

ADACDC Utility Error and Warning Messages

Overview of Messages

ERROR-121 | ERROR-122 | ERROR-123 | ERROR-124 | ERROR-125 | ERROR-126 |
WARNING-127 | WARNING-128 | ERROR-129 | WARNING-130 | ERROR-131 |
WARNING-132 | WARNING-133 | WARNING-134 | ERROR-135 | WARNING-136 |
ERROR-137 | ERROR-138 | WARNING-142 | WARNING-143 | ERROR-144 |
ERROR-145 | ERROR-146 | ERROR-147

ERROR-121 Open failed for *filename* file

Explanation ADACDC attempted to open the file *filename* but the open failed. Processing cannot continue with this file; utility execution is terminated.

Action Provide the missing file in the ADACDC job.

ERROR-122 Read error for *filename* file

Explanation An error occurred reading the file *filename*; utility execution is terminated. This error generally occurs when the format of the file indicated is not valid.

Action Ensure that the file in question has a valid format as specified in the ADACDC documentation. In addition, check any special considerations for the operating system where the utility is being executed.

ERROR-123 Write error for *filename* file

Explanation An error occurred writing to the file *filename*; utility execution is terminated. This error normally occurs when the output file is full, but it may also occur when the format of the file indicated is not valid.

Action If the file is full, increase the size of the file and rerun the job. Otherwise, ensure that the file in question has a valid format as specified in the ADACDC documentation. In addition, check any special considerations for the operating system where the utility is being executed.

ERROR-124 Unable to write data to *filename* file

Explanation The ADACDC utility was not able to write data to the file *filename* because the amount of data to be written exceeded the maximum data that could be written to this file. Additional input in the message indicates the length of data that was to be written and the maximum data that can be written to the file.

Action Increase the block size on the file to accommodate the largest record that may be written to the file.

ERROR-125 Invalid transaction file 'CDCTXI'

Explanation The ADACDC utility opened the CDCTXI file and successfully read the first record; however, the first record is not a transaction file record produced by the ADACDC utility.

Action Supply a valid input transaction file to the utility or specify the RESETTXF so that the CDCTXI specification is ignored if the file name is correct.

ERROR-126 Inconsistent PLOG input provided: current DBID=*olddb* PLOG=*oldplog* BLOCK=*oldblock* received DBID=*newdb* PLOG=*newplog* BLOCK=*newblock* PLOGS provided to ADACDC must be in sequence

Explanation The ADACDC utility encountered inconsistent input while processing the input PLOG data. It was expecting the next block based on the values *olddb*, *oldplog*, and *oldblock*, but found that the next block to be processed was *newdb*, *newplog*, and *newblock*. This can occur if

- a PLOG is encountered that was produced by a different database ID;
- *newplog* and *oldplog* are the same, but *newblock* is not 1 more than *oldblock*, or *newplog* is less than *oldplog*.

Action PLOGs must be provided to the ADACDC utility in the chronological order in which they were created. Provide the PLOG input in the correct chronological order. If this has occurred due to information on the input transaction file and you still wish to process the PLOGs provided, specify the RESETTXF option.

WARNING-127 Conversion error, Response Code=55.
Length error decompressing field file=*file* ISN=*isn* fieldname=*name*
PE-INDEX=*peindex*
The record for the ISN will be written with a flag indicating that the record was not decompressed successfully.
This occurs when the decompressed value for the field is longer than the default length for the field in the FDT. Data can legally be written longer than this value or the FDT definition may have changed.

Explanation An error occurred while decompressing a record (on file *file* for ISN number *isn*, field name *name*, and PE Index *peindex*) before writing it to the primary output file. The record cannot be decompressed. This can occur if

- the FDT for a file has changed; or
- a data area longer than the default for the field in the FDT has been written, which can legally occur.

The record is written with an indication that decompression failed.

Action If the FDT has changed, the record cannot be decompressed unless the FDT is changed back. If the record(s) being written are longer than the default value for the field in the FDT, either increase the default value to accommodate the larger field sizes or avoid writing fields longer than the default value.

WARNING-128 Input transaction file is empty

Explanation The CDCTXI file was successfully opened but was empty; no transaction information is available.

Action If you are aware that the input transaction file is empty, you can ignore this warning; however, if you expect the input transaction file to contain some data, you need to determine why ADACDC is seeing an empty file.

ERROR-129 *value* invalid for PHASE parameter

Explanation The *value* specified for the PHASE parameter is invalid.

Action Specify "1", "2", or "BOTH", as appropriate.

WARNING-130 FDT for file *file* not found

Explanation A data record is decompressed before writing to the primary output file. Before the record can be decompressed, the FDT for the file *file* must exist in the database. In this case, the FDT was not found; so the record could not be decompressed. Processing for the file is disabled after this message is issued.

Action Make the FDT for the file available again in the database and rerun the job with FILE=file to process the records for the file in question.

ERROR-131 Database ID mismatch ADARUN DBID=*rundbid*, extract file DBID=*extdbid*

Explanation The first record was read from the extract file provided for a phase 2 execution of the ADACDC utility. This record indicates that the extract file was generated for the *extdbid* database; however, the ADARUN DBID is *rundbid*.

Action Run the phase 2 job using the correct database.

WARNING-132 Inconsistent NOET specification transaction data on CDCTXI ignored

Explanation ADACDC detected that NOET was specified on the previous run of ADACDC, but not on this run-or vice versa. Changing the NOET parameter between runs renders the data on the transaction file redundant; therefore, this data is ignored.

Action If this is expected, the message may be ignored. If it is not expected, you should determine why the ADACDC utility is being run with different options.

**WARNING-133 Decompressed record too long. Response Code=53
Length error decompressing entire record file=*file* ISN=*isn*
The record for the ISN will be written with a flag indicating that the record was not decompressed successfully.
This will occur when the maximum length available on the CDCEXT/CDCOUT output file is insufficient to contain the decompressed record. Increase the maximum record length for the output file.**

Explanation Before writing the the ISN (*isn*) for the file (*file*) to the primary output data set, an attempt was made to decompress the data. This attempt failed because the maximum length of output that can be written to the primary output data set is not large enough to accommodate the decompressed record. The record is written to the primary output data set with an indicator in the CDCO/CDCH header that the record is still in compressed format.

Action Increase the block size of the primary output data set. If the data record requires more than 32K when decompressed, it cannot be supported by the ADACDC utility.

WARNING-134 Error occurred during the data decompression for File=*file*, ISN=*isn*, SeqNumber=*seqnum*. RC=*rc*, ResponseCode=*rspcode*, ResponseSubcode=*subcode* was given from the decompression routine.
Report this to your DBA.
The record will be written with a flag 'Still compressed'.

Explanation A decompression error occurred while decompressing the data record for the unique sequential number *seqnum* for ISN *isn* on file *file* prior to output to the primary output data set. The return code (*rc*), response code (*rspcode*), and subcode (*subcode*) are given in the message. The record is written to the primary output data set with an indicator in the CDCO/CDCH header that the record is still in compressed format.

Action Report the error to your DBA.

ERROR-135 Error limit exceeded

Explanation The error limit is exceeded if the combined number of WARNING-127, WARNING-133, or WARNING-134 messages exceeds 99. This error generally indicates a fundamental problem with FDT definition(s) that should be corrected.

Action Correct the reason for some or all of the warning messages and rerun the job.

WARNING-136 PLOG(S) may be missing current=*currplog* next=*nextplog*

Explanation While reading the input PLOGs, ADACDC detected that the PLOG number jumped from *currplog* to *nextplog* indicating that PLOG numbers in between may have been skipped.

This can legally occur during online saves because they increase the session number without the PLOG number being changed. This causes the next PLOG started to use a number more than 1 greater than the previous PLOG.

If that does not explain the difference, verify that all appropriate PLOGs for the database have been provided in sequence.

ERROR-137 PLOG *plognumber* not of current version

Explanation ADACDC cannot handle protection logs (PLOGs) from previous versions.

Action Use the ADACDC from the load library of the appropriate version.

ERROR-138 PLOG *plognumber* was produced by a cluster nucleus session and has not been merged yet

Explanation The protection log (PLOG) was created by an Adabas cluster nucleus and has not yet been merged with other PLOGs using an ADARES PLCOPY procedure.

Action Use the ADARES PLCOPY function before extracting data using ADACDC.

WARNING-142 IGNORESPANNED parameter specified, Spanned record has been skipped

Explanation The IGNORESPANNED parameter was supplied. All spanned records found in the PLOG will be ignored.

Action If processing spanned records is required, omit the IGNORESPANNED parameter and rerun the ADACDC job. Otherwise, no action is required for this informational message.

WARNING-143 Secondary spanned segment without associated primary spanned segment encountered for File = *file*, Primary ISN = *isn*, SeqNumber = *seqnum*. Such spanned record cannot be decompressed. The record will be written with a flag 'Still compressed'.

Explanation A standalone secondary segment (that is, a secondary segment without a primary segment) of the spanned record identified with the unique sequential number *seqnum* for primary ISN *isn* on file *file* was encountered in the protection log (PLOG). This segment cannot be decompressed. The record is written to the primary output data set with an indicator in the CDCH header that the record is still in compressed format.

Action No action is required for this informational message. Probably Adabas was started with the ADARUN nucleus parameter SRLOG=UPD, which instructs the nucleus to write to the PLOG only modified spanned record segments. If it is necessary to include all spanned PLOG records into ADACDC processing, restart the Adabas nucleus after changing the ADARUN SRLOG parameter to ALL or PART.

ERROR-144 ISN chain for spanned record is not intact. File=*file*, Primary ISN = *isn*, SeqNumber = *seqnum*. Report this to your DBA.

Explanation The primary and secondary segments of a spanned record are connected using their ISNs. A secondary segment was found in the PLOG with a different ISN from what was expected.

Action Report this problem to your DBA.

ERROR-145 Length of compressed spanned record exceeds the maximum PLOG/TXI record length multiplied by SEGLFAC value.
File = *file*, Primary ISN = *isn*, SeqNumber = *seqnum*.
Report this to your DBA.

Explanation The length of the buffer for assembling spanned records is not large enough to contain the complete spanned record.

Action Report this problem to your DBA.

ERROR-146 Number of segments in spanned record exceeds SEGLFAC value.
File = *file*, Primary ISN = *isn*, SeqNumber = *seqnum*.
Report this to your DBA.

Explanation The number of segments of a spanned record exceeds the SEGLFAC value (maximum number of spanned record segments allowed in one spanned record).

Action Report this problem to your DBA.

ERROR-147 MAXLOGRECLLEN value less than minimum.
Must be at least 32768 bytes.

Explanation The value specified for the MAXLOGRECLLEN parameter is too small. Valid values must be at least 32768 bytes.

Action Correct the parameter value and rerun the job.