

Advantage™ CA-PanAudit® Plus

Messages Guide

3.0



Computer Associates®

This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. ("CA") at any time.

This documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of CA. This documentation is proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, the user may print a reasonable number of copies of this documentation for its own internal use, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software of the user will have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to return to CA the reproduced copies or to certify to CA that same have been destroyed.

To the extent permitted by applicable law, CA provides this documentation "as is" without warranty of any kind, including without limitation, any implied warranties of merchantability, fitness for a particular purpose or noninfringement. In no event will CA be liable to the end user or any third party for any loss or damage, direct or indirect, from the use of this documentation, including without limitation, lost profits, business interruption, goodwill, or lost data, even if CA is expressly advised of such loss or damage.

The use of any product referenced in this documentation and this documentation is governed by the end user's applicable license agreement.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227.7013(c)(1)(ii) or applicable successor provisions.

© 1984-2001 Computer Associates International, Inc., One Computer Associates Plaza, Islandia, New York 11749. All rights reserved.

All trademarks, trade names, service marks, or logos referenced herein belong to their respective companies.

Contents

Chapter 1: Messages

Diagnostic Message Format	1-1
Message ID	1-1
Diagnostic Message	1-1
Message Supplement	1-2
A001 - A046 Runtime Messages	1-2
B001 - B313 Syntax Messages	1-8
Asterisk Prefixed Runtime Messages	1-40
DB2 Diagnostic Messages	1-41
CA-PanAudit Plus Message Identification	1-49
CA-PanAudit Plus Message Format	1-49
Message ID	1-49
Diagnostic Message	1-49
Message Supplement	1-49
PAP100 - PAP324 Runtime Messages	1-50

Chapter 2: Debugging Techniques

Utility Programs	2-2
EZTPX01	2-2
EZTPX03	2-2
EZTPX04	2-2
EZTPX05	2-3
Compile Listing	2-3
Header	2-3
Statement Listing	2-4
Listing Control Statements	2-5
LIST	2-5
NEWPAGE	2-6
SKIP	2-6
PUSH	2-6

POP	2-6
Example	2-6
Diagnostic Format	2-7
Parameter (PARM) Listing	2-7
CLIST	2-7
DMAP	2-8
PMAP	2-11
XREF	2-12
Execution Listing	2-12
Abnormal Termination	2-13
Syntax Errors	2-13
Execution Errors	2-13
Error Analysis Report	2-14
Cause A Data Exception	2-16
Analyzing the Report	2-16
System-Defined Fields and Return Codes	2-19
General Fields	2-19
File Fields	2-20
Report Fields	2-23
Screen Fields	2-24
Symbol References	2-25
Reserved Words	2-27

Index

Messages

CA-PanAudit Plus runtime messages are all prefixed with PAP. Messages with other prefixes are from CA-Easytrieve. CA-Easytrieve Plus provides a comprehensive set of diagnostic messages that describes the types of errors that can occur when a program is compiled and executed.

Diagnostic messages fall into four groups that describe:

- Operational errors — A001 through A046 prefix
- Program syntax errors — B001 through B313 prefix
- IDD interface errors — asterisk prefix
- SQL interface errors — no message prefix

Diagnostic Message Format

All CA-Easytrieve Plus diagnostic messages conform to the same format:

```
*****XNNN X-----X - S-----S
      Message   Diagnostic   Message
      ID        Message      Supplement
```

Message ID

The message ID, a four-character code, identifies each error message. The first character of the message ID designates the type of error. Message IDs beginning with the character A identify operational errors. Errors beginning with a B identify program syntax errors.

Diagnostic Message

The diagnostic message is a description of the detected error.

Message Supplement

The message supplement is optional, depending on the diagnostic message and context of the message. If possible, CA-Easytrieve Plus provides a message supplement to identify the particular object that is in error.

A001 - A046 Runtime Messages

Following is a list of operational diagnostic messages with brief explanations for each.

A001 FILE OPEN ERROR – *file-name*

The operating system detected an error while attempting to open the indicated file. The file remains unopened, and the job is terminated. Validate the existence and characteristics of the actual file. Ensure that the characteristics of the file match those parameters specified on the FILE statement.

A002 INVALID BLOCK SIZE – *file-name*

The FILE statement for the indicated file specified an incorrect value for the block size. Any of the following can cause this error:

- The device assigned to the file cannot support a block size as large as that specified.
- FULLTRK was specified for a file contained on a device other than disk.
- The required block size value is not specified.

A003 INSUFFICIENT STORAGE AVAILABLE

The region where CA-Easytrieve Plus is running is too small. If possible, a supplemental message is provided that defines the necessary storage type, amount, and the identity of the CA-Easytrieve Plus routine that requested the storage. You can usually correct this problem by increasing the partition size or region size. If the supplemental message is EZTVFM, allow VFM to go to a disk or increase the VFM core storage.

A004 CATASTROPHIC ERROR IN MODULE – *modname*

If another *Annn* message preceded this error message, correct the problem that the preceding message described and rerun your program. If the error persists, contact Computer Associates Technical Support for assistance.

A005 I/O ERROR – *file-name*

The operating system detected an input/output error for the indicated file. For OS/390 and z/OS, the contents of the SYNADAF buffer are appended to the message to provide additional debugging information.

A006 PROGRAM INTERRUPT – CODE *x*

CA-Easytrieve Plus intercepted a program interrupt, codes 1 through 11. See the chapter “[Debugging Techniques](#)” for further information and the associated debugging techniques on this message.

A007 TABLE INPUT IS NOT IN SEQUENCE – *file-name*

The indicated external table file is not in ascending sequence by the defined argument (ARG), or the table has a duplicate key. This message applies to external tables only.

A008 TOO MANY TABLE ENTRIES – *file-name*

There are more table entries in the indicated external table file than specified on the related FILE statement. Recompile the program after increasing the value in the file’s TABLE parameter.

A009 REPORT PROCESSING ABORTED DUE TO ERROR

Report processing aborted due to a program interrupt or sort error. CA-Easytrieve Plus error analysis (see the chapter “[Debugging Techniques](#)”) or the sort program provides an error message describing the cause of termination.

A010 INVALID FILE REFERENCE – *file-name*

A field name was referenced in a file that had no active record. The file might be closed, at end of file, or in synchronized file mode with no active record. The variable *file name* specifies the name of the file.

A011 VSAM - type ERROR - FILE *file-name* - CODE *nnn(xx)*

An uncorrectable VSAM error occurred. CA-Easytrieve Plus displays the name of the file and the error code that appears in both decimal (*nnn*) and hexadecimal (*xx*) formats. One of four possible types of VSAM errors are possible:

- GENCB indicates that an error occurred while building or modifying a VSAM control block.
- OPEN indicates that an error occurred while attempting to open the specified file.
- LOGICAL indicates that an illogical I/O request was made (such as a random keyed access to an ESDS).
- PHYSICAL indicates that a physical I/O error occurred for the specified file.

These error codes *nnn(xx)* are described in the IBM *VSAM Programmer's Guide*.

A012 INVALID LENGTH – *file-name*

The record length of the specified output file is incorrect. The record length must be in the range 1 to 65535.

A013 WRONG LENGTH RECORD – *file-name*

The current record length for the specified input file is incorrect. Verify that the FILE statement parameters correctly describe the actual file.

A014 PREMATURE TERMINATION DUE TO PREVIOUS ERROR(S)

Some previously identified error caused a termination of the CA-Easytrieve Plus execution.

A015 UNEXPECTED DBMS ERROR – *file-name* - FEEDBACK CODE = *cc* - PARM LIST IS AT - *xxxxxx*

An unexpected condition occurred during the path processing of a database. This is probably a result of incorrect definition of the database to CA-Easytrieve Plus or a failure of the database system. CA-Easytrieve Plus provides the feedback code and the address of the parameter list for the last reference to the database system. The CA-Easytrieve Plus snap dump or the operating system dump (whichever is available) contains the parameter List.

The feedback code *cc* is described in the appropriate *DBMS Programmer's Guide*.

A016 LOAD ERROR – *program-name*

An error occurred while attempting to load *program-name*.

A017 DBD IS NOT FOUND WITHIN PSB

The DBD specified in the FILE statement cannot be located in the PSB that was passed to CA-Easytrieve Plus through DLI-IMS/DB.

A018 DLI-IMS/DB UPDATE IS NOT ALLOWED – *function*

The system was installed with the option UPDTDLI=NO, and a variable function request of DLET, ISRT, or REPL was made.

A020 VSAM UPDATE IS NOT ALLOWED

The system was installed with the option UPDTVVS=NO, and an attempt was made to extend a file by using the CREATE or UPDATE option of the FILE statement.

A021 SELECTABLE UNIT IS NOT AVAILABLE

The selectable option that you requested is not available at this time. Check with your system programmer to ensure the CA-Easytrieve Plus installation was complete and that you are pointing to all necessary execution libraries.

A022 INSUFFICIENT MEMORY FOR VFM SPACE ALLOCATION

There is insufficient memory in the region or partition to allocate the amount of memory the VFMSPACE parameter specified.

A023 VFM SPACE ALLOCATION EXCEEDED

The amount of memory the VFMSPACE parameter specified was not enough to contain all the virtual file data. Memory is assumed when no JCL for VFM is found.

A024 ERROR OPENING THE DATASET – *rc*

An irrecoverable error occurred opening the VFM data set.

The supplemental text *rc* is printed only when executing the program under CMS. It is the return code from the FSOPEN macro. See the *CMS Command and Macro Reference* guide for an explanation of this return code.

A025 UNABLE TO RESTART THE VFM DATASET

The CA-Easytrieve Plus Restart Control Program received a non-zero return code from the Restart Initialization call to VFM. A more detailed message may come from VFM.

A026 RESTART ID IS INVALID – *supplemental*

IMS did not accept the Checkpoint ID you specified in the JCL. The supplemental message gives further details.

Note: If IMS did not find the Checkpoint ID specified on the restart log tape, IMS abends with a U0102 return code.

A027 ERROR RESTORING DATA FOR RESTART – *supplemental*

The CA-Easytrieve Plus Restart Control Program found unreconcilable discrepancies between the checkpointed data or control blocks and the restarting data or control blocks. This occurs when the CA-Easytrieve Plus program is illegally modified between checkpoint and restart. The supplemental message gives details.

A028 ERROR WRITING VFM HEADER BLOCK – *rc*

An irrecoverable error occurred writing the file descriptor block during the initialization of the EZTVFM data set.

When the program is executed under OS/390, z/OS, or VSE, an A005 message precedes this message in the listing.

The supplemental text *rc* is printed only when the program is executed under CMS. It is the return code from the FSWRITE macro. See the *CMS Command and Macro Reference* for an explanation of this return code.

A029 ERROR WRITING A VFM DATA BLOCK – *rc*

An irrecoverable error occurred during an attempt to write to the EZTVFM data set.

When the program is executed under OS/390, z/OS, or VSE, an A005 message precedes this message in the listing.

The supplemental text *rc* is printed only when the program is executed under CMS. It is the return code from the FSWRITE macro. See the *CMS Command and Macro Reference* for an explanation of this return code.

A030 ERROR READING A VFM DATA BLOCK – *rc*

An irrecoverable error occurred during an attempt to read from the EZTVFM data set.

When the program is executed under OS/390, z/OS, or VSE, an A005 message precedes this message in the listing. The supplemental text *rc* is printed only when the program is executed under CMS. It is the return code from the FSWRITE macro. See the *CMS Command and Macro Reference* for an explanation of this return code.

A031 CHECKPOINT-ID EXCEEDS 99,999,999

A maximum of 99,999,999 checkpoints are allowed.

A032 DATA BASE COULD NOT BE REPOSITIONED

The database could not be repositioned. Possible reasons are paths changed, or database segments were added or deleted.

A044 INVALID OPTION TABLE PARAMETER

You specified a CA-Easytrieve Plus option that is not valid in your operating system environment. Verify your CA-Easytrieve Plus options.

A045 SORT TERMINATED DUE TO NON-ZERO RETURN CODE – *value*

SORT did not run successfully. The value parameter indicates the SORT return code.

A046 SQL – supplemental

An SQL error occurred. The supplemental message gives detailed information on the SQL error condition as returned from the SQL interface.

B001 - B313 Syntax Messages

The following group of messages describe errors detected while syntax checking the CA-Easytrieve Plus source program. The optional supplemental messages in this group specify:

- Additional diagnostic information
- The specific object in error
- The word most likely in error

B001 LITERALS CANNOT EXCEED 254 CHARACTERS

The literal exceeds the maximum length of 254 characters.

B002 INVALID HEXADECIMAL CHARACTER STRING

A hexadecimal character string is constructed incorrectly. (See the CA-Easytrieve Plus *Reference Guide*.)

B003 EXPECTED CONTINUATION NOT RECEIVED

A statement continuation was indicated, but end of file on the CA-Easytrieve Plus source input file was detected.

B004 REPORT EXCEEDS PAGESIZE

The number of lines required to print a detail record plus any title or heading exceeds the PAGESIZE value.

B005 INSTALLATION ERROR -- CALL CA-EASYTRIEVE PLUS SUPPORT

An installation error occurred. Contact Computer Associates Technical Support for assistance.

B006 MACRO SYSTEM -{PAN|PDS|USER|VSAM} -- *additional diagnostic information*

An error occurred in the macro system library interface. The type of library interface routine is indicated. A supplemental message, supplied by the indicated macro library interface, describes the particular problem.

B007 INVALID IF/END-IF PAIRING – *count*

An END-IF statement is missing or incorrectly placed. Every IF statement must be delimited by an END-IF statement. The value, *count*, indicates how many IF statements were not delimited by an END-IF. Look for previous error messages B173 or B185 indicating that the nesting level was in error.

B010 INVALID BLOCKSIZE – *file-name*

An inconsistent value for the block size is specified on the FILE statement for the indicated file.

For fixed-length records, the block size must be an integral multiple of the record length.

For variable-length records, the minimum block size is the record length plus four (4).

B011 TABLE INPUT IS NOT IN SEQUENCE

The current INSTREAM table file is not in ascending sequence by the argument (ARG), or the table has a duplicate key.

**B012 DUPLICATE NAME – {*file-name* }
 {*field-name* }
 {*report-name* }**

The indicated name is a duplicate. The name can be a file name, field name, or report name.

B013 ASSIGNMENT OPERATOR MISSING – *field*

An assignment statement was being processed, but an equal sign (=) was not found in the second position. *Field* indicates the symbol found where the equal sign was expected.

B014 UNABLE TO RECOGNIZE STATEMENT – *word*

The indicated statement is not recognizable as a CA-Easytrieve Plus source or control statement. The optional supplemental message indicates the invalid statement. If the supplemental message is not present, the entire job stream is unrecognizable, and the input is flushed.

B015 SELECTABLE UNIT IS NOT AUTHORIZED

An attempt to use a function of a CA-Easytrieve Plus selectable unit was made; however, your site does not have that selectable unit.

B016 INVALID OR CONFLICTING KEYWORD – *word*

The indicated word is not valid for the associated statement, or it is inconsistently used.

B017 VSAM UPDATE IS NOT ALLOWED

The system was installed with the option UPDTVVS=NO, and a WRITE or PUT statement was issued for a VSAM file, or a PUT statement was issued for a file other than FILE file name, VS(CREATE).

B018 DLI - IMS/DB UPDATE IS NOT ALLOWED – *function*

The system was installed with the option UPDTDLI=NO, and a literal function code of DLET, ISRT, or REPL was specified.

B019 ADJUSTMENT NOT ALLOWED OR INVALID – *word*

For a DEFINE statement, an overlay redefinition for a field is in error. The redefining field must be in the redefined field.

For a TITLE or LINE statement, the space adjustment is invalid. A negative adjustment cannot cause line item overlay. A positive adjustment cannot extend beyond the end of the logical line.

B020 PARAMETER MUST BE NUMERIC – *word*

The indicated word must be numeric.

B021 PARAMETER IS TOO LARGE – *word*

The value of the indicated word is too large. See the statement description for the valid range.

B022 IDMS UPDATE IS NOT ALLOWED

The system was installed with the option UPDTIDM=NO, and an IDMS command of MODIFY, STORE, ERASE, CONNECT, or DISCONNECT was specified.

B023 DECIMAL SPECIFICATION NOT ALLOWED

Decimal places are not allowed in this field.

**B025 MASK DOES NOT MATCH FIELD - *{letter }*
{literal }
*{ }***

The number of digit selectors in the indicated mask does not match the number of digits in the associated field.

B026 REQUIRED PARAMETER IS NOT CODED – *word*

Additional parameters are required. That is, parameters, subparameters, or their associated values are missing. See the statement syntax description for correct information.

**B027 NOT A VALID NAME – *{file-name }*
{field-name }
*{report-name }***

The indicated name is not valid or is used out of context.

B028 VALUE DOES NOT MATCH FIELD TYPE OR SIZE – *word*

A VALUE clause specified an initial value that does not match the type or size of the associated field.

B029 PARAMETER IS INVALID – *word*

The indicated word is invalid as used in the current statement.

B030 UNBALANCED PARENTHESES

Parentheses must be balanced across a statement. (See the CA-Easytrieve Plus *Reference Guide*.)

B031 UNBALANCED APOSTROPHES

Apostrophes must be balanced across a statement. (See the CA-Easytrieve Plus *Reference Guide*.)

B032 LITERAL MUST BE SPECIFIED – word

You must specify a literal for the indicated word.

B033 INCOMPATIBLE WORK FILE FOR CHECKPOINT

The file specified on the FILE parameter must be virtual.

B034 CANNOT BE ENCLOSED IN PARENTHESES – word

The indicated word cannot be enclosed in parentheses.

B035 CANNOT BE ENCLOSED IN APOSTROPHES – word

The indicated word cannot be enclosed in apostrophes.

B036 NUMBER MUST BE POSITIVE INTEGER – word

The indicated word must be a positive integer.

B037 MUST BE ALPHA LITERAL – word

The indicated word must be an alphanumeric literal.

B038 MUST BE NUMERIC LITERAL – word

The indicated word must be a numeric literal.

B039 NAME IS AMBIGUOUS – *name*

The specified name cannot be uniquely identified. Qualify the field or record.

B040 FILE CANNOT HAVE FIELDS – *file-name*

Fields cannot be defined for files with the PRINTER attribute. Fields cannot be defined immediately following an IDMS FILE statement.

B041 IMPROPER USE OF PARENTHESES – *word*

Parentheses cannot be specified in the indicated context.

B043 STATEMENT CANNOT HAVE LABEL – *statement*

A statement label is invalid for the indicated statement.

B044 OPERAND IS MISSING

A required operand is missing for the current statement as it is coded.

B045 CHECKPOINTED USER AREA EXCEEDS MAXIMUM ALLOWED BY IMS – *length*

IMS allows 7 user areas of 32 KB each to be checkpointed. The combination of USING field areas and internal CA-Easytrieve Plus data that needs to be checkpointed exceeds this value. The supplemented message shows the length that was attempted. Decrease the number of USING field areas.

B046 CANNOT PERFORM A STATEMENT LABEL

You coded a PERFORM command that referenced a statement label. A PERFORM command must reference a PROC label. A PROC label must be immediately followed by another sentence consisting only of the word PROC.

B047 CANNOT GOTO A PROC LABEL

You coded a GOTO command that referenced a PROC label. A GOTO command must reference a statement label. A statement label cannot be followed by another sentence consisting only of the word PROC.

B048 FIELDNAME NOT IN FILE – *field-name*

The indicated field name is not contained in the specified file.

B049 PARAMETER IS IGNORED – *word*

The indicated word is not processed by the CA-Easytrieve Plus syntax check. A message is generated for each parameter skipped during syntax check. Syntax check termination occurs whenever the syntax of a statement becomes unrecognizable.

B050 DUPLICATE STATEMENT LABEL – *label-name*

The indicated label is a duplicate of a previous label. A statement label must be unique in a procedure or the main body of a job. A procedure label must be unique in a job activity or report subactivity.

B051 NO COMMON FIELDS FOUND

The MOVE LIKE command did not find any fields with the same names in the two files.

B052 MORE THAN ONE COMMON FIELD – *field*

An error was detected while processing a MOVE LIKE statement. For a field in the TO file, more than one field of the same name was found in the FROM file.

B053 FILE SHOULD BE IS OR VS – *file-name*

A READ, WRITE, or POINT statement references an unkeyed file. The file must be ISAM or VSAM.

B054 NOT A VALID FILE – *file-name*

An invalid file was specified for the current statement.

B055 INVALID LENGTH, TYPE OR DECIMAL PLACES – *word*

There is an inconsistency in the current field definition. For example: an A-type field cannot have any decimal places specified, or a two-byte packed field can have a maximum of three decimal places.

B056 UNRESOLVED LABEL REFERENCE – *label-name*

The indicated label was referenced on a statement, but a corresponding label was never specified. This is a deferred message generated at the end of each job activity. The program must be scanned to locate the invalid reference.

B057 INVALID LABEL REFERENCE

The indicated label has a reference that extends outside the allowed scope of reference for a label.

B058 UNRESOLVED REPORT REFERENCE

A PRINT statement specified a report name for which a corresponding report was never specified.

B059 PREMATURE END OF FILE

An end of file was detected (on the CA-Easytrieve Plus source program input file) before CA-Easytrieve Plus could identify a valid program. Verify that the JCL statements that associate a data set with SYSIN (SYSIPT) are valid. This message is generated when a null source program is encountered.

B060 MISSING END-PROC STATEMENT

The current procedure was not terminated properly with an END-PROC statement.

B061 REPORT LINE OVERFLOWED BY – *amount*

The current report output line overflowed the LINESIZE by the amount specified. Solutions include:

- Reduce the width of one or more line items
- Reduce the number of fields on the line
- Reduce the SPACE value
- Increase the LINESIZE value if possible

For detailed information on the previous option, see the CA-Easytrieve Plus *Reference Guide*.

B062 FIELD REFERENCED IN UNAVAILABLE FILE – *file-name*

One or more fields were referenced in the identified file, but the file is not used in the job activity. This is a deferred message that is generated at the end of each job activity. This message is always accompanied by one or more B063 messages that identify which fields were referenced in the unavailable file.

B063 FIELD REFERENCED WAS – *field-name*

This message always accompanies the B062 message and identifies which fields were referenced in the unavailable file.

B064 NUMBER OF KEYS MUST BE SAME FOR ALL FILES – *file-name*

You must specify the same number of corresponding keys for each file defined in a synchronized file group.

**B065 JOB OR SORT STATEMENT INVALID AFTER JOB - JOB
SORT**

A JOB or SORT statement immediately follows a JOB statement. There is no job activity defined for the preceding JOB statement.

B066 INVALID FILENAME, FILENAME SUBSTITUTED – *file-name*

The file name on a FILE statement was not defined properly. Substitute a file name to continue processing the program. Correct the file name definition on the FILE statement.

B067 CONFLICTING OR DUPLICATE OPTION – *option*

The specified option is not allowed or was specified more than once.

B068 PARAMETER INVALID WITH 'ADJUST' – COL

You must use the COL parameter with the REPORT NOADJUST parameter.

B069 INVALID RECORD LENGTH – *word*

The indicated record length is not valid.

B070 VALUE NOT WITHIN ACCEPTABLE RANGE – *word*

The indicated value is not in the range required for its use. See the description of the current statement for the valid range.

B071 INVALID MACRO SUBSTITUTION WORD – *word*

The indicated word is not a valid substitution word. The format of the name is incorrect, the name was not previously defined, or the name is too long.

B072 EXCESSIVE OR MISPLACED POSITIONAL PARAMETER – *word*

One of the following conditions exists:

- The number of positional parameters specified for the macro invocation exceeds the number defined in the macro definition.
- You specified a positional parameter after a keyword parameter on the macro invocation statement.
- You misspelled a keyword parameter, which CA-Easytrieve Plus interpreted as a misplaced positional parameter.

B073 NUMERIC VALUE MUST BE INTEGER – *word*

The indicated word must be an integer.

B074 IMPROPER USE OF AMPERSAND IN MACRO

You used an ampersand incorrectly in a macro definition. One of the following conditions exists:

- A substitution word was not preceded by a CA-Easytrieve Plus delimiter or the macro variable concatenation character (a period).
- You tried to define an ampersand in a literal. You must code two ampersands to define the literal ampersand. This is the same basic rule that is used for the apostrophe.

B075 UNDEFINED SUBSTITUTION WORD – *word*

The indicated word is used as a substitution word in the macro body but is not defined in the macro prototype.

B076 REMAINDER OF STATEMENT IS IGNORED

A previous syntax error occurred that makes it impossible to continue the syntax check of the current statement. The rest of the parameters on the statement are ignored.

B077 INVALID 'DO' / 'END-DO' PAIRING – *count*

An END-DO statement is missing or incorrectly placed. Every DO WHILE statement must be delimited by an END-DO statement. The value, *count*, indicates how many DO WHILE statements were not delimited by an END-DO. Look for previous error messages B173 or B185 indicating that the nesting level was in error.

B078 INVALID 'FILE EXIT USING' PARAMETER – *word*

The indicated word is not valid as a USING parameter. You cannot specify file fields for USING parameters on FILE exits.

B079 'PRESIZE' OVERFLOWED, INCREASE IT

The compiler work file's record length is too small for the current job. The value used is provided in the parameter listing at the end of the compile output. Increase that value by 512 through the PARM PRESIZE parameter and rerun the job.

B080 PARENTAGE IS INCORRECT – *record-name*

The parent of this RECORD cannot be found. The parent name on this RECORD statement is incorrect, or the RECORD statement for the parent is missing.

B081 INCORRECT QUALIFICATION – *name*

The specified name is incorrectly qualified, or the qualifier used is invalid. You used a colon instead of a period in an SQL INCLUDE.

B082 NAME IS UNDEFINED – *name*

The specified name cannot be found.

B083 CANNOT PERFORM PROC WITHIN SAME PROC

A procedure contained a PERFORM statement that references the procedure itself. This constitutes a recursive call of the procedure and is not allowed in CA-Easytrieve Plus.

B084 PARAMETER INVALID FOR IDMS – *word*

This parameter is invalid on an IDMS statement.

B085 NAME TOO LONG

The length of a name exceeded:

- 40 characters for field names
- 8 characters for OS file names
- 16 characters for CA-IDMS record names
- 8 characters for CA-IDMS PROGRAM, DB, NODE, SCHEMA, or SUBSCHEMA names

B086 INVALID LENGTH – *name*

The length of this field is not valid.

B087 NULL LITERAL INVALID

A literal must have at least one character.

B088 LITERAL NOT FOLLOWED BY DELIMITER

The specified literal was not terminated properly.

B089 INVALID MACRO NAME – *name*

The specified name on an MSTART statement for an instream macro is invalid. The name must be no more than eight characters long, and the first character must be alphabetic.

B090 NAME IS RESERVED – *name*

The referenced name is a reserved keyword. Your use of the keyword is invalid.

B091 INVALID TABLE FILE

The table is not in ascending sequence or there is a duplicate ARG.

B092 WORKAREA NOT VALID FOR THIS FILE

Workarea is not supported for this file type.

B093 FILE ORGANIZATION REQUIRES DISK DEVICE

A disk is required for the type of file.

B094 INVALID RECORD FORMAT

The record format is not valid, or it is inconsistent with the block size specified.

B096 LITERAL TOO LONG

Literals are limited to 254 characters.

B097 LENGTH INVALID FOR TYPE

The length attribute is invalid for the type of field requested.

B098 NOT A VALID TYPE

The type attribute is not valid.

B099 DECIMAL SPECIFICATION TOO LARGE

The number of decimal places cannot exceed the length of the field.

B100 LOCATION INVALID

You tried to qualify a field used in a definition with an invalid file. Work qualified fields cannot have an offset. You cannot define file qualified fields in working storage.

B101 IMPROPER FIELD OVERLAY

The overlay field cannot be longer than, or extend past, the overlaid field.

B102 TOO MANY UNIDENTIFIED MASKS DEFINED

You can only define 192 unidentified masks in any one CA-Easytrieve Plus run.

B103 OVERLAY CONFLICTS WITH QUALIFICATION

You tried to overlay a qualified field onto a field from another file.

B104 INDEX FIELD REQUIRED – *name*

The specified name in the INDEX parameter of a DEFINE statement was previously defined as a working storage field. You can only use names not previously defined or previously defined as index names in the INDEX parameter.

B105 VALUE NOT ALLOWED FOR THIS FIELD

The VALUE keyword is not allowed for this field.

B106 NAME INVALID FOR TABLE FILE

The only valid names for a TABLE file are ARG and DESC.

B107 NUMBER MUST BE NON-NEGATIVE INTEGER

You must use a zero or positive integer in this field.

B108 FILE ORGANIZATIONS INCOMPATIBLE

You cannot copy from a flat file to a database file, or from a record to a database file.

B109 DEVICE NOT ALLOWED FOR VIRTUAL FILE

You cannot specify a device for a virtual file.

B110 RECORD FORMAT REQUIRED FOR VIRTUAL FILE

The record format is required for a virtual file.

B111 SPECIFIED DEVICE NOT VALID FOR TABLE FILE

The specified device does not support table files.

B112 FILE ORGANIZATION NOT VALID FOR INSTREAM TABLES

You cannot specify a file organization keyword for an instream table file.

B113 DECIMAL PLACES NOT ALLOWED

An integer value is required in this field.

B114 NUMERIC FIELD REQUIRED

You tried to use an alphanumeric literal or a field name where a numeric field is required.

B115 FIELD IS READ/ONLY

You tried to update a read-only field.

B116 INVALID QUALIFIER

The name you tried to define is syntactically incorrect, you tried to qualify a field name with another field name, you had too many qualifiers, or you attempted to qualify a field with a reserved word other than WORK.

B117 INVALID LENGTH FOR TABLE FIELD

The length for the ARG and DESC fields cannot exceed 254.

B118 NAME MISSING

You referenced a null value in parentheses.

B119 INVALID SUBSCRIPT QUALIFIER

The specified subscript was not valid, or you used a literal where literals are not allowed.

B120 SUBSCRIPT INVALID ON A SUBSCRIPTED FIELD

You cannot subscript a field with a subscripted field.

B121 NAME EXPECTED AFTER A QUALIFIER

Qualification was started and not completed.

B122 NAME MUST BE A FILE OR FIELD – *name*

The name entered is not a field or a file.

B123 NAME NOT DEFINED IN FILE OR RECORD

The specified name cannot be found in a file or a record.

B124 NAME CANNOT BE QUALIFIED

You tried to qualify a name that cannot be qualified. Verify that you are referencing the correct CA-Easytrieve Plus variable.

B125 SUBSCRIPTS NOT ALLOWED

You incorrectly specified a subscript where subscripted fields are not allowed.

B126 TOO MANY SUBSCRIPTS

Only three subscripts are allowed.

B127 NOT A REPORT PROCEDURE NAME

While in a REPORT section, you tried to perform a procedure that is not a valid REPORT procedure. See the CA-Easytrieve Plus *Reference Guide* for a list of valid report procedures.

B128 INCORRECT NUMBER OF SUBSCRIPTS

You tried to reference a subscripted field. The number of subscripts depends on how you defined the field in the library section. See the *CA-Easytrieve Plus Reference Guide* for more details.

B129 INDEXED FIELD NOT ALLOWED

While using a variable as a subscript identifier, you referenced an indexed field. You cannot reference indexed fields on certain statements, such as USING parameter of SORT statement.

B130 SORT FIELD CANNOT EXCEED 255 BYTES

The maximum length of a sort field is 255 bytes. Use more than one adjacent field when it is necessary to sort on long fields.

B131 EXIT NOT ALLOWED FOR INSTREAM TABLES

A file exit is not allowed on an instream table file.

B132 APOSTROPHE NOT PRECEDED BY SPACE

You must precede apostrophes by a space or a left parenthesis.

B133 COMMA NOT FOLLOWED BY SPACE

A space must follow a comma.

B134 ALL OTHER PARAMETERS IGNORED FOR DATABASE

Only the DLI or IDMS keywords (and their associated sub-keywords) are valid for database file definitions.

B135 REPORT FILE LENGTH EXCEEDED

The length of a report work file record exceeded the record length specified on the FILE statement, or, if VFM is used to spool the report, the report work file record length exceeded 65535.

B136 SUM FILE LENGTH EXCEEDED

The length of the sum file record exceeded the record length specified on the FILE statement for the SUMFILE, or, if VFM is used for the sum file, the sum file record length exceeded 65535.

B137 LITERAL CONTAINS DBCS DATA FROM MULTIPLE DBCS CODE SYSTEMS—word

The identified word contains the shift codes of more than one DBCS code system. You cannot mix DBCS code systems in the one literal.

B138 DBCS HEX LITERAL MUST DEFINE AN EVEN NUMBER OF BYTES

A DBCS hexadecimal literal must contain an even number of bytes. This means that you must define it using a multiple of four hexadecimal characters.

B139 RELATIONAL OPERATOR MUST BE EQ OR NE

Series tests and range tests must use the relational operator EQ or NE.

B140 INVALID OR UNSUPPORTED DBCS CODE SYSTEM—word

The specified DBCS code system is not defined for your site.

B141 EXTENDED REPORTING PRINTER NAME NOT DEFINED—word

The printer name defined for the EXTENDED keyword is not defined in the extended reporting options module.

B142 INVALID FILE DEFINITION FOR EXTENDED REPORTING PRINTER

An extended reporting printer file must be a sequential file. It cannot have a device type of CARD or PUNCH. It cannot be a TABLE file.

B143 UNSUPPORTED DBCS CODE SYSTEM FOR EXTENDED REPORTING PRINTER

The DBCS code system assigned to an extended reporting printer was not defined in the DBCS options module.

B144 INVALID OR UNSUPPORTED USE OF DBCS DATA—word

The identified field or literal containing Double Byte data is invalid or unsupported in the context where it is being used.

B145 INVALID OR UNSUPPORTED CONVERSION LITERAL IDENTIFIER—word

The source conversion identifier was not defined in the DBCS options module, and it cannot be converted.

B146 INVALID LITERAL LENGTH RETURNED BY CONVERSION LITERAL

The length of a source conversion literal the conversion routine returned is invalid. It must be between 1 and 254 and have an even length if the literal is all DBCS characters.

B147 INVALID OR UNSUPPORTED DBCS CODE SYSTEM RETURNED BY SOURCE CONVERSION ROUTINE

The DBCS code system a Source Conversion Routine returned is not defined in the DBCS Options module.

B148 CONVERSION ERROR – message

A Source Conversion Routine error occurred. The conversion routine returned this message text to CA-Easytrieve Plus.

B149 MIXED FIELDS NOT VALID FOR DBCS CODE SYSTEM

You can only define MIXED fields for a DBCS code system that supports Wrapping or Header shift codes.

B150 SORT NOT DEFINED FOR DBCS CODE SYSTEM

The DBCS sort options are only supported for IBM and JEF DBCS code systems and only if a KANJI sort was defined in the DBCS options module.

B151 INVALID FONT INDEX VALUE

A font index value must have a value of 1 through 256.

B152 DBCS CODE SYSTEM OF SORT-IN AND SORT-OUT NOT EQUAL

For KANJI sort, the DBCS code system of the sort input file must match the DBCS code system of the sort output file.

B153 ITEM DOES NOT MATCH DATA TYPE OF FONT – *word*

The data format of the print item does not match the data format of the font to use to print the item.

B154 MAXIMUM OVERPRINT RECORDS EXCEEDED FOR PRINT LINE

A print line the indicated statement generated requires more overprint records than the assigned extended reporting printer supports.

B155 PRINT ITEM OVERLAPS EXISTING PRINT ITEM – *item*

For extended reporting printers, print items (fields and literals) cannot overlap or print on top of one another.

B156 FONT DEFINED FOR HEADING LITERAL NOT FOUND – *word*

The font number defined for one of the headings of the item identified as word is not defined in the extended reporting options module for the assigned extended reporting printer.

B157 FONT NOT DEFINED – *font-number*

The font number was not found in the extended reporting options module for the assigned extended reporting printer.

B158 INVALID USE OF DBCS DATA IN HEADING – *word*

One of the heading lines of the item identified as word contains DBCS data, and the assigned printer does not support DBCS data.

B159 POSITION FOR PAGE COUNT OCCUPIED BY PRINT ITEM

When positioning the page count on title line 1, CA-Easytrieve Plus detected that a print item already occupies that position.

B160 OPTION ONLY SUPPORTED FOR EXTENDED REPORTING PRINTER – *word*

You can only use a font index value when the report or print line is directed to an extended reporting printer that is not a line printer.

B161 LINESIZE TOO SMALL TO POSITION TAG LITERAL

The value of LINESIZE is too small to position the TAG literal on the first summary line of the report. Increase the LINESIZE value or decrease SPACE or the size of the TAG literal.

B162 EXCESSIVE PARAMETERS SPECIFIED ON STATEMENT, REMAINDER IGNORED

The statement requires a fixed number of parameters. This value was exceeded for the indicated statement.

B163 MAXIMUM RECORD LENGTH EXCEEDED FOR EXTENDED REPORTING PRINTER

A print record CA-Easytrieve Plus generated for the indicated statement exceeds the maximum record size defined for the assigned extended reporting printer.

B164 MAXIMUM DATA LENGTH EXCEEDED FOR EXTENDED REPORTING PRINTER

A print record CA-Easytrieve Plus generated for the indicated statement contains more print data than the assigned extended reporting printer supports.

B165 UNABLE TO POSITION LINE COMPLEX ITEM AT EXACT PRINT POSITION – *item-number*

The elements on each line of a Line Complex must be positioned at exactly the same print position. Due to the positioning of other print elements on the same print record, one of the elements of a line complex could not be positioned at the same print position. You must vary the fonts and character widths of elements on the print lines to enable CA-Easytrieve Plus to position these elements correctly.

B166 ITEM EXCEEDS MAXIMUM BYTE COUNT SUPPORTED BY A FONT – *item-number*

The number of bytes in the identified print item exceeds the data byte count of the assigned font. Reduce the item's width.

B167 MAXIMUM PAGE EXCEEDED BY PAGESIZE OR LINESIZE VALUE – *word*

The indicated word is too large for the maximum form size defined for the associated extended reporting printer.

B168 SPACE REPLACE CHARACTER NEEDED TO USE ASSOCIATED FONT WITH PRINT ITEM – *item-number*

The indicated print item occurring on the specified CA-Easytrieve Plus statement is positioned on an OVERPRINT record that is not the first OVERPRINT record, and the print item **must** have a space that prints at the associated font's width. The requirement for a space that prints at the associated font's width is due to the item containing MIXED format data, or the item is a field on a detail or summary line that must print as spaces (due to DTLCTL and SUMCTL options).

B170 EXPRESSION NOT ALLOWED

Expressions are not allowed in series or range tests.

B171 SUBJECT MUST BE A FIELD OR FILE

The subject of a comparison must be a field or a file. Literals are not allowed.

B172 SYNCHRONIZED FILE PROCESSING NOT ACTIVE

You coded a MATCH or DUP statement but did not code a JOB statement specifying synchronized file processing.

B173 NO MATCHING 'IF' STATEMENT

An END-IF statement was encountered without a valid matching IF statement or at the same nesting level as a DO WHILE statement.

B174 MORE THAN ONE 'ELSE' STATEMENT

You coded more than one ELSE statement in a single IF/END-IF construct.

B175 CONDITION IS INCOMPLETE

The IF statement does not contain enough information to construct a valid comparison.

B176 OBJECT OF CONDITION IS MISSING

You did not specify the object of a comparison or a condition.

B177 INVALID FIELD CONDITION

The subject is incompatible with the condition test, or the condition test is undefined.

B178 SECOND OBJECT OF RANGE IS MISSING

A range test was detected, but the second object is missing.

B179 FIRST OBJECT OF RANGE IS MISSING

A range test was detected, but the first object is missing.

B180 ARITHMETIC OPERATOR IS MISSING

Two operands were encountered that were not separated by an arithmetic operator.

B181 INVALID ARITHMETIC OPERATOR

The arithmetic operator specified is not valid.

B182 HEXADECIMAL LITERAL REQUIRED

When the object of a BIT test is a literal, the literal must be a hexadecimal literal.

B183 'AND' OR 'OR' IS MISSING

You must join compound comparisons with an 'AND' or an 'OR'.

B184 SUBJECT OF CONDITION IS MISSING

The subject of a comparison is missing.

B185 NO MATCHING 'DO' STATEMENT

An END-DO statement was encountered without a valid matching DO WHILE statement or at the same nesting level as an IF statement.

B186 FILE NAME REQUIRED

The name specified is not a file name.

B187 RECORD NAME REQUIRED

The name specified is not a record name.

B188 FIELD NAME REQUIRED

The name specified is not a field name.

B189 PROGRAM NAME REQUIRED

The name specified is not a program name.

B190 MASK NAME REQUIRED

The name specified is not a mask name.

B191 STATEMENT OR PROCEDURE NAME REQUIRED

The name specified is not a statement or a procedure name.

B192 CURSOR NAME REQUIRED

The name specified is not a previously defined CURSOR name.

B196 FILE OR RECORD NAME REQUIRED

The name specified is not a file or record name.

B197 FILE, RECORD OR FIELD NAME REQUIRED

The name specified is not a file, record, or field name.

B198 RECORD OR FIELD NAME REQUIRED

The name specified is not a record or field name.

B199 FILE OR FIELD NAME REQUIRED

The name specified is not a file or field name.

B200 FIELD OR TABLE NAME REQUIRED

The name specified is not a field or table name.

B201 LOGICAL RECORD NOT ALLOWED

You specified a logical record name where it is not allowed.

B202 NAME MUST BE A LOGICAL RECORD

You failed to specify a logical record name. You cannot specify a field name or literal.

B203 FIND STATEMENT NOT VALID WITH LOGICAL RECORD

You cannot specify a logical record name on an IDMS FIND statement.

B204 SQL – *supplemental*

An SQL error occurred. The supplemental message gives detailed information on the reason for the error as returned from the SQL interface.

B205 QUALIFYING RECORD NAME NOT VALID

The record name specified as a qualifier of a field in the WHERE parameter is not valid.

B206 OPERAND FOLLOWING PREFIX + OR - MISSING

Following a leading + (plus) or - (minus) sign, you must code a field name, numeric literal, or an expression enclosed in parentheses. The end of the expression or another operator was found instead.

B207 CLOSING RIGHT PARENTHESIS MISSING

You omitted the right parenthesis that closes the nested expression.

B208 NUMERIC HEXADECIMAL LITERAL MUST HAVE SAME LENGTH AS SUBJECT – *literal*

When comparing a numeric field to a hexadecimal literal, the length of the literal must be the same as the length of the numeric field.

B209 OPERAND FOLLOWING + OR - MISSING

Following a + (addition) or - (subtraction) operator, you must code a field name, numeric literal, or an expression enclosed in parentheses. The end of the expression or another operator was found instead.

B210 OPERAND FOLLOWING * OR / MISSING

Following an * (multiplication) or / (division) operator, you must code a field name, numeric literal, or an expression enclosed in parentheses. The end of the expression or another operator was found instead.

B211 ELEMENT DOES NOT EXIST IN DICTIONARY

The specified name is not defined in the IDD.

B212 ELEMENT DOES NOT EXIST IN SUBSCHEMA

The specified name is not defined as part of the subschema that contains the logical record being accessed.

B213 ELEMENT NOT UNIQUE IN SUBSCHEMA

The subschema definition contains more than one definition of the specified name.

B214 FIELD USED IN ARITHMETIC EXPRESSION IS NOT NUMERIC

The indicated field was used as an operand in an arithmetic expression. The data type of the operand is alphanumeric. Operands used in arithmetic expressions must be numeric.

B215 OPERAND OF A LOGICAL CONNECTIVE IS INVALID

The operands of a logical connective (AND/OR/NOT) must be a comparison, a DBA-defined name, or a Boolean expression enclosed in parentheses.

B216 FIELD USED IN MATCHES/CONTAINS TEST IS NOT ALPHANUMERIC

The indicated field was used as an operand in a MATCHES/CONTAINS test. The data type of the operand is numeric. Operands used in MATCHES/CONTAINS tests must be alphanumeric.

B217 DIVISOR EXCEEDS MAXIMUM SIZE OF 8 BYTES

The second operand of a division operator (/) in the WHERE parameter cannot exceed 8 bytes in length.

B218 INVALID NESTED CONDITION

A nested condition in the WHERE parameter is invalid.

B219 INVALID EXPRESSION IN PARENTHESES

An expression enclosed in parentheses is invalid.

B220 LOGICAL RECORD CANNOT HAVE FIELDS – *record-name*

A LOGICAL-RECORD cannot have fields associated with it. All fields that are defined as part of a LOGICAL-RECORD must follow an ELEMENT-RECORD statement that follows the LOGICAL-RECORD statement.

B221 ELEMENT RECORD MUST FOLLOW LOGICAL RECORD – *record-name*

An ELEMENT-RECORD statement was coded following a FILE statement without an intervening LOGICAL-RECORD statement. ELEMENT-RECORD statements must follow a LOGICAL-RECORD statement or another ELEMENT-RECORD statement.

B222 DATA BASE FILE NOT SPECIFIED AS INPUT

You must specify the file that contains the logical record specified by the SELECT statement on the INPUT parameter of the JOB statement.

B223 ELEMENT RECORD NOT PART OF LOGICAL RECORD – *record-name*

The name specified on the ELEMENT-RECORD statement does not match any of the record names specified for the logical record in the data dictionary.

B224 ELEMENT RECORD ALREADY DEFINED – *record-name*

The specified element record name was already defined for this logical record.

B225 SUBSCHEMA NAME NOT FOUND – *subschema-name*

While trying to locate the entry for a logical record in the data dictionary, CA-Easytrieve Plus could not locate an entry for the subschema name specified on the FILE statement.

B226 RECORD NAME NOT FOUND – *record-name*

The record name specified on a LOGICAL-RECORD statement is not defined as part of the subschema specified on the FILE statement.

B227 RECORD NOT VALID FOR THIS FILE ORGANIZATION

A RECORD statement can only follow a FILE statement that specifies an IDMS or DLI database. A LOGICAL-RECORD or an ELEMENT-RECORD statement can only follow a FILE statement that specifies an IDMS database.

B228 WHERE PARAMETER SYNTAX ERROR -- REMAINDER OF WHERE PARAMETER IGNORED

A syntax error prevented the WHERE parameter from fully processing. The unprocessed portion was ignored. This message follows the message for the original syntax error.

B229 VARYING ALLOWED ONLY ON ALPHA, KANJI, OR MIXED FIELDS

You tried to define a varying field with a data format other than A, K, or M. You cannot define a varying field as numeric. See the *CA-Easytrieve Plus Reference Guide*.

B230 INVALID INDICATOR ARRAY

The indicator array specified as an SQL host variable is not valid. Indicator arrays must be two-byte binary fields occurring more than one time.

B231 RESET INVALID WITH REDEFINE

You can only specify the RESET keyword on simple definitions. You cannot redefine RESET fields, and they cannot be part of a redefinition.

B232 RESET ONLY VALID FOR 'W' WORKING STORAGE

The RESET keyword is only valid on W-type working storage fields.

B233 INVALID USE OF VARCHAR FIELD

A VARYING field is not valid in this context.

B254 LOCATION REQUIRES A W, S, * OR INTEGER

The LOCATION clause specified where the first field must be defined. It must be W for a W-type working storage field, S for an S-type working storage field, * for the next available position in the current FILE, or an integer specifying the position in the current FILE of the first field to be defined.

B255 SQL INCLUDE STATEMENT CANNOT FOLLOW SQL LOGIC STATEMENT

The SQL INCLUDE statement cannot follow any other SQL statement except for another SQL INCLUDE statement, and the SQL INCLUDE statement must be coded in the library section.

B256 SQL PROCESSING BYPASSED DUE TO PREVIOUS ERROR

A severe error occurred while processing a previous SQL statement. The previous error prevents further SQL processing. Correct the error listed previously to allow SQL processing to continue.

B257 INVALID USE OF BIND PARAMETER

The BIND parameter is only valid if you set the SQLBIND option in your CA-EASYTRIEVE PLUS options table to ANY or blanks (the default).

B258 'WHEN' STATEMENT REQUIRED AFTER 'CASE' STATEMENT

You must follow a CASE statement with a WHEN statement. You can insert comments and compiler directives between a CASE and WHEN pair, but no statements are allowed between them.

B259 FIELD LENGTH MAY NOT EXCEED 254 BYTES

You specified a field with a length greater than 254 bytes. The compiler only accepts fields with a length of 254 or fewer bytes.

B260 LITERAL LENGTH MUST BE EQUAL TO FIELD LENGTH

You specified a literal whose length was not exactly equal to the length of the subject field. The compiler requires the lengths to be exactly equal.

B261 SERIES AND RANGES MUST BE UNIQUE

In a CASE structure, the series and ranges you specify for WHEN statements must be unique across all the structure's WHEN statements. A series value cannot be duplicated. A range cannot be duplicated. A series value cannot fall in a range. A range cannot fall in or overlap another range.

B262 INVALID 'CASE' / 'END-CASE' PAIRING – *count*

An END-CASE statement is missing or incorrectly placed. You must delimit every CASE statement with an END-CASE statement. The value (*count*) indicates how many CASE statements were not delimited by an END-CASE. Look for previous error messages B173, B185, or B263 indicating that the nesting level was in error.

B263 NO MATCHING 'CASE' STATEMENT

An END-CASE statement was encountered without a valid matching CASE statement or an END-CASE statement at the same nesting level as a DO WHILE or IF statement.

B297 PLANNAME CANNOT BE THE SAME AS LINKNAME

The SQL planname for the STATIC SQL application plan cannot be the same as the linked program name.

B298 INVALID OPTIONS TABLE

The version of the CA-Easytrieve Plus options table does not agree with the version of CA-Easytrieve Plus that you are running. See the CA-Easytrieve Plus *Installation Guide* for information on generating the options table.

B299 SELECTABLE UNIT IS NOT AVAILABLE

The selectable option that you requested is not available at this time. Check with your system programmer to be sure the CA-Easytrieve Plus installation was complete and that you are pointing to all necessary execution libraries.

B300 MIX OF IDD AND * EZTPIDD NOT ALLOWED

IDD statements and * EZTPIDD statements are not supported concurrently. Recode the EZTPIDD statements into IDD statements.

B301 IDD IDMS ERROR

The IDD statement's access into IDMS resulted in an unexpected return code. Resolve any other error messages and try again. If the messages persist, contact Computer Associates Technical Support for assistance.

B302 RECORD NAME IN SELECT NOT FOUND

The SELECT CLAUSE of an IDD SUBSCHEMA or IDD FILE statement specified a record name not found in the given dictionary.

B303 UNABLE TO RESOLVE IDD ENTITY

The SUBSCHEMA, FILE, or RECORD requested was found in the dictionary. However, the version was incorrect; or the given program name was not authorized for the entity, not in the given schema, or not registered as valid.

B304 IDD ENTITY WAS NOT FOUND

The SUBSCHEMA, FILE, or RECORD specified was not found in the given dictionary.

B305 NO REQUEST WAS ISSUED FOR IDD

An internal error occurred in the IDD statement processor. Contact Computer Associates Technical Support for assistance.

B306 USERID REQUIRED FOR SQL/DS

You did not specify the user ID parameter on the PARM statement. It is required for SQL/DS programs.

B307 ONLY VALID WITHIN LIBRARY DEFINITION SECTION

The statement issued is only valid in the library definition section of a CA-Easytrieve Plus program.

B308 ONLY VALID WITHIN ACTIVITY DEFINITION SECTION

The statement issued is only valid in the activity definition section of a CA-Easytrieve Plus program.

B309 KEYVALUE PARAMETER NOT CONSISTENT WITH RECORD KEY PARAMETER

Beginning with CA-Easytrieve Plus release 5.2, multiple CALC keys are supported. The syntax supporting multiple CALC keys on the RECORD statement and the keyvalue clause of the RETRIEVE statement was changed. While the pre 5.2 release syntax of these two statements is still supported, mixing the old and the new syntax is not allowed.

B310 MAXIMUM OCCURS VALUE IS 50

The maximum OCCURS value for the TO parameter of a HOSTDISK file is 50.

B311 MUST BE DEFINED IN 'S' WORKING STORAGE

Only fields that specify a location parameter of S are allowed.

B312 FIELD MUST BE ALPHABETIC

Only fields that specify a type of A are allowed.

B313 OCCURS VALUE MUST MATCH 'TO' FIELD'S OCCURS VALUE

The OCCURS value of the HOSTDISK file's validate field must match the OCCURS value of file's TO field.

Asterisk Prefixed Runtime Messages

The CA-Easytrieve Plus IDD interface generates the following list and explanation of operational diagnostic messages.

Note: These messages do not have a message ID.

***** ERROR ON {BIND }
 {READY } AT *nn*
 {FIND }
 {USE }

An internal error occurred. Contact Computer Associates Technical Support for assistance.

***** EXPECTED CONTINUATION NOT RECEIVED

A statement continuation was indicated, but end-of-file on the CA-Easytrieve Plus IDD source input or a non-IDD statement was detected.

***** IDD-NAME NOT FOUND OR UNAUTHORIZED

The SUBSCHEMA, FILE, or RECORD name was not found or is unauthorized in the IDMS dictionary. Verify the spelling and rerun your job.

***** NOT A VALID NAME

The indicated name is not valid or is used out of context.

***** NUMBER MUST BE POSITIVE INTEGER

The indicated word must be a positive integer.

***** PARAMETER IS INVALID

The indicated word is invalid as used in the current statement.

***** PARAMETER IS TOO LARGE

The value of the word is too large. See the statement description for the valid range.

******* REQUIRED PARAMETER IS NOT CODED**

Additional parameters are required. That is, parameters, subparameters, or their associated values are missing. See the statement syntax description for the correct information.

******* VERSION PARAMETER IS INVALID**

The version parameter specified is invalid. See the statement syntax description for correct information.

******* UNBALANCED PARENTHESES**

Parentheses must be balanced across a statement. (See the CA-Easytrieve Plus *Reference Guide*.)

DB2 Diagnostic Messages

The following messages are supplemental messages to the CA-Easytrieve Plus diagnostic messages B204, or A046. Some error explanations reference message guides. See the proper guides for the version of SQL you are running. These guides are:

- *IBM Database 2 Messages and Codes* (SC26-4113)
- *Oracle Error Messages and Codes Manual* (3605)

SQL WARNING, CODE IS xxxx

A positive SQL code was returned from a “PREPARE” of the statement. Look up the SQLCODE in the SQL message guides.

xxx CURSORS, MAXIMUM USEABLE CURSORS AT RUNTIME ARE yy

The user program defined more cursors than installation parameters for the interface specified. See the CA-Easytrieve Plus *Installation Guide* to increase the number of predefined interface CURSORS.

SQL ERROR, SQL CODE IS xxxx

A negative SQL code was returned from a “PREPARE” of the statement. Look up the SQLCODE in the IBM SQL message guides.

KEYWORD IS INVALID OR AN UNSUPPORTED COMMAND: xxxxxxxx

The first word of the SQL statement must be an SQL command keyword.

STATEMENT CONTAINS INVALID CHARACTER OR TOKEN: xxxxxxxx

A secondary keyword of a command is not valid. In the case of a cursor name, the cursor name contains invalid characters.

THE OBJECT OF THE DECLARE STATEMENT IS NOT CORRECT

The object of the DECLARE CURSOR must be a SELECT statement or an INSERT statement (SQL/DS only).

ANOTHER KEYWORD WAS EXPECTED

The SQL statement is incomplete; additional keywords were expected but not found.

CURSOR NAME MUST BE SPECIFIED

A valid cursor name was not found immediately after the SQL command word.

END OF STATEMENT EXPECTED

Extra characters were found beyond the valid SQL statement.

ERROR LIMIT EXCEEDED, PROCESSING OF STATEMENT SUSPENDED

Too many errors were encountered when trying to process the SQL statement. Further processing of the statement is suspended.

MESSAGE DATA: xxxxxxxx

This is a supplemental message for a previous message that displayed the SQLCODE. This message shows the information to insert in the message text when the SQLCODE is looked up in the message guide.

KEYWORD IN ERROR : xxxxxxxx

This is a supplemental message for a previous syntax error message. The invalid keyword or token is displayed.

SQL ERRORS FOUND

An ACCESS MODULE was not created for this program because SQL statement errors were encountered.

EMPTY OR INVALID ACCESS MODULE

An ACCESS MODULE was not created because the ACCESS MODULE does not contain any SQL table processing statements, such as FETCH, INSERT, DELETE.

CURSOR NAME xxxxxxxx PREVIOUSLY DEFINED

A cursor name can be declared only once.

OPEN COMMAND REQUIRED FOR CURSOR xxxxxxxx

A cursor was referenced in a FETCH, UPDATE, or DELETE without an OPEN statement being executed.

DECLARE STATEMENT REQUIRED FOR CURSOR xxxxxxxx

The cursor name was used in an SQL statement without being defined by an SQL DECLARE CURSOR statement.

DECLARE FOR INSERT REQUIRED FOR CURSOR xxxxxxxx

The cursor name was found in an SQL PUT statement without being defined by an SQL DECLARE FOR INSERT.

DECLARE FOR SELECT REQUIRED FOR CURSOR xxxxxxxx

The cursor name was found in an SQL FETCH statement without being defined by an SQL DECLARE FOR SELECT.

CURSOR xxxxxxxx NOT REFERENCED IN AN SQL FETCH COMMAND

A cursor was declared for SELECT but was not referenced in a FETCH statement, or a cursor was referenced in an UPDATE or DELETE WHERE CURRENT OF CURSOR-NAME but was not referenced in a FETCH statement.

CURSOR xxxxxxxx NOT REFERENCED IN AN SQL PUT COMMAND

A cursor was declared for INSERT but was not referenced in a PUT statement.

CONFLICTING USE OF CURSOR xxxxxxxx

A cursor name was declared for INSERT use with a PUT statement, yet the same cursor name is referenced in a FETCH, UPDATE, or DELETE CURRENT OF CURSOR-NAME.

START UP ERROR REPORTED BY INIT

The SQL interface could not initialize itself. When trying to terminate the interface, resources could not be released due to the error in the initialization.

ERROR WHILE EXECUTING AN INTERNAL SQL STATEMENT. SQL CODE IS xxxx

A severe error occurred in the SQL interface:

For DB2 the error is related to the Call Attach facility.

Look up the SQLCODE in the IBM message guides.

INSUFFICIENT STORAGE TO LOAD xxxxxxxx

Not enough storage was available to load the module.

MODULE xxxxxxxx NOT FOUND

The module was not found.

UNABLE TO LOAD MODULE xxxxxxxx

An error occurred loading a module. Look for IBM error messages that accompany this message.

UNABLE TO OBTAIN GLOBAL STORAGE

Storage was not available for the SQL interface to run.

ALL INTERNAL CURSORS HAVE BEEN USED, NO CURSORS REMAINING FOR DECLARE

During the execution of the user module, more cursors were defined and concurrently opened than were predefined by the interface during the installation. See the installation documentation for information about increasing the number of predefined cursors for the interface.

CURSOR xxxxxxxx MUST BE DECLARED BEFORE EXECUTING RELATED CURSOR STATEMENTS

You must define a cursor in an SQL DECLARE statement before it can be referenced in any subsequent SQL statements.

SQL CONNECT ERROR, SQLCODE = xxxx

The interface executed an explicit CONNECT on behalf of the programmer to preprocess the SQL statements. A non-zero SQL code was returned. Verify that the proper authorization exists for the given user ID to access the SQL/DS subsystem. Further processing of the user program is suspended.

SQL CREATE ACCESS MODULE ERROR, SQLCODE = xxxx

The interface executed an SQL CREATE PROGRAM on behalf of the programmer to create an ACCESS MODULE for the SQL statements. A non-zero SQL code was returned to the interface. Refer the SQL code to a systems programmer. Further processing of the user program is suspended.

RELEASE LEVEL MISMATCH BETWEEN DB2 AND THE CALL ATTACH FACILITY

The release level of the CALL ATTACH facility and that of the DB2 subsystem do not match. The interface cannot execute in this environment. Further processing of the user program is suspended.

DB2 SUBSYSTEM xxxx IS NOT ACTIVE

The DB2 subsystem ID that was specified by the user is not currently active. The interface cannot execute in this environment. Further processing of the user program is suspended.

DB2 SUBSYSTEM xxxx DOES NOT EXIST

The interface cannot establish a connection to the DB2 subsystem ID that the user specified. Correct the subsystem ID and rerun the job. Further processing of the user program is suspended.

PLAN NAME xxxxxxxx NOT AUTHORIZED

The user was not granted authorization to execute the plan. Further processing of the user program is suspended.

DB2 ERROR IN SUBSYSTEM zzzz; RETURN CODE xxxx, REASON CODE X'yy'

An unanticipated error was encountered when trying to use the DB2 CALL ATTACH facility. Contact your systems programmer. Further processing of the user program is suspended.

PLAN xxxxxxxx NOT FOUND

The plan name specified does not exist in the DB2 subsystem.

MAXIMUM CONNECTIONS TO DB2 SUBSYSTEM xxxx EXCEEDED

The maximum number of concurrent connections was exceeded. Retry the job later. Connection limits are specified during DB2 installation for TSO, Batch, and Call Attachment environments.

DB2 DENIED ACCESS TO SUBSYSTEM xxxx. DB2 EXECUTING IN RESTRICTED ACCESS MODE

A request to connect to a DB2 subsystem was rejected. DB2 was started in restricted access mode. Only user IDs authorized to perform maintenance functions are permitted access.

INTERNAL ERROR xxxx yyyyyyyy

An internal error occurred in the SQL interface.

xxxx	A code defining the error
yyyyyyyy	Text data associated with the error

STATIC COMMAND PROGRAM xxxxxxxx NOT FOUND

The program requested an SQL static-only execution, but the command program could not be found. Verify that the secondary steps ran successfully.

TIME STAMPS DO NOT AGREE

The PAN/SQL time stamp of the SQL static command program does not match that of the CA-Easytrieve Plus program. Verify the correct running of the secondary steps.

xxxxxxx FOUND INSTEAD OF xxxxxxxx

The SQL command program contains the wrong program. Verify the correct running of the secondary steps.

-818 RETURNED FOR STATIC COMMAND

DB2 detected a time stamp problem between the DBRM in the application plan and the command program. The GENDATA file the CA-Easytrieve Plus SQL program created must be reprocessed. Rerun the DB2BIND proc.

-911 RETURNED FOR COMMAND PROGRAM

DB2 detected an authorization problem between the user and the application plan. The user ID executing the application plan was not granted RUN authority by the user ID who created it. Execute with the same user ID that created the plan or grant access on the plan to the user ID that needs to execute it.

INSUFFICIENT STORAGE TO LOAD xxxxxxxx

There was not enough storage to load the command program. Increase your region size.

UNABLE TO EXECUTE USING STATIC SQL

The execution mode is STATIC-ONLY; however, the user program cannot be executed using static SQL at this time. The reason is reported in the next error message.

EXECUTION MODE CHANGED TO DYNAMIC EXECUTION DUE TO ERROR

The OPEN for the GENDATA DD statement failed. The execution mode is changed to DYNAMIC to allow the remainder of the program to compile. Verify the presence of the GENDATA DD statement. Look on the console log for information relating to the OPEN failure.

NUMBER OF HOST VARIABLES EXCEEDS 999

An SQL statement cannot have more than 999 host variables. If necessary, convert the SQL statement into multiple statements.

DELETE FAILED FOR STATIC COMMAND PROGRAM xxxxxxxx

PAN/SQL could not delete the named command program. Check the console log for DB2 error messages.

FETCH MUST BE EXECUTED PRIOR TO UPDATE OR DELETE WHERE CURRENT OF CURSOR

A cursor must be executed in an SQL FETCH statement before it can be referenced in any subsequent SQL statement.

THE CURSOR MUST BE DECLARED FOR UPDATE IN ORDER TO UPDATE OR DELETE WHERE CURRENT OF

A cursor must be defined in an SQL DECLARE statement before it can be referenced in any subsequent SQL statement.

MVS OPEN ERROR ON GENDATA FILE, OPEN RC=xxxxxxx

The open on the GENDATA FILE failed for the given reason. Report the error to your systems programmer.

INSUFFICIENT FILE SPACE FOR FILE=GENDATA

A PUT to the GENDATA file failed due to insufficient space. Increase the disk space for file GENDATA.

MVS CLOSE ERROR ON GENDATA FILE, OPEN RC=xxxxxxx

The close on the GENDATA FILE failed for the given reason. Report the error to your systems programmer.

CA-PanAudit Plus Message Identification

CA-PanAudit Plus provides two sets of diagnostic messages:

- CA-Easytrieve Plus diagnostics
- CA-PanAudit Plus diagnostics

CA-Easytrieve Plus diagnostics consist of the error messages that occur in a CA-Easytrieve Plus program. They are issued for errors made when using the host language. For a complete list of the CA-Easytrieve Plus diagnostic messages, see the beginning of this chapter.

CA-PanAudit Plus diagnostics have a similar format and are listed on the following pages. All CA-PanAudit Plus diagnostics begin with the identifier PAP and describe errors detected in the CA-PanAudit Plus routines.

CA-PanAudit Plus Message Format

All CA-PanAudit Plus diagnostic messages conform to a format similar to that of CA-Easytrieve Plus:

```
****PAPNNN X-----X - S-----S
  Message   Diagnostic      Message
    ID       Message        Supplement
```

Message ID

The message ID equals a three-character code that identifies each error message.

Diagnostic Message

The diagnostic message is a description of the detected error.

Message Supplement

The message supplement is optional, depending on the diagnostic message and context of the message. If possible, CA-PanAudit Plus provides a message supplement to identify the particular object that is in error.

PAP100 - PAP324 Runtime Messages

CA-PanAudit Plus provides a comprehensive set of diagnostic messages that describe errors detected in the CA-PanAudit Plus routines. The most likely reasons for a CA-PanAudit Plus diagnostic message are:

- Parameter value is incorrect or out of acceptable range.
- Input data is incorrect or out of acceptable range.

PAP100 INVALID PARAMETERS

The value of a parameter is not in the specified valid range of values. See the syntax section of the routine for information regarding the valid range of parameter values.

PAP101 DATECALC KEYWORD IS NOT PLUS OR MINUS

The third parameter in DATECALC is not PLUS or MINUS. See the syntax section of the routine for a description of this parameter.

PAP102 INVALID NUMBER OF RECORDS PARAMETER

The number of records requested for generation by FILEGEN exceeds the limit of 100,000,000. See the description of the number parameter in the FILEGEN routine.

PAP103 INVALID HEX/NOHEX PARAMETER

The HEX/NOHEX parameter is invalid. See the syntax section of the routine for the correct syntax of the HEX/NOHEX parameter.

PAP200 INVALID DATE FORMAT – *format*

An invalid date format was specified. See the syntax section of AGING for a description of valid date formats.

PAP201 INVALID BASE DATE FORMAT – *format*

An invalid date format for the BASEDATE parameter was specified. See the syntax section of AGING for a description of valid date formats.

PAP202 NUMBER OF INTERVALS GREATER THAN LIMIT – *intervals*

The value specified for the INTERVALS parameter is greater than the allowable limit of 8.

PAP203 AGE IS LESS THAN ZERO – *age*

The AGING routine calculated the age of a record to be less than zero. Any of the following can cause this error:

- The datefield parameter contains bad data. This can include bad numeric data or an invalid date that is chronologically after the basedate.
- The format of the date in the datefield parameter does not correspond to the format specified by format1.
- The date specified in the basedate parameter is incorrect. The date specified by basedate must be chronologically after all dates specified by the datefield parameter.
- The format of the date in the basedate parameter does not correspond to the format specified by baseform.

PAP204 BAD KEY PROCESSING PARAMETER

The PERFORM DOLUKEY procedure was invoked by the user to perform key processing, but the required KEY parameter was not properly coded. See the syntax section of DOLUNIT for a description of the KEY parameter.

PAP205 BAD REPORT/NOREPORT PARAMETER

The REPORT/NOREPORT parameter in DOLUNIT is invalid. See the syntax section of DOLUNIT for a description of this parameter.

PAP206 INVALID KEY/TOP PARAMETER

The KEY/TOP parameter in DOLUNIT is invalid. See the syntax section of DOLUNIT for a description of these parameters.

**PAP208 NUMBER OF PRIMARY KEYS DOES NOT EQUAL NUMBER OF SECONDARY KEYS
NUMBER OF PRIMARY KEYS – *primary*
NUMBER OF SECONDARY KEYS – *secondary***

The indicated number of primary and secondary keys are not equal. See the syntax section of the FILECOMP routine for a description of the PRIKEYS and SECKEYS parameters.

PAP209 NO RECORDS IN PRIMARY INPUT FILE

The primary input file to FILECOMP contains no records.

PAP210 NO RECORDS IN SECONDARY INPUT FILE

The secondary input file to FILECOMP contains no records.

PAP212 END OF FILE REACHED ON – *primary*

End of file was reached on the primary input file. This is an informational message indicating that the primary input file for FILECOMP reached EOF.

PAP213 END OF FILE REACHED ON – *secondary*

End of file was reached on the secondary input file. This is an informational message indicating that the secondary input file for FILECOMP reached EOF.

PAP214 ALL REMAINING RECORDS ARE UNMATCHED

End of file was reached on the primary or secondary input file for FILECOMP. Any remaining records on the other file are processed as unmatched.

PAP215 INVALID SIZE PARAMETER OF – *size*

The indicated SIZE parameter for the INTSAMP routine is less than or equal to zero. See the syntax section of INTSAMP for a description of the size parameter.

PAP216 FILE COMPARE ENDED

TOTAL OF - *count* - *primary* - RECORDS READ

TOTAL OF - *count* - *secondary* - RECORDS READ

The file comparison process is completed. This is an informational message indicating that the FILECOMP routine reached EOF on both the primary and secondary input files. The message lists the record count and file names of the primary and secondary input files.

PAP217 NO UNEQUAL RECORDS FOUND

No unmatched records were found during the file comparison process. This is an informational message indicating that FILECOMP found no unmatched records in the locations and lengths specified.

PAP218 TOTAL OF - *count* - UNEQUAL RECORDS FOUND

The indicated number of unmatched records were found during the file compare process. This is an informational message indicating the number of unmatched records FILECOMP found.

PAP220 *UNEQUAL PAIR* - *count* - IN - *description*

An unmatched pair of records was found during FILECOMP processing. The number of the unmatched pair and a description are provided.

PAP221 JOB ENDED DUE TO EXCESSIVE ERRORS

The maximum number of unequal record pairs specified in the maximum parameter in FILECOMP was exceeded. This message is followed by a count of the number of unmatched record pairs found during FILECOMP processing.

PAP222 PERCENT VALUE < 00% OR > 100% – *percent*

The indicated value for the percent parameter is less than 0 or greater than 100. See the syntax section of the routine for a description of the percent parameter.

PAP223 ASTERISKS VALUE < 1 OR > 9 – *number*

The indicated value for the asterisks parameter is less than 1 or greater than 9. See the syntax section of the routine for a description of the asterisks parameter.

PAP224 NO RECORDS SELECTED FOR SORT

The sort the routine performed resulted in zero records sorted. A probable cause for this error is that the input file to the routine contains no records.

PAP225 NO REPORT PRODUCED, CHECK DATE PARAMETERS

A JIF report was requested, but no report was produced. Probable causes are that the date parameters specified did not coincide with dates found on the SMF data set, or that a selection criterion of the report was never fulfilled, and no lines of the report were ever generated.

PAP226 STD DEVIATION IS ZERO OR LESS – *value*

The indicated standard deviation calculated by the routine is less than or equal to zero. Execution is terminated. If processing is to continue, invalid statistical results may be obtained. A probable cause for this error is that all input values are zero, or the population for which the standard deviation is calculated contains only one member, or all members have the same value.

PAP227 BAD CONFIDENCE VALUE – *confidence*

The value for the confidence parameter is invalid. See the syntax section of the routine for a description of the confidence parameter.

PAP228 TARGET IS LESS THAN OR EQUAL TO ZERO

The value for the target parameter in SPS is less than or equal to zero. Because the absolute value of the input field is used in SPS processing, a target value less than or equal to zero would force the selection of all records for the sample file.

PAP229 INVALID LIST OPTION – *parameter*

The parameter that specifies the type of listing to produce is not ALL or CHANGES. See the syntax section of SRCECOMP for a description of this parameter.

PAP230 TOO MANY RECORDS IN *oldfile* – *count*

The old file in the SRCECOMP routine has greater than 16000 records. See the operation section of SRCECOMP for a discussion of the statement limitations of SRCECOMP.

PAP231 TOO MANY RECORDS IN *newfile* – *count*

The new file in the SRCECOMP routine has greater than 16000 records. See the operation section of SRCECOMP for a discussion of the statement limitations of SRCECOMP.

PAP232 TOO MANY RECORDS IN MERGED FILE – *count*

The merged file created from the old and new files has greater than 16000 records. See the operation section of SRCECOMP for a discussion of the statement limitations of SRCECOMP.

PAP233 NUMBER OF SAMPLE SIZES AND NUMBER OF OUTFILES DO NOT MATCH

The number of sample files specified in the numsamp parameter is not equal to the number of output files specified. See the syntax section of STOPORGO for a description of the appropriate parameters.

PAP234 PREMATURE END OF FILE ON – *file*

The number of records requested for a sample file in STOPORGO is greater than the number of records in the associated input file. The value of the SSn parameter is greater than the number of records in the indicated input file.

PAP235 IMPROPER INDEPENDENT VARIABLE NUMBER USE 2 OR 3

The value specified for the type parameter is not 2 or 3. See the syntax section of MULTREG for a description of the type parameter.

PAP236 UNABLE TO PERFORM REGRESSION DUE TO IMPROPER DATA

The data supplied to solve the regression equation is insufficient to calculate the necessary coefficients. A probable cause for this error is that insufficient data points were input to MULTREG for it to solve the regression equation.

PAP237 FILE SIZE, SAMPLE SIZE, OR PERCENTAGE NOT VALID – *size*

The requested sample size is greater than the total file size or is less than zero. To generate a valid sample, the required sample size must be less than the total number of records in the file, and it must be greater than zero. See the syntax section of the routine for a description of the parameter that determines the sample size.

PAP238 STANDARD DEVIATION OF ALL STRATA IS EQUAL TO ZERO

STRATIF uses the standard deviation of the strata to determine sample size. When the standard deviation of all strata is equal to zero, it cannot perform the necessary calculation. The probable cause is that all values in each stratum are identical, which results in a stratum standard deviation value of zero.

PAP239 THE STRTBL FILE HAS NO RECORDS IN IT

The internal table STRTEVL uses has no records. A probable cause for this error is that the table is not generated in the associated execution of STRATIF. Check that the STRTEVL keyword parameter is specified in the execution of STRATIF.

PAP240 THE – *file-name* - FILE HAS NO RECORDS IN IT

The input file to STRTEVL does not contain any records. Check that the sample file STRATIF created is not empty. Also check that the file is not accidentally erased when the audited amounts for the sample file are entered.

PAP241 INCORRECT NUMBER OF DECIMAL DIGITS IN RECORDED AMOUNT

The number of decimal digits considered significant to STRTEVL is two. If the fields specified for the recorded and audited amounts contain more than two decimal digits, one of the first two digits to the right of the decimal point must contain non-zero values. If this is not the case, a table search operation fails to attribute a recorded amount to a stratum.

PAP242 DID NOT FALL WITHIN ESTABLISHED BOUNDARIES

The table search operation in STRTEVL failed. A probable cause for this error is that the STRTBL file was modified or is in error.

PAP243 PREMATURE END OF FILE ON STRTBL

The input file to STRTEVL has records beyond the limits of the strata define in the stratification process STRATIF performed. A probable cause for this error is that the STRTBL file was modified or is in error.

PAP244 UNEXPECTED END OF FILE ON – *file-name*

An internal error occurred in CBLCNVRT. Contact Computer Associates Technical Support for assistance.

PAP245 MISMATCH WITH FIELDS AND DM1

An internal error occurred in CBLCNVRT. Contact Computer Associates Technical Support for assistance.

PAP246 OCCURS DEPENDING ON NOT FULLY SUPPORTED PLEASE SUPPLY OCCURS DEPENDING ON FIELDS WITH INDEXES

CBLCNVRT encountered a field using the OCCURS DEPENDING ON clause. You must supply these fields with CA-Easytrieve Plus indexes.

PAP247 NUMBER OF ALLOWABLE DATA DEFINITIONS EXCEEDED

The CBLCNVRT NUMDEF parameter was exceeded. For a description of the CBLCNVRT parameter, see the [Macro Reference Guide](#).

PAP248 INVALID DAYS PARAMETER – *days*

The DAYS parameter is invalid. See the syntax section of AGING for a description of this parameter.

PAP252 TOLERROR IS LESS THAN OR EQUAL TO ZERO

The TOLERROR parameter is less than or equal to zero. See the syntax section of the routine for a description of this parameter.

PAP253 EXPERROR IS LESS THAN OR EQUAL TO ZERO

The EXPERROR parameter is less than or equal to zero. See the syntax section of the routine for a description of this parameter.

PAP254 CALCULATED TARGET VALUE IS LESS THAN OR EQUAL TO ZERO

The target value the routine calculated is less than or equal to zero. Execution is terminated. A probable cause for this error is that the expected error is too large in relation to the tolerable error. This causes the target value to be less than zero and can result in the entire file being chosen for the sample file.

PAP299 PARAMETER – *parameter* – DOES NOT CORRESPOND TO DEFINED DATA FORMAT

The indicated parameter is not defined with a valid numeric data format. See the syntax section of the routine for a description of the parameter in question.

PAP300 MISSING REQUIRED BARS COMMAND

The BAR parameter was specified in the GRAPH routine, and the associated BARS command is missing from the Keyword File. See the syntax of the BAR graph for a description of the BARS parameter.

PAP301 MISSING REQUIRED FLDN COMMAND

A FLDn command is missing from the Keyword File. For each non-zero f(n) field specified in the invocation of the BAR graph, a FLDn command in the Keyword File must specify the method used to compute the output values. See the syntax of the BAR graph for a description of the FLDn parameter.

PAP302 MISSING REQUIRED VARNUM COMMAND

The VARNUM command is missing from the Keyword File for a PLOT graph. VARNUM is a required command that specifies the number of x variables used in the PLOT graph. See the syntax of the PLOT graph for a description of the VARNUM parameter.

PAP303 INVALID COMMAND LINE – NOTHING ON CARD

A command line in the Keyword File is blank. Enter a valid command or delete the blank line.

PAP304 INVALID COMMAND LINE – SYNTAX INCORRECT

The syntax of a command line in the Keyword File is incorrect. Examine the statements in the Keyword File for correct syntax in accordance with the appropriate graph type.

PAP305 INVALID ACTION CODE

An internal error occurred in the GRAPH routine. Contact Computer Associates Technical Support for assistance.

PAP306 INVALID GRAPH TYPE – *type*

The graph type specified is invalid. Valid graph types are BAR, DEV, HIST, and PLOT. See the syntax of the GRAPH routine for a description of the type parameter.

PAP307 INVALID COMMAND SYNTAX OR ACTION IN DEV ROUTINE REFER TO GRAPH LISTING FOR ERROR DESCRIPTION

The syntax of a command line in the Keyword File is incorrect in the DEV type graph. The graph listing contains a detailed description of the error condition.

PAP308 DIVISION BY ZERO – *value or field-name*

Division by zero cannot be done in the macro. The value of the divisor is zero, or the field listed contains a zero.

PAP309 SEQUENCE NUMBERS NOT IN CORRECT ORDER

The numbers are not in ascending order. Try using the U (unsorted) option instead of S (sorted) in GAPCHK2.

PAP310 FIELD CANNOT BE ZERO – *value or field-name*

The field cannot be zero. If it is zero, a division by zero in the macro occurs.

PAP311 AS A RESULT OF THE PARAMETERS SPECIFIED, A NUMBER OF STRATA IN EXCESS OF *value* HAS OCCURRED

As a result of the parameters specified on the STRATIF1 routine, the total number of strata to create exceeded the value of MAXSTRATA, and the job terminated. See the syntax description of STRATIF for a description of the MAXSTRATA parameter.

PAP312 MAXSTRATA IS LESS THAN OR EQUAL TO ZERO

The value specified for the MAXSTRATA parameter on the STRATIF1 macro is less than or equal to zero. See the syntax description of the MAXSTRATA parameter.

PAP313 RANGE IS LESS THAN OR EQUAL TO ZERO

The value specified for the RANGE parameter on the AGING1 macro call is less than or equal to zero and invalid. See the syntax description of AGING for a description of the RANGE parameter.

PAP314 RANGE *num* IS LESS THAN OR EQUAL TO ZERO

The value specified for the RANGEx parameter on the AGING1 macro call is less than or equal to zero and invalid. See the syntax description of AGING for a description of the RANGE parameter.

PAP315 VALUE LESS THAN ZERO – *value or field-name*

The value or field on the SQRT invocation was less than zero and invalid. See the syntax description of SQRT for a description.

PAP316 INVALID SIZE PARAMETER – *size*

The syntax of the size parameter is incorrect for this routine. See the parameter specifications for the routine and correct.

PAP318 INTERTAB MUST BE SPECIFIED

INTERTAB was not coded correctly on the INTERVL1 invocation. See the syntax description of INTERVL for a description of the INTERTAB parameter.

PAP319 LOWERLIM IS GREATER THAN ZERO – *value or field-name*

The value specified for LOWERLIM on the INTERVL1 invocation is greater than zero. See the syntax description of INTERVL for a description of the LOWERLIM parameter.

PAP320 STRATTAB MUST BE SPECIFIED

STRATTAB was not coded correctly on the STRATIF1 invocation. See the syntax description of STRATIF for a description of the STRATTAB parameter.

PAP321 INTERTAB END POINTS NOT IN ASCENDING ORDER

The end points specified for INTERTAB must be in ascending order. See the syntax description of INTERVL for a description of the INTERTAB invocation and parameters.

PAP322 STRATTAB END POINTS NOT IN ASCENDING ORDER

The end points specified for STRATTAB must be in ascending order. See the syntax description of STRATIF for a description of the STRATTAB invocation and parameters.

PAP323 INVALID VALUE PARAMETER – *parameter*

The VALUE parameter on the SPS1 invocation is invalid. See the syntax description of SPS for a description of the VALUE parameters.

PAP324 MUST BE VALID OR INVALID – *parameter*

The VALID/INVALID parameter in the FLDVALx2 invocation is incorrect. See the syntax description of FLDVALx for a description of the VALID/INVALID parameters.

Debugging Techniques

CA-Easytrieve Plus system facilities assist you in writing and debugging programs. These facilities include:

- The format of the compile listings
- Information printed at execution time
- Abnormal termination facilities

This chapter discusses the format of compile listing options in detail, including:

- Page header
- Statement listing
- Parameter listings
- CLIST statement offset map
- DMAP data map
- PMAP program map
- Cross-reference report

This chapter also discusses system-defined fields and lists symbol references and reserved words.

Utility Programs

The following utility programs are distributed with CA-Easytrieve Plus.

EZTPX01

EZTPX01 is a called subprogram used to interrogate a PARM coded on a JCL EXEC statement.

EZTPX01 requires two parameters, the system-defined PARM-REGISTER and a user-defined input/output field. The input/output field must consist of a 2-byte binary field immediately followed by a character portion to contain the actual PARM information.

You are responsible for placing the maximum length you expect for the PARM information (the character portion) into the 2-byte binary field before calling EZTPX01. EZTPX01 moves your PARM data from your JCL EXEC statement into the character portion of your input/output field and updates the length in the 2-byte binary portion.

If the PARM data is larger than the maximum length you specified, it is truncated on the right, and the maximum length you defined is retained. If the PARM data is shorter than the length you specified, the length is updated to reflect the actual length of the PARM data. See the chapter “Processing JCL Parameters” in the CA-Easytrieve Plus *Application Guide* for a sample program using EZTPX01.

EZTPX03

EZTPX03 builds the double byte character set options module. See the chapter “DBCS Option Installation” in the CA-Easytrieve Plus *Installation Guide* for more information.

EZTPX04

EZTPX04 builds the extended reporting options module. See the chapter “Options Module” in the CA-Easytrieve Plus *Extended Reporting Facility Guide* for more information.

EZTPX05

EZTPX05 is a called subprogram used to recreate the original control statements that generate your current CA-Easytrieve Plus options table. EZTPX05 requires one parameter, an 80-byte alphanumeric array occurring once for each possible CA-Easytrieve Plus option. Currently, there are approximately 80 options available. See the CA-Easytrieve Plus *Installation Guide* for a sample program using EZTPX05.

Compile Listing

CA-Easytrieve Plus compiles your source program into executable machine language. The compiler also produces several listings that inform you about the results of the compilation. Optional PARM statement parameters select the types of printed output generated by the compilation. A compiled program uses the options that were in effect at the time of compilation.

The compile listing is always directed to the CA-Easytrieve Plus system output file (SYSPRINT). If the Extended Printer Options module (EZTPXRPT) has associated this file with an extended reporting printer, then the logical print records satisfy the requirements of that printer.

Header

The header consists of two lines. The first line contains the following:

- The date and the time of the compile.

If an extended-reporting printer is defined for the system output file (SYSPRINT) and that printer supports DBCS format data, then the date specified in the DBCS Option module (DATE parameter) is printed. If that date is not specified, or the system output file does not support DBCS data, the date is in the format defined in the CA-Easytrieve Plus system option module.

The date is in the format specified by the DATE option during installation. The time is in *hh.mm.ss* format.

- Compiler identification.
- The page number.

As with the date, if the system output file is associated with an extended-reporting printer that supports DBCS type data, then CA-Easytrieve Plus edits the page count and converts it into the corresponding DBCS value. This number is then combined with the Kanji page character. If the printer does not support DBCS format data, then CA-Easytrieve Plus formats the page number using the value coded in the CA-Easytrieve Plus system option module (the PAGEWRD keyword).

The second line contains the installation name as defined by the option COMPNAME. The header is repeated at the top of each page of compiler-generated output.

If the printer associated with the system output files (SYSPRINT) supports DBCS data, and the COMPNAME option in the DBCS Option module is specified, then the DBCS installation name is output on this line. If the printer does not support DBCS data, then CA-Easytrieve Plus uses the company name defined in the CA-Easytrieve Plus Options module. See the COMPNAME option in the CA-Easytrieve Plus *Installation Guide*.

Statement Listing

Input to the CA-Easytrieve Plus compiler is listed one record per line. The line consists of a CA-Easytrieve Plus-generated statement number (A), followed by the input record (B). If the input is from a macro, '-macroname' (C) is appended to the line.

(A)	(B)	(C)
1	FILE TESTIN	
2	%DATADEF	
3	REGION 1 1 N	-DATADEF
4	BRANCH 1 1 N	-DATADEF
5	EMP# 9 5 N. CODE 16 1 N	-DATADEF
7	SEX 127 1 A	-DATADEF
8	JOB INPUT(TESTIN)	
9	DISPLAY NEWPAGE REGION BRANCH EMP#	
10	STOP	

If the compilation processes a DBCS literal (as identified by the presence of shift codes), and the CA-Easytrieve Plus system output file is directed to a printer that supports DBCS data, then CA-Easytrieve Plus prints the DBCS data as part of the normal compile listing.

Listing Control Statements

Listing control statements allow you to control (format) the physical layout of the statement listing:

- You can place listing control statements anywhere in the CA-Easytrieve Plus source.
- Listing control statements must be on a record by themselves.
- Listing control statements do not appear in the printed output.

The five listing control statements of CA-Easytrieve Plus are:

- LIST
- NEWPAGE
- SKIP
- PUSH
- POP

LIST

LIST regulates the printing or suppression of all statements. Its syntax is:

LIST [ON|OFF] [MACROS|NOMACROS]

[ON|OFF] ON specifies that all subsequent statements are printed. OFF suppresses the printing of all subsequent statements.

[MACROS|NOMACROS] MACROS specifies that macro statements are printed if a LIST ON is in effect. NOMACROS suppresses the printing of macro statements.

The default is LIST ON MACROS.

To suppress all CA-Easytrieve Plus listing information, use LIST OFF

PARM LIST(NOPARM NOFILE)

See the CA-Easytrieve Plus *Reference Guide* for more information about the PARM statement.

NEWPAGE

NEWPAGE ejects the printer to the top of the next page before printing the next line of the source program. Its syntax is:

NEWPAGE

SKIP

SKIP spaces the printer the designated integer number of lines before printing the next line. The integer must be positive. Its syntax is:

SKIP integer

PUSH

PUSH saves the current status of the listing control indicators. Its syntax is:

PUSH

POP

POP restores the previous status of the listing control indicators. Its syntax is:

POP

PUSH and POP are especially useful in macros to control the listing of the macro expansion without affecting listing control outside the macro.

Example

The following example illustrates some of the listing control statements:

Input program:

```
FILE TESTIN
SKIP 1
LIST NOMACROS
%DATADEF
SKIP 1
JOB INPUT(TESTIN)
  DISPLAY NEWPAGE NAME
STOP
```


Produces this compile listing:

```
1 FILE TESTIN
2 %DATADEF
8 JOB INPUT(TESTIN)
9 DISPLAY NEWPAGE NAME
10 STOP
```

Diagnostic Format

When CA-Easytrieve Plus detects syntax errors in the program source, an error message with the following format prints:

- Number of the statement where the error occurred (A).
- Seven asterisks to bring attention to the error (B).
- Diagnostic message (C). This message is a standard EBCDIC message or the EBCDIC message translated into Japanese and printed using the DBCS code system assigned to the CA-Easytrieve Plus system printer.
CA-Easytrieve Plus makes the selection of the message type when the DBCS options module is generated. See the CA-Easytrieve Plus *Installation Guide* for more details.

```
ERRFIELD W 2 A 1
```

```
(A) (B) (C)
2*****B055 INVALID LENGTH, TYPE OR DECIMAL PLACES - 1
```

The format of diagnostic messages is also described in the chapter “[Messages](#)” earlier in this guide.

Parameter (PARM) Listing

By default, CA-Easytrieve Plus produces a complete listing of PARM statement options following the statement listing. The heading OPTIONS FOR THIS RUN - precedes this list. Override is through the LIST NOPARM parameter of the PARM statement.

CLIST

The CLIST option produces a cross-reference between statement numbers and the relative storage locations where the machine language code for the statements begins. The code for a job can span one or more 4 KB blocks of storage. A set of entries is produced for each block. A header identifying the program storage block number on line (A) always prefices the CLIST. The next line (B) consists of the compiler name, maintenance level, date and time of the compilation, and the activity identification. The actual condensed list of offsets is a line pair: the first line (C) shows the hexadecimal offsets corresponding to the CA-Easytrieve Plus statement numbers depicted on line two (D) of the pair.

These line pairs are repeated for the length of each activity. END OF PROGRAM is printed at the end of all activities.

```

CLIST
(A) PROGRAM STORAGE BLOCK NUMBER 1
(B) E Z T PLUS 5.0A- 4/01/01-11.44-JSN00003
(C) OFFSET (HEX) 00F2 011E 0122 0126 0130
(D) STATEMENT NO. 4 5 6 7 8
END OF PROGRAM

```

DMAP

Code the DMAP option of the PARM statement to format a printed data map following the parameter listing. The data map is a table of all files and fields in the program together with their declared and default attributes. Imported field definitions also import appropriate information to the DMAP. The following exhibit shows the overall structure of the DMAP.

```

DMAP
work fields

file fields
activity 1 report 1 work fields
                    summary file fields
                    ...
                    report n fields
                    sum fields

FILE - (0008) FILENAME file-attributes

LOGICAL-RECORD - (008) EMP-JOB-LR

BASE ST  DSPL  LENGTH FMT DEC OCCURS ED M R KE LVL NAME
      0000    46  A          01 EMPLOYEE
      0028    6  AV 32,767 .. D . . . 02 START-DATE-0415
      0028    2  N          03 START-YEAR-0415
      002A    2  N          03 START-MONTH-0415
      002C    2  N          03 START-DAY-0415

FILE - (0008) FILENAME DLI (DBDNAME)

RECORD - (0009) SKILL ROOT

BASE ST  DSPL  LENGTH FMT DEC OCCURS ED M R KE LVL NAME
      0000    4    N          .K . . D. 01 SKILL_ID-0455

RECORD - (0009) EXPERTISE PARENT : SKILL

BASE ST  DSPL  LENGTH FMT DEC OCCURS ED M R KE LVL NAME
      0000    4    N          .K . . K.

...
activity n ...
...

```

Field Group Header

The first line of the DMAP is an identifying header preceding each field group. The field group header has the format:

field-type [-(*integer*)] *name*

field-type *Field-type* is WORK, FILE, or REPORT. When the field type is FILE, the attributes from your FILE statement definition also display.

-(*integer*) *Integer* is the relative CA-Easytrieve Plus file number or the report group relative reference number.

name *Name* is the file name or report name being mapped. When a report is mapped, a second field group header specifies the relative report reference number.

Logical Record Display (IDMS)

This IDMS-only portion of the DMAP displays logical record data. VERB occurs once for each valid verb. Its values are OBTAIN, STORE, MODIFY, and ERASE. KEYWORDS are 32 character IDMS-defined names printed three across.

Record Display (DLI)

For DLI files ROOT identifies the DLI root record, and PARENT identifies the DLI record's parent record.

Field Header

In each field group a field header identifies attributes associated with each field. The field header has the format:

BASE ST DSPL LENGTH FMT DEC OCCURS ED M R KE LVL NAME

base—*base* is the identifier for the storage block where the field is stored. Only working storage fields and internally generated work fields have BASE specified.

st—*st* is a one-character indicator of storage type (W, S, or I).

dspl—*dspl* is the hexadecimal representation of the relative displacement of the field from the beginning of the record or storage block.

length—*length* indicates the decimal length of the field in bytes.

fmt—*fmt* identifies the data type, or format, of the fields:

A (ALPHA) - alphanumeric

M (MIXED) - mixed

K (KANJI) - double byte

V (VARYING) - varying length field

B (BINARY) - binary

I (INDEX) - this entry is an index for the most recent field that is not an index.

P (PACKED) - packed decimal

U (UNSIGN) - unsigned packed decimal

Z (ZONED) - zoned decimal

dec—*dec* is the number of decimal positions for a quantitative field.

occurs—*occurs* is the value in the OCCURS clause.

ed—*ed* indicates whether (H) HEX, (B) BWZ, or (K) KANJI was specified for the field.

m—*m* identifies the mask associated with the field. Mask indicators A through Y specify installation default or programmer specified masks.

r—*r* designates that the R (RESET) option was specified for a W field.

ke—*ke* is the indicator for key fields. Possible entries are D (DLI key) or C (IDMS CALC key).

lvl—*lvl* is the level of the field. Levels 2 through 5 are indented, and 6 through 50 are at the same level as 5.

name—*name* is the name of the field.

PMAP

Code the PMAP option to request a formatted map of the compiled code. CA-Easytrieve Plus prints this listing after the statement and DMAP listings. The PMAP is prefaced by a header that identifies the program storage block number. The following example illustrates the formatted listing of compiled code.

PMAP

PROGRAM STORAGE BLOCK NUMBER 1

```
0000      BC  F0  F032
...
0180  20  PACK 71  3118  5001
0186      CP  71  3118  6F58
018C      BC  70  B1A6
0190  21  LA   00  0015
|         |         |
|         |         | generated code
|         |         |
|         |         | statement number
|         |         |
|         |         | relative displacement
```

The first column of the map in the previous example is the relative displacement (in hexadecimal) from the origin of the storage block. The statement number corresponds to the statement listing numbers and marks the beginning of generated code for that statement. The actual code is next in opcode/operand format. Interspersed throughout the PMAP are titles and literals as they appear in your source program. The code to support the STATE and FLOW options for each statement is not printed.

XREF

The XREF option of the PARM statement produces a cross-reference listing for all file names, field names, procedure names, report names, segment names, and statement labels. This listing follows the statement, DMAP, and PMAP listings. The following example illustrates the format of the cross-reference listing.

XREF LONG

SYMBOL	DEFINED AT	REFERENCES
COMPUTE-OVERTIME	15	27
FLDW	3	37
FLDS	4	
NAME	7	
NET-PAY	8	23 34
OVERTIME	9	44

Execution Listing

When each activity ends, CA-Easytrieve Plus optionally produces file statistics that provide information on the files used during the activity. The file statistics are prefaced by a header, the compiler name, the maintenance level, the date and time of compilation, and the activity identification.

Note: Database file statistics are only maintained during automatic retrieval.

FILE STATISTICS - E Z T PLUS 4.0A 3/17/01-14.05-JSN00021

(A)	(B)	(C)	(D)	(E)	(F)	(G)
ESTIN	48	INPUT	SAM	FIX BLK	150	1800
EZTR001	3	OUTPUT	VFM	FIX UNBLK	29	N/A

Column (A) is the file name.

Column (B) is the record count.

Column (C) is the file type.

Column (D) is the file access method.

Column (E) is the file format.

Column (F) is the logical record length.

Column (G) is the block size (or VSAM control interval size).

Virtual files (VFM) are automatically blocked.

Abnormal Termination

Most programming errors fall into two categories:

- Syntax errors
- Execution errors

When CA-Easytrieve Plus encounters a syntax error, it prints diagnostic messages that pinpoint the error and terminates after completing the compilation of the entire program.

When CA-Easytrieve Plus encounters an execution error, it prints a diagnostic message for the error and terminates immediately.

If the error was generated by an interrupt code of 1 through 11, CA-Easytrieve Plus optionally produces an error analysis report through the ABEXIT parameter of the PARM statement.

Syntax Errors

Most of the errors made in programming are syntax errors relating to clerical mistakes or a misunderstanding of the language used. The CA-Easytrieve Plus simple syntax rules and logical program structure nearly eliminate these errors. To pinpoint violations, CA-Easytrieve Plus provides an extensive set of diagnostic messages. See the chapter “[Messages](#)” earlier in this guide.

Execution Errors

You can also encounter execution errors, most of which are easily remedied. The execution errors that CA-Easytrieve Plus intercepts include:

- Insufficient storage
- File OPEN errors
- Table file out-of-sequence
- Database errors
- Program interrupts 1 through 11

You can code `DEBUG(FLDCHK)` on the `PARM` statement to request that CA-Easytrieve Plus validate all references to data fields at execution time. This validation detects invalid file field references such as referring to a field in a file after end-of-file. When `FLDCHK` is active, and CA-Easytrieve Plus detects an invalid field reference, it produces the message:

```
*****A010 INVALID FILE REFERENCE - MASTER
```

By examining the statement involved, you can resolve the great majority of errors detected at execution time. The error message indicates the number of the incorrect statement if the `STATE` or `FLOW` options of the `PARM` statement are in effect. For program interrupts, you may need to analyze the problem in more depth. For interrupt codes 1 through 11, CA-Easytrieve Plus provides the error analysis report and supporting `DEBUG` options.

The operating system detects executions errors and gives a cross-referenced diagnostic for such things as input data validity, data set format, illogical access method requests, security violations, and program interrupts.

Error Analysis Report

The error analysis report has eight possible sections:

Section 1

```
27 *****A006 PROGRAM INTERRUPT - CODE 7 (DATA EXCP)
```

Section 2

```
INTERRUPT OCCURRED AT 0130 BLOCK 1 FROM EP E Z T PLUS 5.3 - 12/01/00
```

Section 3

```
INSTRUCTION AT 09D140 IS FA775AA85AB8
FIRST OPERAND ADDRESS 0A7AA8 CONTENTS 000000000000001C
SECOND OPERAND ADDRESS 0A7AB8 CONTENTS 00000000000000B5
```

Section 4

```
FLOWTABLE - MAXIMUM ENTRIES 100
 16 18 31 19 43 20 21 51
```

Section 5

```
PSW AT INTERRUPT 078D0007 EC09D146
```

Section 6

```
REGISTERS AT INTERRUPT
R0 0000001B R1 00097784 R2 000988DC R3 000A7BA0 R4 00097000
R8 000975B8 R9 000A7FA0 R10 8009865A R11 0009D010 R12 00000017
```

Section 7

FILE ID/NAME	RECORD ADDRESS	RECORD LENGTH	STATUS
0001 SYSPRINT	09EF78	56	ACTIVE
0002 SYSIN	000000	80	CLOSED
0003 WORK	000000	0	CLOSED
0004 EZTPRE	09C8F8	8	CLOSED
0005 TESTIN	09C8F8	150	ACTIVE

Section 8
WORKING STORAGE

BLOCK ADDRESS

0001 097000
0002 0A7000

- | | |
|-----------|--|
| Section 1 | Identifies the statement number where the interrupt occurred (if the STATE or FLOW option of the PARM statement were coded) and the type of interrupt. |
| Section 2 | Gives the relative displacement of the failing machine instruction from the entry point of the indicated activity. |
| Section 3 | Gives the storage location of the failing machine instruction and its hexadecimal image. It also lists the operands of the instruction and their storage addresses. |
| Section 4 | <p>The optional FLOW table.</p> <p>If the FLOW option is in effect, and an abnormal termination occurs, CA-Easytrieve Plus prints a formatted list consisting of statement numbers. The list is prefaced by the header FLOW TABLE - MAXIMUM ENTRIES 100, where the 100 is set at installation time or by FLOWSIZ on the PARM statement. See the CA-Easytrieve Plus <i>Reference Guide</i> for more information. The list of statement numbers follows the header and is read left-to-right, top-to-bottom, corresponding to the most recently executed statements. The flow table is created in a wraparound manner.</p> |
| Section 5 | Shows the Program Status Word (PSW) at the time of the interrupt. |

- | | |
|-----------|---|
| Section 6 | Lists the general purpose registers at the time of the interrupt |
| Section 7 | Lists the files used, the address of the current record of each file, the record length of each file, and the file's status. |
| Section 8 | Lists the starting location for each working storage block.

For OS, the SNAP dump option prints the standard (formatted) portion first, followed by the save area trace and the storage areas. |

Cause A Data Exception

CA-Easytrieve Plus enhanced debugging aids process only program interrupt codes 1 through 11. In the following example, we contrive an interrupt code 7 (data exception) at statement number 27.

```
24 ***** CAUSE A DATA EXCEPTION *****
25     DEFINE  BADDATA W 2 A VALUE  '$$'
DEFINE  WORSTDATA BADDATA 2 N 0
27     WORSTDATA = 1 + WORSTDATA
28 ***** CAUSE A DATA EXCEPTION *****
```

To generate the data exception, define an alpha field and give it an initial value (statement 25), then redefine the alpha field as numeric (statement 26). When the field WORSTDATA is used in a numeric computation, the operating system detects the invalid numeric value \$\$ and generates a data exception. The interrupt is passed to CA-Easytrieve Plus, and it prints an error analysis report.

Analyzing the Report

When you analyze the error analysis report, first determine the statement number where the interrupt occurred. There are numerous methods for finding the statement:

Section 1

If the STATE or FLOW options were active, the error analysis report contains the statement number (27) in the message.

```
27 *****A006 PROGRAM INTERRUPT - CODE 7 (DATA EXCP)
```

If you did not specify STATE or FLOW, you can locate the failing statement in two other ways:

```
INTERRUPT OCCURRED AT 01BC BLOCK 1 FROM EP E Z T PLUS 4.0A-3/19/84 9.4TEST
```

Section 2

If the CLIST option is in effect, locate program storage block 1 from entry point EZT CA-Easytrieve Plus 4.0A - 3/19/84-9.44-TEST in the CLIST. When that is found, scan the offsets until the displacement (01BC in this case) is equal to or between two offsets.

If the displacement is equal to one of the offsets, the corresponding statement number is the failing statement. If the displacement is between two offsets, the lower offset corresponds to the failing statement, as illustrated in the following example:

```
PROGRAM STORAGE BLOCK NUMBER 1 E Z T PLUS 4.0A - 3/19/84 - 9.44-TEST
OFFSET (HEX) ... 0190 01A6 01D4 ...
STATEMENT NO. ... 21 27 29 ...
```

If the PMAP option is in effect, locate program storage block 1 from entry point EZT PLUS 4.0A- 3/19/84- 9.44-TEST in the PMAP listing. Scan down the offsets until the displacement (01BC) is found. When that is found, scan up to the nearest CA-Easytrieve Plus statement number (27). That is the failing statement.

```
0190 21 LA 00 0015
...
      BALR EF
01A6 27 LA 00 001B
01AA L F0 3158
01AE BALR EF
01B0 ZAP 70 65C0 6F48
01B6 PACK 71 65D0 6F50
01BC AP 77 65C0 65D0
01C2 TM 03 65C7
...
```

Section 3

Section 3 of the error analysis report identifies the actual machine instruction image and the operands involved in the failure.

```
INSTRUCTION AT 0ED1CC IS FA7765C065D0
FIRST OPERAND ADDRESS 0F75C0 CONTENTS 000000000000001C
SECOND OPERAND ADDRESS 0F75D0 CONTENTS 0000000000000BB5
```

The instruction image is for an AP (add packed) instruction, and the second operand is not a valid packed number, thus the data exception. (0BB5 is the result of packing 5B5B (\$\$), before the add.)

Section 4

Section 4 of the error analysis report is the FLOW trace. It shows the immediately preceding statement numbers. The last number listed is the failing statement.

```
FLOW TABLE - MAXIMUM ENTRIES 100
 16 17 18 31 19 43 20 21 51 27
```

Section 5

Section 5 displays the operating system's Program Status Word (PSW) that you can use to derive the address of the failing instruction in the dump.

Section 6

Section 6 follows the PSW with a formatted listing of the general purpose registers at the time of interrupt. You can use the register's contents with the PMAP listing (or the dump) to do more detailed, classical debugging.

Section 7

Section 7 locates file fields for error determination. To locate the field, first determine the file that contains the field by referencing the DMAP listing. When you locate the field in the DMAP, add the displacement (DSPL - from the DMAP) and the corresponding RECORD ADDRESS from the error analysis report. This gives the storage location for the field.

For example, if the field was SOCSECNUM, you reference the DMAP listing and find:

DMAP

WORK FIELDS

```
FILE - (0005) TESTIN
BASE    DSPL    LGH DEC TYPE OCCR MASK EDIT NAME
... 0003 5    PACKED  A    SOCSECNUM
...
```

Then, referencing the error analysis report, you would find:

FILE ID/NAME	RECORD ADDRESS	RECORD LENGTH	STATUS
0005 TESTIN	0EF8F8	150	ACTIVE
...			

```
adding:    RECORD ADDRESS 0EF8F8
to:        DSPL          0003
```

You get storage location: 0EF8FB, which is the storage address of the field SOCSECNUM. In the SNAP dump, that location would appear as:

```
0EF8E0.... .... F4F0F102 5305228C *.....*
```

Section 8

Section 8 locates working storage fields in a dump. To locate the field, reference the DMAP listing and determine the working storage block number and the displacement of the field. Now add the displacement (DSPL) to the corresponding BLOCK ADDRESS from the error analysis report.

For example, to locate the field WORSTDATA in the dump, you reference the DMAP listing and find:

```
DMAP
WORK FIELDS

BASE 0 DSPL LGH DEC TYPE OCCR MASK EDIT NAME
0002 W 0F50 2 ZONED WORSTDATA
...
```

Then, referencing the error analysis report, you would find:

```
WORKING STORAGE

BLOCK ADDRESS

0001 0E7000
0002 0F7000

adding: BLOCK ADDRESS 0F7000
to: DSPL 0F50
```

You get storage location: 0F7F50, which is the storage address of the field WORSTDATA. In the dump the field appears as:

```
0F7F40 .... 5B5B0000 .... *.....*
```

System-Defined Fields and Return Codes

CA-Easytrieve Plus automatically provides four categories of system-defined fields:

- General
- File related
- Report related
- Screen related

General Fields

CA-Easytrieve Plus automatically provides the following general fields.

SYSDATE

SYSDATE is an 8-byte alphanumeric field that contains the system date at the start of CA-Easytrieve Plus execution. SYSDATE is refreshed with the current date during each terminal I/O associated with a SCREEN. The DATE option set in the site options determines the format of the date. A slash (/) normally separates the month, day, and year components of the date (for example, *mm/dd/yy*).

SYSTIME

SYSTIME is an 8-byte alphanumeric field that contains the system time at the start of CA-Easytrieve Plus execution. SYSTIME is refreshed with the current time during each terminal I/O associated with a SCREEN. A colon (:) normally separates the data into hours, minutes, and seconds (for example, *hh:mm:ss*).

RETURN-CODE

RETURN-CODE is a 4-byte binary field whose contents are returned to the OS/390 or z/OS operating system in register 15 when CA-Easytrieve Plus terminates. RETURN-CODE is initialized to zero, but you can set it to any value.

File Fields

CA-Easytrieve Plus automatically provides the following special data fields for each of your files. These fields are stored as part of working storage but can be qualified by file name. As working storage fields, they are not subject to invalid file reference errors.

RECORD-LENGTH

RECORD-LENGTH is a 4-byte binary field used for all file types to determine or establish the length of the current data record. For variable length records, this field contains only the length of the record's data. CA-Easytrieve Plus automatically adjusts the field to account for the 4-byte record-control word and 4-byte block-control word. For variable-length files, assign the length of the record to the RECORD-LENGTH field before the PUT or WRITE statement is executed.

RECORD-COUNT

RECORD-COUNT is a read-only 4-byte binary field that contains the number of logical input operations performed to the file.

FILE-STATUS

FILE-STATUS is a read-only field that contains the results of the most recent I/O operation on a file. FILE-STATUS is available when you code STATUS on the I/O statement. If you do not code STATUS, an appropriate error message is generated. The error message contains one of these codes.

FILE-STATUS codes and their meanings are:

0000 Operation successful.

This is not an error condition. It indicates that the last I/O operation was successful. No additional information is required.

0004 End of file.

This is not an error condition. It indicates that the file position was moved beyond the last record in the file.

This condition occurs following a GET statement when the current record is the last record in the file. It can occur for sequential, indexed, and relative files.

0008 Record with a duplicate alternate key exists.

This is not an error condition. It indicates that the key of this record matches the key of the record that follows it in the sequential order of this file.

This condition can occur following a GET or READ statement for an indexed file that does not have unique keys.

Following a GET statement, this condition indicates that at least one more record with a matching key is waiting to process.

Following a READ statement, this condition indicates that there is at least one more record in the file with a matching key (a GET statement must be used to retrieve any remaining records).

In CICS/VS, OS/390 (batch and TSO), z/OS, and CMS/OS, an indexed file can have non-unique keys if the associated data set is a VSAM PATH and the auxiliary index data set was defined with non-unique keys.

0012 Duplicate key.

This error condition indicates that there was an attempt to store a record with a duplicate key, or there is a duplicate record for an alternate index with the unique key option.

This condition can occur following a PUT or WRITE ADD statement for an indexed file or a PUT statement for a relative file.

For an indexed file, it indicates that the key of the record matches the key of a record already present in the file. For a relative file, it indicates that the slot designated by the relative record number already contains a record (the slot is not empty).

This condition can also occur following a WRITE UPDATE statement for a sequential or indexed file. It indicates that:

- There is at least one alternate index associated with this file.
- The alternate index was defined with the unique key and the upgrade option.
- The updated record caused a duplicate key condition to occur when the alternate index was updated.

0016 Record not found.

This error condition indicates that the record designated by the KEY parameter is not found in the file.

This condition can occur following a READ or POINT statement for an INDEXED or RELATIVE file. For an INDEXED file, it indicates that no record in the file matches the key specified by the statement. For a RELATIVE file, this condition indicates that the slot designated by the relative record number is empty.

0020 Record locked.

This error condition indicates that there was an attempt to access or update a record that has a lock placed on it by another process.

This condition is possible only on the PC.

0024 Logical or physical error condition.

This error condition indicates that the access method routines used to access the file detected a logical or physical error condition. The specific cause of the error is displayed in a runtime abend message.

Report Fields

CA-Easytrieve Plus automatically provides the following special data fields for your reports. These fields are stored as part of working storage and are read-only.

LINE-COUNT

LINE-COUNT is a 2-byte binary field that contains the number of lines printed on the page.

LINE-NUMBER

LINE-NUMBER is a 2-byte binary field that contains the number of the line being printed within the line group.

PAGE-COUNT

PAGE-COUNT is a 2-byte binary field that contains the number of pages printed.

PAGE-NUMBER

PAGE-NUMBER is a 2-byte binary field that contains the number of the page printed.

TALLY

TALLY is a 10-byte packed decimal field that contains the number of detail records in a control break.

LEVEL

LEVEL indicates the control break level. See the CONTROL Statement.

BREAK-LEVEL

BREAK-LEVEL indicates the highest field in the break.

Screen Fields

CA-Easytrieve Plus automatically provides the following special data fields for your screens. These fields are stored as part of working storage and are read-only.

KEY-PRESSED (PC Only)

KEY-PRESSED is a 2-byte binary field that contains a value representing the most recent terminal key pressed by the terminal user.

CA-Easytrieve Plus automatically defines symbolic names that correspond to values for the most common keys. Only keys with symbolic names can be used on a KEY statement.

Terminal Key	Symbolic Name	Constant Value
Unknown		0
Enter	ENTER	1
Clear	CLEAR	11
PA1 thru PA3	PA1 thru PA3	12 thru 14
PF1 thru PF24	F1 thru F24	21 thru 44
F1 thru F12	F1 thru F12	21 thru 32
Test Request		220
Op ID Card Reader		222
Magnetic Slot Reader		223
Trigger Action		224
Structured Field		230
Clear Partition		231
Read Partition		232
No Aid Generated		255

TERM-COLUMNS (PC Only)

TERM-COLUMNS is a 2-byte binary field containing the maximum number of columns the screen supports. You can test TERM-COLUMNS to execute a SCREEN activity designed specifically for the terminal being used.

TERM-ROWS (PC Only)

TERM-ROWS is a 2-byte binary field containing the maximum number of rows the screen supports. You can test TERM-ROWS to execute a SCREEN activity designed specifically for the terminal being used.

TERM-NAME (PC Only)

TERM-NAME is a 16-byte alphanumeric field containing the terminal identification. This field is set only in CICS environments.

SYSUSERID (PC Only)

SYSUSERID is a 16-byte alphanumeric field identifying the terminal user.

Symbol References

The following list contains CA-Easytrieve Plus symbols. Associated with each symbol are one or more references. The references describe the various ways in which you can use the symbol. An R in the column after the symbol indicates it is reserved.

SPECIAL SYMBOL	RESERVED	REFERENCE
.		Syntax delimiter (period) Macro parameter concatenation (period)
<		Conditional expression
<=		Conditional expression
(Syntax delimiter (left parenthesis)
:		Syntax delimiter (colon)

SPECIAL SYMBOL	RESERVED	REFERENCE
+		Assignment Continuation of statements and words DISPLAY LINE TITLE
&		Macro variable prefix
*		Assignment Comment statement DEFINE
)		Syntax delimiter (right parenthesis)
¬<		Conditional expression POINT
¬>		Conditional expression
¬=		Conditional expression
-		Assignment Continuation of statements and words DISPLAY LINE TITLE
**	R	Reserved for future use
/		Assignment
'		Syntax delimiter (single quote)
%		Macro invocation
>		Conditional expression
>=		Conditional expression POINT
,		Syntax delimiter (comma)
=		Assignment Conditional expression POINT
@	R	Reserved for future use

Reserved Words

The following list includes all CA-Easytrieve Plus reserved words. The reserved words are listed in alphabetical order.

ACCESS	EQ	LQ	RETURN-CODE
AFTER-BREAK	ERROR	LS	ROLLBACK
AFTER-LINE	EXECUTE	LT	ROW
AFTER-SCREEN	EXIT	MASK	S
AND	EXTERNAL	MATCHED	SCREEN
ATTR	F1, F2,..F24	MESSAGE	SEARCH
BEFORE	FETCH	MOVE	SECONDARY
BEFORE-BREAK	FILE	NE	SELECT
BEFORE-LINE	FILE-STATUS	NEWPAGE	SEQUENCE
BEFORE-SCREEN	FILL	NOMASK	SIZE
BREAK-LEVEL	FINAL	NOPRINT	SKIP
BUSHU	FIRST	NOT	SOKAKU
BY	FIRST-DUP	NOTE	SORT
CALL	FOR	NOTITLE	SQL
CASE	GE	NOVERIFY	STOP
CHECKPOINT	GET	NQ	SUM
CHKP	GO	NULL	SYSDATE
CHKP-STATUS	GOTO	OF	SYSIN
CLEAR	GQ	OR	SYSIPT
CLOSE	GR	OTHERWISE	SYSLST
COL	GT	PA1..PA3	SYSPRINT
COLOR	HEADING	PAGE-COUNT	SYSSNAP
COMMIT	HEX	PAGE-NUMBER	SYSTIME
CONTROL	HIGH-VALUES	PARM-REGISTER	SYSUSERID
COPY	IDD	PATH-ID	TALLY
CURSOR	IDMS	PATTERN	TERM-COLUMNS
D	IF	PERFORM	TERM-NAME
DECLARE	IN	POINT	TERM-ROWS
DEFAULT	INITIATION	POS	TERMINATION
DEFINE	INSERT	PRIMARY	TITLE
DELETE	JOB	PRINT	TO
DENWA	JUSTIFY	PROC	TRANSFER
DISPLAY	KANJI-DATE	PROCEDURE	TRC
DLI	KANJI-TIME	PROGRAM	UNIQUE
DO	KEY	PUT	UNTIL
DUPLICATE	KEY-PRESSED	READ	UPDATE
E	KOKUGO	RECORD	UPPERCASE
ELSE	KUN	RECORD-COUNT	VALUE
ELSE-IF	LAST-DUP	RECORD-LENGTH	VERIFY
END	LE	REFRESH	W
END-CASE	LEVEL	RELEASE	WHEN
END-DO	LIKE	RENUM	WORK

END-IF	LINE	REPEAT	WRITE
END-PROC	LINE-COUNT	REPORT	X
ENDPAGE	LINE-NUMBER	REPORT-INPUT	XRST
END-REPEAT	LINK	RESHOW	
ENDTABLE	LIST	RESTART	
ENTER	LOW-VALUES	RETRIEVE	
EOF			

Note: In addition to the previous list, all keywords on a statement are reserved for that statement. For example, even though NAME is not on the previous list of reserved words, it is reserved on the JOB statement because it is a keyword.

Index

D

data
 exception, 2-16
 map, 2-8
DBCS options module, 2-2

E

error messages
 asterisk prefixed, 1-40
 CA-PanAudit Plus, 1-49
 DB2, 1-41
 execution, 2-13
 format, 1-1
 IDD interface, 1-40
 operational diagnostic, 1-2
 syntax, 2-13
 syntax checking, 1-8
extended reporting options module, 2-3

F

field definitions, system-defined fields, 2-20

K

KEY statement, 2-24

O

options
 module
 DBCS, 2-2
 extended reporting, 2-3
 table, regeneration of, 2-2

P

PARM List, 2-7

R

recreating the options table, 2-2
reports, system-defined fields, 2-23

S

SCREEN activity, 2-24
screen fields, 2-24
symbol references, 2-25
system-defined fields
 and return codes, 2-19
 file, 2-20
 report, 2-23
 screen, 2-24

T

terminal keys, KEY-PRESSED, 2-24

EZTPX03, 2-2

EZTPX04, 2-2

EZTPX05, 2-3

U

utility programs
EZTPX01, 2-2

W

working storage fields, 2-20