structure definition.

System Action: Processing continues with the next declaration.

User Response: Correct the error.

ABX8109E **(File *filename* line *dd*): Unexpected DEFAULT attribute *attribute***

Explanation: The listed attribute was unexpected.

System Action: Processing continues with the next declaration.

User Response: Correct the error.

ABX8110E **(File *filename* line *dd*): Bad DEFINE ORDINAL *statement***

Explanation: A syntax error was detected when parsing a define ordinal statement.

System Action: Processing continues with the next declaration.

User Response: Correct the error.

ABX8111E **(File *filename* line *dd*): Bad DIMENSION list *list***

Explanation: A syntax error was detected when parsing a dimension list.

System Action: Processing continues with the next declaration.

User Response: Correct the error.

ABX8112 **(File *filename* line *dd*): Redefinition of ordinal name *name***

Explanation: The named variable is already defined.

System Action: The redefinition is ignored.

User Response: None.

ABX8113E **(File *filename* line *dd*): unexpected end- of-file**

Explanation: The parser was expecting more characters, but the end of the source file was encountered.

System Action: The parser stops.

User Response: Correct the error.

ABX8114E **(File *filename* line *dd*): length expression for *name* not constant *length-expression***

Explanation: The parser cannot determine the length of the object.

System Action: No length is specified for the declared object.

User Response: Make the length a compile-time constant.

ABX8115E **(File *filename* line *dd*): LIKE object *name* is not a STRUCTURE or UNION**

Explanation: A name referenced in a LIKE clause is invalid.

System Action: Processing continues with the next declaration.

User Response: Ensure that the name is a structure or a union and is defined within the current program scope.

ABX8116E **(File *filename* line *dd*): #error preprocessor command encountered *statement***

Explanation: A #error preprocessor statement was evaluated.

System Action: The parser stops.

User Response: Correct the problem and rerun.

ABX8117E ***Error in PARMS list: \n parameter***

Explanation: the command line or command file parameter is not valid.

System Action: The parameter is ignored.

User Response: Correct the problem.

ABX8118E **(File *filename* line *dd*): error in RANGE() LIST OF default STATEMENT *statement***

Explanation: A syntax error was detected in the RANGE list of a DEFAULT statement.

System Action: Processing continues with the next statement.

User Response: Correct the error.

ABX8119E **(File *filename* line *dd*): invalid structure definition *statement***

Explanation: The statement listed is not valid in the current structure.

System Action: Processing continues with the next declaration.

User Response: Ensure that the structure is correctly defined.

ABX8120E **(File *filename* line *dd*): mismatched #endif**

Explanation: The command is incorrectly structured.

System Action: The #endif statement is ignored.

User Response: Ensure that all preprocessor commands are correctly structured.

ABX8121E **(File *filename* line *dd*): expecting RANGE() for DEFAULT statement; found: *statement***

Explanation: A syntax error was detected while parsing a RANGE list for a DEFAULT statement..

System Action: Processing continues with the next statement.

User Response: Correct the error.

ABX8122E **(File *filename* line *dd*): expecting *';'* found:*token***

Explanation: A semicolon character was expected but another token was found.

System Action: Processing continues with the next declaration.

User Response: Correct the error and rerun.

ABX8123E **(File *filename* line *dd*): no DEFINE ALIAS found for TYPE *name***

Explanation: A name referenced as a TYPE attribute is not defined within the current scope of the program.

System Action: Processing continues with the next declaration.

User Response: Correct the error.

ABX8124E **(File *filename* line *dd*): Not a declaration:*statement***

Explanation: The identified statement is not a declaration.

System Action: Processing continues with the next declaration.

User Response: Correct the declaration and rerun.

ABX8125E **(File *filename* line *dd*): ORDINAL member *name* VALUE *value* is less than the previous value.**

Explanation: Values explicitly defined for ordinal members ***must*** increase

System Action: The value is ignored.

User Response: Correct the error.

ABX8126E **(File *filename* line *dd*): invalid precision,scale expression *expression***

Explanation: A syntax error was detected while parsing a precision/scale expression for a variable.

System Action: Processing continues with the next declaration.

User Response: Correct the error.

ABX8127E **(File *filename* line *dd*): REFER variable *name* not declared in current structure**

Explanation: REFER variables must be declared previously within the current structure.

System Action: Processing continues with the next declaration.

User Response: Correct the error.

ABX8128E **(File *filename* line *dd*): unexpected attribute*attribute***

Explanation: A valid data attribute was expected, but another token was found.

System Action: Processing continues with the next declaration.

User Response: Correct the error.

ABX8129E **(File *filename* line *dd*): control stack underflow**

Explanation: The PL/I parser can process incomplete program segments but the block structure must be consistent. The error indicates that an end statement that has no corresponding beginning was found.

System Action: The parser stops.

User Response: Ensure that all block structure constructs are matched correctly.

ABX8130E **(File *filename* line *dd*): label *label* on END statement not defined; ignored**

Explanation: The identified label was not previously defined.

System Action: The END statement is processed as if it had no label specified (that is, a single block is exited).

User Response: Ensure that all block structure constructs are matched correctly and consistent.

ABX8131E **(line *dd*): No ORDINAL definition for variable *name***

Explanation: No DEFINE ORDINAL definition was found in the current program context for the named variable.

System Action: The variable declaration is ignored.

User Response: Correct the error.

ABX8198S **Parser exception *cause***

Explanation: An unrecoverable error has occurred.

System Action: The parser stopped due to an unexpected condition.

User Response: Contact support.

ABX8199S **Internal error. Abnormal termination requested from *filename*(line*dd*)**

Explanation: An internal error has occurred in the named file.

System Action: The parser stopped due to an unrecoverable error.

User Response: Note the file name and line number and description and contact support.

ABX9000E **'count' must be greater than 'size'**

Explanation: The value for 'count' is invalid.

System Action: The buffer is not initialized and left in a state that could not be used without further alterations.

User Response: To use this buffer, set the buffer equal to another buffer.

ABX9001E Buffer overrun: count must be less than size()-offset+1

Explanation: The value for 'count' is invalid.

System Action: No buffer is returned.

User Response: Provide a count that, when added to the offset, will not result in values greater than the size of the buffer.

ABX9002E 'pos' must be smaller than 'size'

Explanation: The value for 'pos' is invalid.

System Action: The position currently pointed to remains unchanged.

User Response: Provide a position that is less than the size of the buffer.

ABX9020E recordMaxSize may not be zero or less

Explanation: RecordMaxSize must be a positive integer.

System Action: The cursor is initialized incorrectly.

User Response: The cursor is corrupt and cannot be used.

ABX9021E Invalid cursor type value

Explanation: The cursor type is invalid.

System Action: The cursor is not opened.

User Response: Open with a proper cursor type specified.

ABX9022E Cursor cannot be opened: No records found with key=*keyValue*; or Cursor cannot be opened: Empty dataset

Explanation: The cursor key is invalid.

System Action: The cursor is not opened.

User Response: Provide a valid key or provide a key that returns a non-empty dataset.

ABX9023E No more records

Explanation: No records are available.

System Action: No action is taken.

User Response: The cursor has reached its end. No action can be taken.

ABX9024E Error opening cursor.

Explanation: Unexpected I/O error occurred.

System Action: The cursor is not opened correctly.

User Response: Check the error message stack for more information. If required, contact support.

ABX9025E Error opening cursor.

Explanation: Unexpected I/O error occurred.

System Action: The cursor is not opened correctly.

User Response: Check the error message stack for more information. If required, contact support.

ABX9026E Error opening cursor.

Explanation: Unexpected I/O error occurred.

System Action: The cursor is not opened correctly.

User Response: Check the error message stack for more information. If required, contact support.

ABX9027E Error opening cursor.

Explanation: Unexpected I/O error occurred.

System Action: The cursor is not opened correctly.

User Response: Check the error message stack for more information. If required, contact support.

ABX9028E Error reading next record.

Explanation: Unexpected I/O error occurred.

System Action: The next record is not read.

User Response: Check the error message stack for more information. If required, contact support.

ABX9029E Error reading next record.

Explanation: Unexpected I/O error occurred.

System Action: The next record is not read.

User Response: Check the error message stack for more information. If required, contact support.

ABX9030E Error reading next record.

Explanation: Unexpected I/O error occurred.

System Action: The next record is not read.

User Response: Check the error message stack for more information. If required, contact support.

ABX9031E Error resetting cursor position.

Explanation: Unexpected I/O error occurred.

System Action: The next record is read. However, the cursor is now in an uncertain state. Future function calls will result in unpredictable actions.

User Response: Check the error message stack for more information. The returned record is valid and may be used.

ABX9032E Error resetting cursor position.

Explanation: Unexpected I/O error occurred.

System Action: The cursor is now in an uncertain state. Future function calls will result in unpredictable actions.

User Response: Check the error message stack for more information. If required, contact support.

ABX9033E Error opening cursor.

Explanation: Unexpected I/O error occurred.

System Action: The cursor is not opened correctly.

User Response: Check the error message stack for more information. If required, contact support.

ABX9040E fopen failed

Explanation: The VSAM file is not opened, possibly due to incorrect parameters or perhaps the file does not exist.

System Action: The VSAM file is not opened.

User Response: Ensure that the parameters are correct or that the specified VSAM file exists.

ABX9041E fclose failed

Explanation: The VSAM file is not closed, possibly because it is not open or does not exist.

System Action: The VSAM file is not closed.

User Response: Ensure that the specified VSAM file exists and is open.

ABX9042E Cannot rewind on a closed file

Explanation: You are attempting a rewind on a closed file.

System Action: The VSAM file is not reset to its previous file.

User Response: Open the VSAM file.

ABX9043E Cannot execute at End-Of-File on a closed file

Explanation: You are attempting an operation on a closed file.

System Action: The VSAM file is not tested for end-of-file conditions.

User Response: Open the VSAM file.

ABX9044E Cannot execute readRecord on a closed file

Explanation: You are attempting an operation on a closed file.

System Action: A record is not read.

User Response: Open the VSAM file.

ABX9045E Read failed: End of file condition is true.

Explanation: The read operation failed.

System Action: A record is not read.

User Response: Reposition so that end of file condition is not true.

ABX9046E Read error occurred: *VSAM-specific error message*

Explanation: The read operation failed.

System Action: A record is not read.

User Response: The reason for the unexpected I/O error is specified in the VSAM-specific error message.

ABX9047E Cannot execute writeRecord on a closed file

Explanation: You are attempting an operation on a closed file.

System Action: The write is not performed.

User Response: Open the VSAM file.

ABX9048E Write error occurred: *VSAM-specific error message*

Explanation: The write operation failed.

System Action: A write is not performed.

User Response: The reason for the unexpected I/O error is specified in the VAM-specific error message.

ABX9049E **Cannot execute locateRecord on a closed file**

Explanation: You are attempting an operation on a closed file.

System Action: The locate is not performed.

User Response: Open the VSAM file.

ABX9050E **Cannot execute deleteRecord on a closed file**

Explanation: You are attempting an operation on a closed file.

System Action: The delete is not performed.

User Response: Open the VSAM file.

ABX9051E **fdelrec failed**

Explanation: An unexpected I/O error occurred.

System Action: The delete is not performed.

User Response: Review the error message stack or contact support.

ABX9052E **Record not found (key=*key_value*)**

Explanation: The specified record was not found.

System Action: The delete is not performed.

User Response: Ensure that you have specified the correct key value.

ABX9053E **Cannot execute updateRecord on a closed file**

Explanation: You are attempting an operation on a closed file.

System Action: The update is not performed.

User Response: Open the VSAM file.

ABX9054E **fupdate failed**

Explanation: An unexpected I/O error occurred.

System Action: The update is not performed.

User Response: Review the error message stack or contact support.

ABX9055E **Record not found (key=*key_value*)**

Explanation: The specified record was not found.

System Action: The update is not performed.

User Response: Ensure you have specified the correct key value.

ABX9056E **Cannot execute seek on a closed file**

Explanation: You are attempting an operation on a closed file.

System Action: The seek is not performed.

User Response: Open the VSAM file.

ABX9070E **useKey invalid**

Explanation: The useKey value is invalid.

System Action: The key is not resolved.

User Response: Specify a valid key value.

ABX9071E **Record specified for the key doesn't exist**

Explanation: The record you specified is invalid.

System Action: The key is not inserted into the VSAM file.

User Response: Specify a valid key value and ensure that the key is correctly populated.

ABX9072E **A key with name *keyName* for record with name *recordName* already exists**

Explanation: The key to be inserted already exists.

System Action: The key is not inserted into the VSAM file.

User Response: Specify a valid key value and ensure that the key is correctly populated.

ABX9073E **A key with the given key ID already exists.**

Explanation: The key to be inserted already exists.

System Action: The key is not inserted into the VSAM file.

User Response: Specify a valid key value and ensure that the key is correctly populated.

ABX9074E **A key with name *keyName* doesn't exist**

Explanation: The key to be inserted does not exist.

System Action: The key is not inserted into the VSAM file.

User Response: Specify a valid key value and ensure that the key is correctly populated.

ABX9075E The given record ID doesn't exist

Explanation: The record you specified does not exist.

System Action: The key's record ID is set to a nonexistent value.

User Response: Specify an existing record ID.

ABX9076E The given field ID is already associated with the key.

Explanation: This is an error with std. pair.

System Action: The field is not associated correctly with the key.

User Response: Correct the error.

ABX9077E The given field ID doesn't exist.

Explanation: The field ID does not exist.

System Action: The field is not associated with the key.

User Response: Specify an existing key.

ABX9078E The given field ID is not associated with this key.

Explanation: The given field ID may be associated with a key other than this key.

System Action: The field and key relationship is not changed.

User Response: Correct the error. You may have to remove an existing association to correct the problem.

ABX9079E The field that is part of this key is not found in the repository.

Explanation: The given field is not found in the repository.

System Action: The field associated with the key is not returned.

User Response: When associating a key and a field, ensure that the field exists. When removing a field, all key/field associations should be removed.

ABX9080E Initial look-up found the field as part of the key. But no records were returned from the repository.

Explanation: While the field was found, no records were returned.

System Action: The field associated with the key is not returned.

User Response: When associating a key and a field, ensure that the field exists. When removing a field, all key/field associations should be removed.

ABX9090E Predicate is not initialized.

Explanation: The predicate is not initialized.

System Action: The left value is not returned.

User Response: Set the left value of the predicate.

ABX9091E Predicate is not initialized.

Explanation: The predicate is not initialized.

System Action: The right value is not returned.

User Response: Set the right value of the predicate.

ABX9092E Value is not a predicate

Explanation: The specified value is not a predicate.

System Action: The predicate is not returned.

User Response: Use getConstValue or getFieldValue to retrieve the value; or set the value to a predicate.

ABX9093E Value is not a constant

Explanation: The specified value is not a constant.

System Action: The constant value is not returned.

User Response: Use getConstValue or getFieldValue to retrieve the value; or set the value to a constant value.

ABX9094E Value is not a field

Explanation: The specified value is not a field.

System Action: The field is not returned.

User Response: Use getConstValue or getFieldValue to retrieve the value; or set the value to a field.

ABX9095E Associated field is not valid

Explanation: The specified value is not valid.

System Action: The field specified does not exist.

User Response: Specify a valid field.

ABX9096E Predicate is not a tree type. Function call not appropriate.

Explanation: The function call was erroneous.

System Action: No operations are performed.

User Response: Do not perform any of the tree type operations since the current predicate will not support them. Correct the error.

ABX9097E **Predicate is not a string type. Function call not appropriate.**

Explanation: The function call was erroneous.

System Action: No operations are performed.

User Response: Do not perform any of the string type operations since the current predicate will not support them. Correct the error.

ABX9098E **Predicate initializing a Value must be a tree predicate.**

Explanation: You have erroneously specified a string predicate rather than a tree predicate. Value cannot be initialized using a string predicate.

System Action: Value is not initialized properly.

User Response: Correct the error by respecifying the predicate.

ABX9110E **updateRecord failed: Record must specify a unique primary key as useKey**

Explanation: The useKey must be primary and unique.

System Action: Update is not performed.

User Response: When updating, specify a key that is both primary and unique.

ABX9111E **A Unique Primary Key must be specified to delete a record**

Explanation: The key must be primary and unique.

System Action: Delete is not performed.

User Response: When using deleteRecord, specify a key that is both primary and unique.

ABX9112E **No records found with given key**

Explanation: No records have the key you specified.

System Action: Delete is not performed.

User Response: Provide a properly populated and correct object to be deleted.

ABX9113E **Could not find file for *tableName***

Explanation: There is no path for the specified record.

System Action: The system does not get the path.

User Response: Correct the error and rerun.

ABX9114E **Repository is not open**

Explanation: The repository is not open.

System Action: No operation is performed.

User Response: Open the repository.

ABX9115E **Repository is read only**

Explanation: The repository is not available to be written to.

System Action: No operation is performed.

User Response: To write to the repository, open the repository as read/write.

ABX9116E **Error adding record**

Explanation: An unexpected I/O error occurred.

System Action: The record is not added.

User Response: Review the message stack which may contain more information.

ABX9117E **Error updating record**

Explanation: An unexpected I/O error occurred.

System Action: The record is not updated.

User Response: Review the message stack which may contain more information.

ABX9118E **Error updating record**

Explanation: An unexpected I/O error occurred.

System Action: The record is not updated.

User Response: Review the message stack which may contain more information.

ABX9119E **Error updating record**

Explanation: An unexpected I/O error occurred.

System Action: The record is not updated.

User Response: Review the message stack which may contain more information.

ABX9120E **Error getting record**

Explanation: An unexpected I/O error occurred.

System Action: No records are obtained.

User Response: Review the message stack which may contain more information.

ABX9121E Error validating record

Explanation: An unexpected I/O error occurred. The system cannot locate the record.

System Action: No records are validated.

User Response: Review the message stack which may contain more information.

ABX9122E Error adding record

Explanation: An unexpected I/O error occurred.

System Action: No records are added.

User Response: Review the message stack which may contain more information.

ABX9123E Error getting record

Explanation: An unexpected I/O error occurred.

System Action: No records are obtained.

User Response: Review the message stack which may contain more information.

ABX9124E Error getting path

Explanation: An unexpected I/O error occurred.

System Action: The system is unable to obtain a path to the record.

User Response: Review the message stack which may contain more information.

ABX9125E Error getting path

Explanation: An unexpected I/O error occurred.

System Action: The system is unable to obtain a path to the record.

User Response: Review the message stack which may contain more information.

ABX9126E Error getting path

Explanation: An unexpected I/O error occurred.

System Action: The system is unable to obtain a path to the record.

User Response: Review the message stack which may contain more information.

ABX9127E Error deleting record

Explanation: An unexpected I/O error occurred.

System Action: No records are deleted.

User Response: Review the message stack which may contain more information.

ABX9128E Error deleting record

Explanation: An unexpected I/O error occurred.

System Action: No records are deleted.

User Response: Review the message stack which may contain more information.

ABX9129E Error deleting record

Explanation: An unexpected I/O error occurred.

System Action: No records are deleted.

User Response: Review the message stack which may contain more information.

ABX9130E Can't retrieve repository for a closed system

Explanation: The system is unable to retrieve the repository.

System Action: The repository you requested is not returned.

User Response: The repository must be opened.

ABX9131E No repository exists

Explanation: The system is unable to retrieve the repository.

System Action: The repository you requested is not returned.

User Response: The repository must be created.

ABX9140E Not a valid key for file

Explanation: The key was not properly resolved

System Action: The key is not properly resolved.

User Response: Provide a valid useKey.

ABX9141E Parent store does not exist

Explanation: The system is unable to locate the parent store.

System Action: The file is not inserted.

User Response: Ensure that the parent store exists and is correctly populated.

ABX9142E **File of type: *file_type* and name: *file_name* already exists**

Explanation: A file of the same file name and file type has already been inserted.

System Action: The file is not inserted.

User Response: Correct the error and rerun.

ABX9143E **File of type: *file_type* and name: *file_name* does not exist**

Explanation: A file you are trying to delete does not exist.

System Action: The file is not deleted.

User Response: Correct the error and rerun.

ABX9144E **Cannot delete this file because a record selector relationship exists**

Explanation: A file you are trying to delete has a record selector relationship in place.

System Action: The file is not deleted.

User Response: Remove all record selector relationships before trying to delete a file.

ABX9145E **File of type: *file_type* and name: *file_name* cannot be deleted because it is part of at least one link. Must first delete all links that refer to this file**

Explanation: A file you are trying to delete is linked to other files in the repository

System Action: The file is not deleted.

User Response: Remove all links before trying to delete a file.

ABX9146E **File of type: *file_type* and name: *file_name* does not exist**

Explanation: A file you are trying to update does not exist.

System Action: The file is not updated.

User Response: Correct the error and rerun.

ABX9147E **Parent store does not exist**

Explanation: Parent store is set to a store that does not exist.

System Action: No action is taken.

User Response: Ensure that the parent store exists and is correctly populated.

ABX9148E **Link to record of name:*record_name* does not exist**

Explanation: You are attempting to link to a record that does not exist.

System Action: The specified record is not updated.

User Response: Link the record and the file before attempting an update.

ABX9149E **Record does not exist**

Explanation: You are attempting to create a selector relationship to a record that does not exist.

System Action: The selector relationship is not created.

User Response: Specify an existing record.

ABX9150E **File does not exist**

Explanation: You are attempting to create a selector relationship to a file that does not exist.

System Action: The selector relationship is not created.

User Response: Specify an existing file that is properly populated.

ABX9151E **Selection criteria between record and file already exist**

Explanation: You are attempting to define a selector relationship that already exists.

System Action: The pre-existing selector relationship is not changed.

User Response: Delete the previous selector relationship to add a new one or update the previous one.

ABX9152E ***previousMessage* The following exception was thrown and subsequent rollback failed, leaving the repository in an unexpected state**

Explanation: A selection criterion that was partially added is not completely rolled back because of an unexpected error.

System Action: Partial rollback occurred.

User Response: The repository may contain data that is invalid and unnecessary.

ABX9153E *previousMessage* **The previous exception was thrown and resulted in invalid data in the repository**

Explanation: A selection criterion that was partially added is not completely rolled back because of an unexpected error.

System Action: Partial rollback occurred.

User Response: The repository may contain data that is invalid and unnecessary. Correct the error before proceeding.

ABX9154E **Record does not exist**

Explanation: You are attempting to alter selection criteria for a record that does not exist.

System Action: No selection criteria are altered.

User Response: Specify an existing record.

ABX9155E **File does not exist**

Explanation: You are attempting to alter selection criteria for a file that does not exist.

System Action: No selection criteria are altered.

User Response: Specify an existing file that is properly populated.

ABX9156E **Record is not related to the file**

Explanation: You are attempting to alter selection criteria for a record that is not related to a given file.

System Action: No selection criteria are altered.

User Response: Use a record that is related to the file.

ABX9157E **File does not exist**

Explanation: You are attempting to alter selection criteria for a file that does not exist.

System Action: No selection criteria are altered.

User Response: Specify an existing file that is properly populated.

ABX9158E **File has no selection criteria**

Explanation: You are attempting to delete selection criteria for a file that has no selection criteria.

System Action: No selection criteria are altered.

User Response: You cannot delete record selectors if none exist for the intended file.

ABX9159E **File does not exist**

Explanation: You are attempting to alter selection criteria for a file that has no selection criteria.

System Action: No selection criteria are returned.

User Response: Specify an existing file that is properly populated.

ABX9160E **Record could not be found**

Explanation: You are attempting to alter selection criteria for a record that could not be found. A file may have a selection criterion that selects a record that no longer exists.

System Action: No selection criteria are returned.

User Response: Correct the error.

ABX9161E **Record does not exist**

Explanation: You are attempting to alter selection criteria for a record that does not exist.

System Action: No selection criteria are altered.

User Response: Provide a record that is related to the file.

ABX9162E **File does not exist**

Explanation: You are attempting to alter selection criteria for a file that does not exist.

System Action: No selection criteria are altered.

User Response: Specify an existing file that is properly populated.

ABX9163E **Update cannot be performed because no relationship exists.**

Explanation: You are attempting to update selection criteria for a relationship that does not exist.

System Action: No selection criteria are altered.

User Response: To update selection criteria, a selection relationship must already exist. Create such a relationship if necessary.

ABX9164E *previousMessage* **The previous exception was thrown and suresulted in invalid data in the repository**

Explanation: A selection criterion that was partially added is not completely rolled back because of an unexpected error.

System Action: Partial rollback occurred.

User Response: The repository may contain data that is invalid and unnecessary. Correct the error before proceeding.

ABX9165E Parent store does not exist

Explanation: The parent store does not exist.

System Action: The returning store has invalid data.

User Response: The parent store must be set with a store that exists.

ABX9166E Parent store does not exist

Explanation: The parent store does not exist.

System Action: The returning store has invalid data.

User Response: The parent store must be set with a store that exists.

ABX9180E Invalid key value

Explanation: The specified key value is invalid.

System Action: The key is not properly resolved.

User Response: Provide a valid useKey.

ABX9181E Link field already exists.

Explanation: You have specified a link field that already exists in the repository.

System Action: The link field is not inserted.

User Response: Populate the link field correctly.

ABX9182E Link does not exist.

Explanation: You have specified a link field for a link that does not exist.

System Action: The link field is not inserted.

User Response: Ensure the link exists before adding a link field.

ABX9183E Field does not exist.

Explanation: You have specified a link field for a field that does not exist.

System Action: The link field is not inserted.

User Response: Ensure that the field being linked exists.

ABX9184E Cannot insert child link field because the corresponding field, named *modelField* is not a member of the link's child record.

Explanation: A child link field cannot be inserted for the reason given.

System Action: The link field is not inserted.

User Response: Ensure that the model field exists.

ABX9185E Cannot insert parent link field because the corresponding field, named *modelField* is not a member of the link's parent record.

Explanation: A parent link field cannot be inserted for the reason given.

System Action: The link field is not inserted.

User Response: Ensure that the model field exists.

ABX9186E Invalid link field type

Explanation: The link field type is invalid.

System Action: The link field is not inserted.

User Response: Correct the error to ensure a valid link field type.

ABX9187E Link field does not exist.

Explanation: The link field you want to delete does not exist.

System Action: The link field is not deleted.

User Response: The link field must exist in order to be deleted

ABX9188E Link Fields may not be updated

Explanation: You are attempting to update a link field. Link fields may not be updated.

System Action: The link field is not updated.

User Response: Do not attempt to update this field.

ABX9200E Invalid key value

Explanation: The specified key value is invalid.

System Action: The key is not properly resolved.

User Response: Provide a valid useKey.

ABX9201E Link Field MtoM already exists

Explanation: You are attempting to insert an MtoM link field with the same characteristics as one that already exists.

System Action: The link field MtoM is not inserted.

User Response: Do not attempt to insert a field if an identical one already exists.

ABX9202E Link does not exist

Explanation: You are attempting to insert an MtoM link field when a link does not exist.

System Action: The link field MtoM is not inserted.

User Response: Create a link before adding an MtoM link field.

ABX9203E Parent link field does not exist

Explanation: You are attempting to insert an MtoM link field when a parent link field does not exist.

System Action: The link field MtoM is not inserted.

User Response: Create a parent link field.

ABX9204E Link field MToM to be deleted does not exist

Explanation: You are attempting to delete an MtoM link field when it does not exist.

System Action: The link field MtoM is not deleted.

User Response: The link field MToM must exist in order to be deleted.

ABX9205E Link field MToM record may not be updated

Explanation: You may not update an MToM link field record.

System Action: The link field MtoM may never be updated

User Response: Do not attempt to update the link field MToM record.

ABX9206E Child link field does not exist

Explanation: You are attempting to insert a child link field that does not exist.

System Action: The link field MtoM is not inserted.

User Response: Create a child link field.

ABX9220E Invalid key value

Explanation: The specified key value is invalid.

System Action: The key is not properly resolved.

User Response: Provide a valid useKey.

ABX9221E Link already exists

Explanation: You are attempting to insert a link with the same characteristics as one that already exists.

System Action: The link is not inserted.

User Response: Do not attempt to insert a link if an identical one already exists.

ABX9222E Parent file does not exist

Explanation: You are attempting to insert a link when the parent file does not exist.

System Action: The link is not inserted.

User Response: Provide a valid parent file.

ABX9223E Parent file exists but does not contain the parent record.

Explanation: You are attempting to insert a link when the parent record does not exist.

System Action: The link is not inserted.

User Response: Provide a valid parent file that contains the parent record.

ABX9224E Child file does not exist

Explanation: You are attempting to insert a link when the child file does not exist.

System Action: The link is not inserted.

User Response: Provide a valid child file.

ABX9225E Child file exists but does not contain the child record.

Explanation: You are attempting to insert a link when the child record does not exist.

System Action: The link is not inserted.

User Response: Provide a valid child file that contains the child record.

ABX9226E Link to be deleted does not exist.

Explanation: You are attempting to delete a link that does not exist.

System Action: The link is not deleted.

User Response: The link must exist in order to be deleted.

ABX9227E Links may not be updated

Explanation: You are attempting to update a link, which should not be done.

System Action: The link is not updated

User Response: Do not attempt to update a link, since a link can not be updated.

ABX9228E Parent file does not exist

Explanation: You are attempting to alter a parent file that does not exist.

System Action: The parent file is unchanged.

User Response: Provide a valid parent file.

ABX9229E Parent record does not exist

Explanation: You are attempting to alter a parent record that does not exist.

System Action: The parent record is unchanged.

User Response: Provide a valid parent record

ABX9230E Child file does not exist

Explanation: You are attempting to alter a child file that does not exist.

System Action: The child file is unchanged.

User Response: Provide a valid child file.

ABX9231E Child record does not exist

Explanation: You are attempting to alter a child record that does not exist.

System Action: The child record is unchanged.

User Response: Provide a valid child record

ABX9232E Parent field's vector is empty

Explanation: The parent field's vector is empty.

System Action: No relationship is added.

User Response: Populate the parent field vector.

ABX9233E Null pointer found in parent field's vector

Explanation: You may have passed a null pointer. The repository may contain corrupted links.

System Action: A child link field is added and some parent fields and link fields MToM may have been added.

User Response: You may have to review and clean up the repository. You can prevent this condition by never passing null pointers.

ABX9234E Child field's vector is empty

Explanation: The child field's vector is empty.

System Action: No relationship is added.

User Response: Populate the child field vector.

ABX9235E Null pointer found in child field's vector

Explanation: You may have passed a null pointer. The repository may contain corrupted links.

System Action: A parent link field is added and some child fields and link fields MToM may have been added.

User Response: You may have to review and clean up the repository. You can prevent this condition by never passing null pointers.

ABX9236E Cannot add a link exit parm to this link because this link does not have an exit defined.

Explanation: You are attempting to add a link exit parameter to a link that has no defined exit.

System Action: The link exit parameter is not added.

User Response: Define an exit before adding an exit parameter.

ABX9237E Cannot add a link exit parm to this link because this link does not have an exit defined.

Explanation: You are attempting to add a link exit parameter to a link that has no defined exit.

System Action: The link exit parameter is not added.

User Response: Define an exit before adding an exit parameter.

ABX9238E A Link Exit Parm associated with the given field already exists.

Explanation: You are attempting to add a link exit parameter that is identical to one that already exists.

System Action: The link exit parameter is not added.

User Response: A link exit parameter can be added only once.

ABX9239E Field does not exist

Explanation: You are attempting to add a link exit parameter to a field that does not exist.

System Action: The link exit parameter is not added.

User Response: A link exit parameter must use an exiting field.

ABX9240E Cannot get this link's link exit parms because this link does not have an exit defined.

Explanation: You are attempting to acquire parameters for a link without a defined exit.

System Action: The link exit parameter is not returned.

User Response: Only links with a defined exit can return a link exit parameter.

ABX9241E The field that is part of this link exit is not found in the repository.

Explanation: You are attempting to obtain link exit parms when a field is missing from the repository.

System Action: The field is not returned.

User Response: The link exit parameter should be set up to use a valid and existing field.

ABX9242E Initial lookup found the field as part of the link exit but no records were returned from the repository.

Explanation: You are attempting to obtain link exit parms when a field may be invalid.

System Action: The field is not returned.

User Response: The link exit parameter should be set up to use a valid and existing field.

ABX9260E Length of string must not exceed *maxSize*

Explanation: Your string exceeds the maximum permitted length.

System Action: The value is not set.

User Response: Set the value of the padded string using a string that has length less than or equal to the defined maximum size.

ABX9270E Length of string must not exceed *maxSize*

Explanation: Your string exceeds the maximum permitted length.

System Action: The value is not set.

User Response: Set the value of the padded string using a string that has length less than or equal to the defined maximum size.

ABX9280E Invalid key value (*keyValue*)

Explanation: The useKey value is invalid.

System Action: The key is not resolved.

User Response: Specify a valid key value.

ABX9281E Record already exists

Explanation: You are attempting to add a record that already exists.

System Action: The record is not inserted.

User Response: You cannot add a record if it already exists.

ABX9282E Record with the same name already exists

Explanation: You are attempting to add a record with a name that already exists.

System Action: The record is not inserted.

User Response: You cannot add a record having the same name as a pr-existing record.

ABX9283E Record specifies itself as a subrecord for a field group, but the corresponding field doesn't exist.

Explanation: A parent record for this field does not exist.

System Action: The record is not inserted.

User Response: Ensure that the parent field exists.

ABX9284E Record to be deleted does not exist

Explanation: You are attempting to delete a record that does not exist.

System Action: The record is not deleted.

User Response: You cannot delete a record that does not exist.

ABX9285E **Record cannot be deleted because it has a record selector relationship**

Explanation: You are attempting to delete a record that has a record selector relationship with another record.

System Action: The record is not deleted.

User Response: Remove any record selector relationships before attempting a delete.

ABX9286E **A record that specifies itself as a subrecord for a field group may not be deleted while the corresponding field continues to refer to it.**

Explanation: You are attempting to delete a record that is referred to be a corresponding field

System Action: The record is not deleted.

User Response: Update the parent field so that it no longer has the subject record as a subrecord.

ABX9287E **This record cannot be deleted because it is part of at least one link. Must first delete all links that refer to this record.**

Explanation: You are attempting to delete a record that is linked to another record.

System Action: The record is not deleted.

User Response: Remove all links before attempting to delete.

ABX9288E **Record to be updated must already exist**

Explanation: You are attempting to update a record that does not exist.

System Action: The record is not updated.

User Response: Ensure that the record exists before attempting to update.

ABX9289E **Record specifies itself as a subrecord for a field group but the corresponding field does not exist.**

Explanation: You are attempting to update a record whose field group does not exist.

System Action: The record is not updated.

User Response: Ensure that the parent field exists before attempting to update.

ABX9290E **AbxRecord:: addField(abxField&)error: group field not expected**

Explanation: An unexpected field was encountered during processing

System Action: The child field is not added.

User Response: Use the appropriate addField function if the field is a group field.

ABX9291E **AbxRecord:: addField(abxField&,AbxRecord&)E: subrec type group field expected**

Explanation: An unexpected field was encountered during processing

System Action: The child field is not added.

User Response: Use the appropriate addField function if the field is not a group field.

ABX9292E **AbxRecord:: addField(abxField&)error: record ID does not exist**

Explanation: A record ID that was expected was not found

System Action: The child field is not added.

User Response: Ensure that the record being passed exists.

ABX9293E **Field does not exist**

Explanation: The specified field does not exist.

System Action: The parent field remains unchanged.

User Response: Ensure that the specified field exists.

ABX9294E **Parent field does not exist**

Explanation: The parent field no longer exists and the repository may have invalid data.

System Action: The parent field is not returned.

User Response: Review the repository contents and correct any errors.

ABX9295E **Parent field does not exist**

Explanation: The parent field no longer exists and the repository may have invalid data.

System Action: The parent field is not returned.

User Response: Review the repository contents and correct any errors.

ABX9296E **No parent file exists**

Explanation: The parent file does not exist.

System Action: No files are returned.

User Response: Ensure that the parent files are valid.

ABX9298E **AbxRecord:: addField(abxField&,AbxRecord&)E: redefine type group field expected**

Explanation: An unexpected field was encountered during processing

System Action: The child field is not added.

User Response: Use the appropriate addField function if the field is not a group field.

ABX9299E **AbxRecord:: addField(abxField&,AbxRecord&)Error: record *record_name*does not exist**

Explanation: A record to be added does not exist.

System Action: The child field is not added. Some of the field's child records may now point to the field that was not added.

User Response: Use the appropriate addField function if the field is not a group field. Review the contents of the repository and correct any errors.

ABX9300E **Invalid redefine name. A field with the same name already exists under the record.**

Explanation: A redefine with the specified name already exists. The redefine name must be unique and no other field should have the same name.

System Action: The redefine is not created.

User Response: Correct the error.

ABX9301E **Field to be added to redefine does not exist in the repository.**

Explanation: A field to be added to the redefine is not available in the repository.

System Action: The field is not added to the redefine.

User Response: Before adding a field to a redefine, the field must be added to the repository. Ensure the field being passed is correctly populated.

ABX9302E **The redefine name has not been created under the specified record.**

Explanation: A redefine with the specified name does not exist.

System Action: The field is not added to the redefine.

User Response: Correct the error.

ABX9303E **The redefine was not created correctly.**

Explanation: The redefine does not have an appropriate subrecord.

System Action: The field is not added to the redefine.

User Response: Correct the error.

ABX9304E **The redefine name specifies a field that is not a redefine field.**

Explanation: The specified redefine exists, but it is not defined as a redefine field.

System Action: The field is not added to the redefine.

User Response: Correct the error.

ABX9305E **Field to be added to redefine does not exist in the repository.**

Explanation: Before removing a field from a redefine, the field must be added to the repository. Make sure the field being passed is correctly populated.

System Action: The field is not removed from the redefine.

User Response: Correct the error.

ABX9306E **The redefine name has not been created under the specified record.**

Explanation: A redefine with the specified name does not exist.

System Action: The field is not removed from the redefine.

User Response: Correct the error.

ABX9307E **The redefine name specifies a field that is not a redefine field.**

Explanation: A redefine with the specified name exists but is not a redefine field.

System Action: The field is not removed from the redefine.

User Response: Correct the error.

ABX9308E **The redefine was not created correctly.**

Explanation: The redefine does not have an appropriate subrecord.

System Action: The field is not removed from the redefine.

User Response: Correct the error.

ABX9309E **The field specified is not part of the redefine.**

Explanation: The specified field is not defined as part of a redefine.

System Action: The field is not removed from the redefine.

User Response: Ensure the specified field to be removed is populated correctly and that it is part of the redefine.

ABX9310E **Path is not initialized**

Explanation: The required path has not been initialized.

System Action: The path is not added.

User Response: Provide VSAMKSDSFile pointer that is not null.

ABX9311E **Path *name* does not exist**

Explanation: The named path does not exist.

System Action: The path is not returned.

User Response: Ensure that you are passing the correct path name.

ABX9320E **Cluster *name* is not found**

Explanation: The identified cluster cannot be found.

System Action: The cluster is not returned.

User Response: Ensure that you are passing the correct name.

ABX9321E **Cluster *name* not found**

Explanation: The named cluster cannot be found

System Action: The cluster is not removed.

User Response: Ensure that you are passing the correct cluster name.

ABX9330E **Repository corrupted *numReturned* records returned**

Explanation: The record returned is invalid. There is a repository corruption with zero or more than one connections between one file and one record.

System Action: The record returned is invalid.

User Response: Review your repository and correct the error.

ABX9331E **Parent store does not exist**

Explanation: You have specified a parent store that does not exist.

System Action: The parent store is not changed.

User Response: Ensure that you are using an existing valid store.

ABX9332E **Parent file does not exist**

Explanation: You have specified a parent file that does not exist.

System Action: The parent file is not changed.

User Response: Ensure that you are using an existing valid file.

ABX9340E **GroupField::setDependsOnField error: given field does not exist**

Explanation: You have specified a field in a Depends On that does not exist.

System Action: Depends On is not set.

User Response: Ensure that you are passing a valid field.

ABX9341E **Invalid key value (*keyValue*)**

Explanation: The useKey value is invalid.

System Action: The key is not resolved.

User Response: Specify a valid key value.

ABX9342E **Field with name *fieldName*, type *fieldType*, and position *fieldPos* already exists**

Explanation: The specified field already exists.

System Action: The field is not inserted.

User Response: You can add a field only if it does not already exist.

ABX9343E **Parent record does not exist**

Explanation: The specified parent record does not exist.

System Action: The record is not inserted.

User Response: You must specify a valid existing parent record.

ABX9344E **Either the subrecord specified for this repeating group field does not exist or no subrecord was specified.**

Explanation: The specified subrecord may not exist or may not have been specified.

System Action: The field is not inserted.

User Response: Specify a valid record as a subrecord.

ABX9345E **The depends-on field specified for this repeating group field does not exist.**

Explanation: The specified field does not exist.

System Action: The field is not inserted.

User Response: Specify a valid depends-on value.

ABX9346E **Field with name *fieldName*, type *fieldType*, and position *fieldPos* does not exist**

Explanation: You are attempting to delete the specified field which does not exist.

System Action: The field is not deleted.

User Response: A field must exist in order to be deleted.

ABX9347E **Cannot delete this field because a record selector relationship exists.**

Explanation: The specified field cannot be deleted while it is in record selector relationship.

System Action: The field is not deleted.

User Response: Remove all record selector relationships before deleting.

ABX9348E **Cannot delete this field because it is referenced in a predicate.**

Explanation: The specified field cannot be deleted while it is referenced in a predicate.

System Action: The field is not deleted.

User Response: Remove the field from all record selectors that reference it.

ABX9349E **Cannot delete this field because it is referenced in a predicate.**

Explanation: The specified field cannot be deleted while it is referenced in a predicate.

System Action: The field is not deleted.

User Response: Remove the field from all record selectors that reference it.

ABX9350E **Field with name *fieldName*, type *fieldType*, and position *fieldPos* is part of a link. Must first define the link.**

Explanation: You are attempting to delete a record for which a link exists.

System Action: The field is not deleted.

User Response: Remove all links before deleting.

ABX9351E **Field with name *fieldName*, type *fieldType*, and position *fieldPos* does not exist**

Explanation: You are attempting to update the specified field which does not exist.

System Action: The field is not updated.

User Response: A field must exist in order to be updated.

ABX9352E **Parent record does not exist**

Explanation: The parent record you want to update does not exist.

System Action: The field is not updated.

User Response: The record must exist in order to be updated.

ABX9354E **Either the subrecord specified for this repeating group field does not exist or no subrecord was specified.**

Explanation: The specified subrecord may not exist or may not have been specified.

System Action: The field is not updated.

User Response: Specify a valid record as a subrecord.

ABX9355E **The depends-on field specified for this repeating group field does not exist.**

Explanation: The specified field does not exist.

System Action: The field is not updated.

User Response: Specify a valid depends-on value.

ABX9356E Error removing subrecords

Explanation: One or more of the subrecords were not reset and still point to the field to be deleted.

System Action: The system was unable to remove the subrecords.

User Response: The repository may be corrupt and may need to be fixed.

ABX9357E Parent record does not exist

Explanation: The parent record you want to alter does not exist.

System Action: The parent record is not changed.

User Response: Set the parent record to a valid value.

ABX9358E Not a group, no subgroup

Explanation: The group you have so defined is not a group since it has no subgroup.

System Action: The record is not updated.

User Response: The field can only update records if it is a group field and has subrecords.

ABX9359E Record not marked as a subrecord for this field.

Explanation: The record is not marked as a subrecord for the field.

System Action: The record is not updated.

User Response: Provide a record that is a subrecord of the field.

ABX9360E Record not a subrecord

Explanation: The record is not a subrecord.

System Action: The record is not updated.

User Response: Pass a record that is a subrecord.

ABX9361E Not a group, no subgroup

Explanation: The group you have so defined is not a group since it has no subgroup.

System Action: The record is not returned.

User Response: The field can only update records if it is a group field and has subrecords.

ABX9363E Record does not exist

Explanation: The record for which you want to add selection criteria does not exist.

System Action: Selection criteria are not added.

User Response: Set the record to a valid value.

ABX9364E Field does not exist

Explanation: The field for which you want to add selection criteria does not exist.

System Action: Selection criteria are not added.

User Response: Ensure that the field is correctly populated.

ABX9365E Selection criteria between record and file already exist.

Explanation: Selection criteria between the record and file already exist.

System Action: Selection criteria are not added.

User Response: The selection criteria must be unique and must not preexist.

ABX9366E *previousMessage* **The following exception was thrown and subsequent rollback failed leaving the repository in an unexpected state.**

Explanation: An exception occurred that may have corrupted the repository

System Action: A selection criterion that was partially added is not completely rolled back due to an unexpected error.

User Response: The repository may contain data that is invalid and unnecessary. Correct the problem.

ABX9367E *previousMessage* **The previous exception was thrown and resulted in invalid data in the repository**

Explanation: An exception occurred that may have corrupted the repository

System Action: A selection criterion that was partially added is not completely rolled back due to an unexpected error.

User Response: The repository may contain data that is invalid and unnecessary. Correct the problem.

ABX9368E Record does not exist

Explanation: The record for which you want to delete selection criteria does not exist.

System Action: Selection criteria are not deleted

User Response: Set the record to a valid value.

ABX9369E Field does not exist

Explanation: The field for which you want to delete selection criteria does not exist.

System Action: Selection criteria are not deleted

User Response: Populate the field correctly.

ABX9370E Record is not related to the file

Explanation: The record for which you want to delete selection criteria is not related to the file.

System Action: Selection criteria are not deleted

User Response: Selection criteria can only be deleted if they exist.

ABX9371E Field does not exist

Explanation: The field for which you want to delete selection criteria does not exist.

System Action: Selection criteria are not deleted

User Response: Populate the field correctly.

ABX9372E File has no selection criteria

Explanation: The file for which you want to delete selection criteria does not exist.

System Action: Selection criteria are not deleted

User Response: Selection criteria can only be deleted if they exist.

ABX9373E Field does not exist

Explanation: The field for which you want to obtain selection criteria does not exist.

System Action: Selection criteria are not returned

User Response: Populate the field correctly.

ABX9374E Record could not be found

Explanation: The record for which you want to obtain selection criteria does not exist.

System Action: Selection criteria are not returned.

User Response: A field may contain selection criteria for a record that does not exist.

ABX9375E Record does not exist

Explanation: The record for which you want to update selection criteria does not exist.

System Action: Selection criteria are not updated

User Response: Specify a valid record.

ABX9376E Field does not exist

Explanation: The field for which you want to update selection criteria does not exist.

System Action: Selection criteria are not updated

User Response: Populate the field correctly.

ABX9377E Update cannot be performed because no relationship exists

Explanation: The field you were attempting to update has no relationship specified.

System Action: Selection criteria are not updated

User Response: A selector relationship must exist in order to update the relationship.

ABX9378E ***previousMessage*** **The previous exception was thrown and resulted in invalid data in the repository**

Explanation: An error may have corrupted the repository.

System Action: A selection criterion that was partially added is not completely rolled back because of an unexpected error.

User Response: The repository may contain unnecessary or invalid data. Review the repository and correct the error.

ABX9379E Parent record does not exist

Explanation: The record does not exist.

System Action: The parent record is not returned.

User Response: The parent record must be a valid and existing record.

ABX9380E Parent record does not exist

Explanation: The record does not exist.

System Action: The parent record is not returned.

User Response: The parent record must be a valid and existing record.

ABX9381E Invalid field

Explanation: The field you are attempting to delete is invalid.

System Action: The field is not deleted.

User Response: Populate the field correctly.

ABX9390E Invalid key value for ABXFileRecMToM)

Explanation: The useKey value is invalid.

System Action: The key is not properly resolved.

User Response: Specify a valid key value.

ABX9391E Relationship between the file and record already exists

Explanation: You are attempting to insert a relationship that already exists.

System Action: File record MToM is not inserted.

User Response: A duplicate of this relationship exists and there cannot be more than one defined.

ABX9392E Store does not exist

Explanation: The store you want to write to does not exist.

System Action: File record MToM is not inserted.

User Response: Provide a valid store.

ABX9393E File does not exist

Explanation: The file you want to write to does not exist.

System Action: File record MToM is not inserted.

User Response: Provide a valid file.

ABX9394E Record does not exist

Explanation: The record you want to write to does not exist.

System Action: File record MToM is not inserted.

User Response: Provide a valid record.

ABX9395E Relationship between file and record does not exist

Explanation: The relationship you want to delete does not exist

System Action: File record MToM is not deleted.

User Response: A file record cannot be deleted if it does not exist.

ABX9396E Relationship between file and record can never be updated

Explanation: The relationship you want to update may not be updated.

System Action: File record MToM is not updated.

User Response: A file record can never be updated.

ABX9410E Invalid key value (*keyValue*)

Explanation: The useKey value is invalid.

System Action: The key is not resolved.

User Response: Specify a valid key value.

ABX9411E The two stores are not related.

Explanation: You are attempting to update unrelated stores.

System Action: The old store is not updated.

User Response: Ensure that the old store and the new store are correctly populated to identify that they are the same.

ABX9412E A store with the same type and name as the specified new store already exists.

Explanation: You are attempting to create a duplicate store.

System Action: The old store is not updated.

User Response: The new store must be given a name and type that, when combined, do not match a store that already exists.

ABX9413E The two records are not related.

Explanation: You are attempting to update unrelated records.

System Action: The old record is not updated.

User Response: Ensure that the old record and the new record are correctly populated to identify that they are the same.

ABX9414E Record's new name already exists

Explanation: You are attempting to create a duplicate record.

System Action: The old record is not updated.

User Response: If the record name is being updated, it must not be updated to a name that already exists.

ABX9430E Invalid key value (*keyValue*)

Explanation: The useKey value is invalid.

System Action: The key is not resolved.

User Response: Specify a valid key value.

ABX9431E The record already exists.

Explanation: You are attempting to specify a duplicate record selector.

System Action: The record selector is not added.

User Response: The record selector has already been added and cannot be added again.

ABX9432E The record selector does not exist.

Explanation: You are attempting to delete a record selector that does not exist.

System Action: The record selector is not deleted.

User Response: The record selector must exist in order to be deleted.

ABX9433E The record does not exist.

Explanation: You are attempting to update a record selector that does not exist.

System Action: The record selector is not updated.

User Response: The record selector must exist in order to be updated.

ABX9450E Invalid key value (*keyValue*)

Explanation: The useKey value is invalid.

System Action: The key is not resolved.

User Response: Specify a valid key value.

ABX9451E Record type relation already exists.

Explanation: This relationship already exists.

System Action: The record type relation is not added.

User Response: The record type relation already exists and cannot be added again.

ABX9452E File/Field does not exist

Explanation: The file or field does not exist.

System Action: The record type relation is not added.

User Response: The file or field that the record type relation is using must exist in order to add the record type relation.

ABX9453E Record does not exist

Explanation: The record does not exist.

System Action: The record type relation is not added.

User Response: The record that the record type relation is using must exist in order to add the record type relation

ABX9454E Record selector does not exist

Explanation: The record selector does not exist.

System Action: The record type relation is not added.

User Response: The record selector being referenced must exist in order to add the record type relation.

ABX9455E Record type relation does not exist

Explanation: The record type relation you want to delete does not exist.

System Action: The record type relation is not deleted.

User Response: The record type relation must exist in order to be deleted.

ABX9456E Cannot update record type relation

Explanation: The system is unable to update the record type relation.

System Action: The record type relation is not updated.

User Response: The record type relation must exist in order to be updated.

ABX9470E Invalid key value (*keyValue*)

Explanation: The useKey value is invalid.

System Action: The key is not resolved.

User Response: Specify a valid key value.

ABX9471E Associate record selector for the predicate does not exist.

Explanation: The associate record selector for the predicate does not exist.

System Action: The selector predicate is not added.

User Response: The record selector for the predicate must exist to be added.

ABX9472E **There may be only one AROOT for a given record selector.**

Explanation: More than one AROOT exists for a record selector.

System Action: The selector predicate is not added.

User Response: Ensure that the selector predicate has only one root if in tree form.

ABX9473E **The record to be deleted must exist.**

Explanation: You are attempting to delete a record that does not exist.

System Action: The selector predicate is not deleted.

User Response: Ensure that the selector predicate exists in order to delete it.

ABX9474E **A predicate may not be updated directly.**

Explanation: You are attempting to update a predicate directly.

System Action: The selector predicate is not updated.

User Response: Do not update a selector predicate directly.

ABX9475E **No root predicate available.**

Explanation: You are attempting to obtain a root predicate that is unavailable.

System Action: The predicate is not returned.

User Response: The selector predicate may be stored incorrectly in the repository. Review the repository and correct the error.

ABX9476E **Could not find a child predicate.**

Explanation: You are attempting to obtain a child predicate that is unavailable.

System Action: The predicate is not returned.

User Response: The selector predicate may be stored incorrectly in the repository. Review the repository and correct the error.

ABX9477E **Could not find a child predicate.**

Explanation: You are attempting to obtain a child predicate that is unavailable.

System Action: The predicate is not returned.

User Response: The selector predicate may be stored incorrectly in the repository. Review the repository and correct the error.

ABX9490E **Invalid key value (*keyValue*)**

Explanation: The useKey value is invalid.

System Action: The key is not resolved.

User Response: Specify a valid key value.

ABX9491E **Store of type: *storeType* and name: *storeName* already exists.**

Explanation: You are attempting to add a store that already exists.

System Action: The store is not added.

User Response: The store already exists. You cannot add it again.

ABX9492E **Store of type: *storeType* and name: *storeName* does not exist.**

Explanation: You are attempting to delete a store that does not exist.

System Action: The store is not deleted.

User Response: You cannot delete a store that does not exist.

ABX9493E **Store of type: *storeType* and name: *storeName* does not exist.**

Explanation: You are attempting to update a store that does not exist.

System Action: The store is not updated.

User Response: You cannot update a store that does not exist.

ABX9494E **The two files are not related.**

Explanation: You are attempting an operation on two files that are not related.

System Action: The old file is not updated.

User Response: Correctly populate the old file and the new file so that they are internally related.

ABX9495E **A file with the new file's parent store, file type, and file name already exists. This combination must be unique.**

Explanation: You are attempting an operation on a file that is a duplicate.

System Action: The old file is not updated.

User Response: Correctly identify the new file so that it is not a duplicate of an existing file.

ABX9510E Platform not supported.

Explanation: You are attempting to run a program under an operating system that is not supported.

System Action: Major timekeeping errors occur. The program terminates.

User Response: Since this program can only run under MVS, ensure that you are running under MVS.

ABX9520E GUID generation error

Explanation: An unexpected error occurred while generating the GUID.

System Action: The GUID is not returned.

User Response: Correct the error.

ABX9521E getClock() error: time of day cannot be determined.

Explanation: The system clock is in error. There was an unexpected error when generating the time stamp.

System Action: The time of day is not returned.

User Response: Correct the error.

ABX9522E The specified ID is too long

Explanation: You specified an ID that was too long.

System Action: The ID is not changed.

User Response: Pass a string whose length is less than or equal to 8 bytes.

ABX9540E The redefine name has not been created under the specified record.

Explanation: A redefine with the specified name does not exist.

System Action: The fields under the redefine are not returned.

User Response: Correct the error.

ABX9541E The redefine name specifies a field that is not a redefine field.

Explanation: The specified redefine exists, but it is not defined as a redefine field.

System Action: The fields under the redefine are not returned.

User Response: Correct the error.

ABX9542E The redefine was not created correctly.

Explanation: A redefine appears to be missing at least one subrecord.

System Action: The fields under the redefine are not returned.

User Response: Correct the error.

ABX9900S Incorrect program usage

System Action: The program terminates. No data is written to the Repository or the log file.

User Response: Invoke the program with correct positional program arguments.

ABX9901S Parser initialization error

System Action: SAX parser failed to initialize.

User Response: Contact support personnel.

ABX9902S XML error

System Action: SAX parser terminated due to XML error.

User Response: Validate XML source.

ABX9903W SAX parser warning

System Action: Cause of the warning is logged and program continues.

User Response: Review the log to locate the cause of the warning in the XML source.

ABX9904E SAX parser error

System Action: Cause of the error is logged and program continues.

User Response: Review the log to locate the cause of the error in the XML source.

ABX9905S SAX parser severe error

System Action: Cause of the error is logged and program terminates.

User Response: Review the log to locate the cause of the error in the XML source.

ABX9906S **ABX Repository severe error**

System Action: Program terminates.

User Response: Review the log for a description. Contact Support if necessary.

ABX9907S **ABX Repository XML Loader severe error**

System Action: Program terminates.

User Response: Review the log for a description. Contact Support if necessary.

ABX9908S **Unexpected severe error**

System Action: Program terminates.

User Response: Contact Support if necessary.

ABX9910I **Repository system opened successfully.**

System Action: The Repository has opened.

User Response: None.

ABX9911I **Repository system closed successfully.**

System Action: The Repository has closed.

User Response: None.

ABX9920E **Unexpected XML start tag**

System Action: The program ignores the XML element and continues.

User Response: See the log for a description. Since this element is skipped, validate your repository data.

ABX9921E **Unexpected XML end tag**

System Action: The program ignores the XML element and continues.

User Response: See the log for a description. Since this element is skipped, validate your repository data.

ABX9922I **No parse-unit name specified**

System Action: Program generates a parse-unit name and continues.

User Response: The parse-unit is never stored in the repository. Information-only message.

ABX9930E **No record type specified for top-level record**

System Action: No record type is assigned. Program continues.

User Response: Record may be written to the Repository with NULL type value. Validate repository data.

ABX9931I **Record successfully written to the repository.**

System Action: The record has been written to the repository.

User Response: See log for description.

ABX9950E **A field element cannot be at root-level in XML source**

System Action: Program ignores this element and continues.

User Response: See log for description. Since this element is skipped, validate repository data.

ABX9951I **Duplicate field name found for given parent record.**

System Action: Program generates field name for internal use only.

User Response: See log for description.

ABX9952E **Cannot retrieve field dependency for given field.**

System Action: Program cannot resolve field dependency. No 'depends on' attribute value assigned for the field. Program continues.

User Response: See log for description. Note that repository data may be corrupt. Validate repository data.

ABX9953E **Repository cannot set field dependency for given field.**

System Action: Repository cannot assign field dependency. No 'depends on' attribute value assigned for the field. Program continues.

User Response: See log for description. Note that repository data may be corrupt. Validate repository data.

ABX9954I **Simple field successfully written to repository**

System Action: Simple field written to repository.

User Response: See log for description.

ABX9970E **A group element cannot be at root-level in XML source.**

System Action: Program ignores this element and continues.

User Response: See log for description. Since this element is skipped, validate repository data.

ABX9971W **No group type specified for given group field**

System Action: Program defaults group field type to type 'SUBREC' and continues.

User Response: See log for description. Validate repository data.

ABX9972W **Specified group type unrecognized for given group field**

System Action: Program defaults group field type to type 'SUBREC' and continues.

User Response: See log for description. Validate repository data.

ABX9980W **Fewer than two subrecords were specified for a given redefine group field**

System Action: Program continues. Redefine group field may be written to repository with less than two subrecords.

User Response: See log for description. Validate repository data.

ABX9981I **Redefine group field successfully written to repository.**

System Action: Redefine group field successfully written to repository.

User Response: See log for description.

ABX9985E **More than one subrecord specified for given subrec group field**

System Action: Only the first specified subrecord is associated with the subrec group field. Any subsequent subrecords will not be associated with the subrec group field. Program continues.

User Response: See log for description. Note that repository data may be corrupt. Validate repository data.

ABX9986E **No subrecord specified for given subrec group field**

System Action: Subrec group field will not be written to the repository. Program continues.

User Response: See log for description. Note that repository data may be corrupt. Validate repository data.

ABX9987I **Subrec group field successfully written to repository**

System Action: Subrec group field written to repository.

User Response: See log for description.

ABX9990E **Specified algorithm string value exceeds Repository-defined max algorithm string length.**

System Action: Program truncates specified value and continues.

User Response: See log for description. Note that repository data may be corrupt. Validate repository.

ABX9999E **Internal logic error *(module name)* -- *description***

Explanation: An internal logic error has occurred in the named module.

System Action: The step terminates.

User Response: Note the module name and description and contact support.

Chapter 6. Return Codes

This chapter provides information about IBM File Export return codes.

Repository XML Loader Return Codes

Table 5. Repository XML Loader Return Codes

Severity Code	Description
0 -- Information/OK	Program terminated normally without warnings or errors.
4 -- Warning	Program terminated normally with warnings.
8 -- Error	Program terminated normally with errors. Data in the repository may be invalid.
12 -- Severe Error	Program terminated abnormally

Part 3. Appendixes

Appendix A. Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10594-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this publication 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 as your own risk.

Licensees of this program who want to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue

San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains 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 contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

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

Advanced Peer-to-Peer Networking	IBM
CICS/ESA	Redbooks
DB2	OS/390
DB2 Universal Database	z/OS

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

Glossary

The following cross-references are used in this glossary:

Contrast with. This refers to a term that has an opposed or substantively different meaning.

Synonym for. This indicates that the term has the same meaning as a preferred term, which is defined in its proper place in the dictionary.

Synonym with. This is a backward reference from a defined term to all other terms that have the same meaning.

See. This refers the reader to multiple-word terms that have the same last word.

See also. This refers the reader to terms that have a related, but not synonymous, meaning.

A

abend. Abnormal end of task.

abend reason code. A 4-byte hexadecimal code that uniquely identifies a problem with DB2. A complete list of DB2 abend codes and their explanations is contained n *DB2 Messages and Codes*.

abnormal end of task (abend). Termination of a task, job, or subsystem because of an error condition that recovery facilities cannot resolve during execution.

active log. The portion of the DB2 log to which log records are written as they are generated. The active log always contains the most recent log records, whereas the archive log holds those records that are older and no longer fit on the active log.

archive log. The portion of the DB2 log that contains log records that have been copied from the active log.

B

backward log recovery. The fourth and final phase of restart processing during which DB2 scans the log in a backward direction to apply UNDO log records for all aborted changes.

bind. The process by which the output from the DB2 precompiler is converted to a usable control structure (which is called a package or an application plan). During the process, access paths to the data are selected and some authorization checking is performed.

bootstrap data set (BSDS). A VSAM data set that contains name and status information for DB2 as well as RBA range specifications, for all active and archive log data sets. It also contains passwords for the DB2 directory and catalog, and lists of conditional restart and checkpoint records.

buffer pool. Main storage that is reserved to satisfy the buffering requirements for one or more table spaces or indexes.

C

catalog. In DB2, a collection of tables that contains descriptions of objects such as tables, views, and indexes.

catalog table. Any table in the DB2 catalog.

clustering index. An index that determines how rows are physically ordered in a table space.

column. The vertical component of a table. A column has a name and a particular data type (for example, CHARACTER, DECIMAL, or INTEGER).

command. A DB2 operator command or a DSN subcommand. A command is distinct from an SQL statement.

customization. The process of describing optional changes to defaults of a software program that is already installed on the system and configured so that it can be used. Contrast with Configuration.

customize. To describe the system, the devices, programs, users, and user defaults for a particular data processing system or network. Contrast with Configure.

D

DASD. Direct access storage device.

data definition name. The name of a data definition (DD) statement that corresponds to a data control block containing the same name.

DB2 catalog. Tables that are maintained by DB2 and that contain descriptions of DB2 objects, such as tables, views, and indexes.

direct access storage device. A device in which access time is independent of the location of the data.

DSN. (1) The default DB2 subsystem name. (2) The name of the TSO command processor of DB2. (3) The first three characters of the DB2 module and macro names.

I

image copy. An exact reproduction of all or part of a table space. DB2 provides utility programs to make full image copies (to copy the entire table space) or incremental image copies (to copy only those pages that have been modified since the last image copy).

index. A set of pointers in DB2 that are logically ordered by the values of a key. Indexes can provide faster access to data and can enforce uniqueness on the rows in a table.

index space. A DB2 page set that is used to store the entries of one index.

ISPF. Interactive System Productivity Facility.

J

JCL. Job control language. Synonym with job control language.

job control language. JCL. A control language that is used to identify a job to an operating system and to describe the job's requirements. Synonym with JCL.

job profile. Customized settings that describe the jobs you want to run.

L

lock. A means of controlling concurrent events or access to data. DB2 locking is performed by the IRLM (Inter-Region Lock Manager).

lock duration. The interval over which a DB2 lock is held.

lock escalation. The promotion of a lock from a row, page, or LOB lock to a table space lock because the number of page locks that are concurrently held on a given resource exceeds a present limit.

locking. The process by which the integrity of DB2 data is ensured. Locking prevents concurrent users from accessing inconsistent data.

log. A collection of records that describe the events that occur during DB2 execution and that indicate their sequence. The information thus recorded is used for recovery in the event of a failure during DB2 execution.

logical unit of work (LUW). The processing that a program performs between synchronization points.

logical unit of work identifier (LUWID). A name that uniquely identifies a thread within a network. This name consists of a fully-qualified LU network name, an LUW instance number, and an LUW sequence number.

log initialization. The first phase of restart processing during which DB2 attempts to locate the current end of the log.

log record sequence number (LRSN). A number that DB2 generates and associates with each log record. DB2 also uses the LRSN for page versioning. The LRSNs that a particular DB2 data sharing group generates form a strictly increasing sequence for each DB2 log and a strictly increasing sequence for each page across the DB2 group.

P

page. A unit of storage within a DB2 table space (4 KB, 8 KB, 16 IB, or 32 KB) or an index space (4 KB). In a table space, a page contains one or more rows of a table. In a LOB table space, a LOB value can span more than one page, but no more than one LOB value is stored on a page.

S

Source. A source is a file (either DB2 table, IMS database, VSAM file, or sequential file) from which data is extracted for the purpose of creating test data.

T

Target. A target is a file that is created using IBM File Export, and is based on the data that is extracted from the source file and your specification of the target output.

Readers' Comments—We'd Like to Hear from You

IBM File Export for z/OS
Messages and Codes
Version 1 Release 1

Publication No. SC31-6871-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	❑	❑	❑	❑	❑

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	❑	❑	❑	❑	❑
Complete	❑	❑	❑	❑	❑
Easy to find	❑	❑	❑	❑	❑
Easy to understand	❑	❑	❑	❑	❑
Well organized	❑	❑	❑	❑	❑
Applicable to your tasks	❑	❑	❑	❑	❑

Please tell us how we can improve this book:

Thank you for your response. May we contact you? ❑ Yes ❑ No

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

Name _____ Address _____

Company or Organization _____

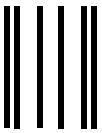
Phone No. _____

Cut or Fold Along Line

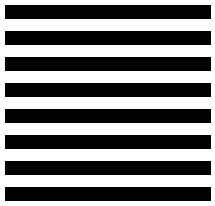
SC31-6871-01

International Business Machines Corporation
IBM Corporation --Silicon Valley Laboratory
Department DTX/E269
555 Bailey Avenue
San Jose, CA
U.S.A.
95141 - 9989

Fold and Tape **Please do not staple** Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

SC31-6871-01
INTERNATIONAL BUSINESS MACHINES CORP
Silicon Valley Laboratory -- Department DTX/E269
555 Bailey Avenue
San Jose, CA 95141-9989

Fold and Tape **Please do not staple** Fold and Tape

Cut or Fold Along Line

Program Number: 5697-I12

SC31-6871-01