

COMPRESS Function Output

This chapter describes the ADACMP COMPRESS function output.

- Compressed Data Records
 - Rejected Data Records
 - ADACMP Report
-

Compressed Data Records

The data records that ADACMP has processed, edited, and compressed are written out together with the file definition information to a sequential data set with the variable blocked record format. This data set may be used as input to the ADALOD utility. The output of several ADACMP executions may also be used as input to ADALOD.

If the output data set contains no records (no records provided on the input data set or all records rejected), the output may still be used as input to the ADALOD utility. In this case, you must ensure that the amount of Associator space allocated to the file is sufficient since an accurate estimate cannot be made by the ADALOD utility without a representative sample of input record values (see the ADALOD utility for additional information).

For information on how to identify MU and PE occurrences greater than 191 in the compressed record, read *Identifying MU and PE Occurrences Greater Than 191 in Compressed Records*.

Rejected Data Records

Any records rejected during ADACMP compression are written to the DD/FEHL error data set. The records are output in variable blocked format and may be segmented into multiple physical records. Each logical rejected record will be preceded by an initial ADAF rejected record header. If the logical record and the ADAF header do not fit in the DD/FEHL physical record length, ADAN rejected record headers will precede the remaining physical rejected record segments that comprise the logical rejected record.

The functions of these two different headers are as follows:

- ADAF headers indicate the error condition and pertinent information.
- ADAN headers are smaller and are used for rejected record continuation and ADAH/ADAC header error reporting.

DSECTs for the ADAF and ADAN headers can be found in members ADAF and ADAN of the distributed Adabas SRCE data set.

Traditionally, the DD/FEHL error data set produced for ADACMP errors has truncated rejected records that exceeded the FEHL physical record length. In Version 8, the rejected records are segmented instead of truncated. Because of this change, the DD/FEHL LRECL setting must be at least 500 bytes.

If HEADER=YES is specified, an error may occur while segmented uncompressed records are being assembled into a logical record. If the ADAH header is in error, the ADAH record is written and subsequent ADAC records are not written until the next ADAH record is processed. If an ADAC header is in error, the preceding ADAH header will be written without its payload data. The ADAC record in error will be written in its entirety. Subsequent ADAC records are not written until the next ADAH record is processed.

The following response codes may occur:

X'E7'(231 - ADARSP231)	Input record too short (COMPRESS)
X'E8'(232 - ADARSP232)	Output record length error (COMPRESS)
X'E9'(233 - ADARSP233)	<p>An invalid ADAH spanned record header has been encountered. The following subcodes provide more detail:</p> <ul style="list-style-type: none"> ● 1 - incorrect ADAH eye-catcher ● 2 - incorrect ADAH header length ● 3 - unexpected continuation indicator ● 4 - reserved area not set to binary zeros ● 5 - invalid segment length ● 6 - total payload data length exceeds MAXLOGRECLLEN setting <p>For complete information about spanned records and the ADAH header, read <i>Spanned Records</i>. The DSECT for the ADAH header can be found in member ADAH of the distributed Adabas SRCE data set.</p>
X'EA'(234 - ADARSP234)	<p>An invalid ADAC spanned record header has been encountered. The following subcodes provide more detail:</p> <ul style="list-style-type: none"> ● 1 - incorrect ADAC eye-catcher ● 2 - incorrect ADAC header length ● 3 - unexpected continuation indicator ● 4 - reserved area not set to binary zeros ● 5 - invalid segment length ● 6 - unexpected continuation record sequence number ● 7 - invalid segment offset ● 8 - accumulated payload data length exceeds specified total length in ADAH ● 9 - accumulated payload data length exceeds MAXLOGRECLLEN setting <p>For complete information about spanned records and the ADAC header, read <i>Spanned Records</i>. The DSECT for the ADAC header can be found in member ADAC of the distributed Adabas SRCE data set.</p>

Only the first incorrect field within a record is detected and referenced. If there are other errors, they are not detected until subsequent runs are made.

Example of Rejected Data Records

The following table depicts the FEHL output for four rejected records during ADACMP compression. Rejected records 1 and 3 have only one FEHL record (ADAF); rejected record 2 is segmented into two FEHL records (ADAF and ADAN); rejected record 4 is segmented into three FEHL records (one ADAF record and two ADAN records):

Note:

DSECTs for the ADAF and ADAN headers can be found in members ADAF and ADAN of the distributed Adabas SRCE data set.

Rejected Record	FEHL Records	FEHL Fields		Description
		Field	Value	
1	ADAF	ADAFEYE	ADAF	ADAF header eye-catcher
		ADAFLEN	72	ADAF header length
		ADAFTYPE	R	Type. Valid values are: H: ADAH header P: ADAH header and payload R: Logical record
		ADAFIND	E	Continuation indicator. Valid values are: C: Continuation record to follow E: End of logical record (last segment)
		Reserved	0	Reserved
		ADAFSLEN	22000	Segment length (length of payload data following the header)
		ADAFOTL	22000	Total length. Note: The sum of the values of all ADANSLEN fields and ADAFSLEN should equal the value of ADAFTOTL (for record 1 in this example, 0 + 22000 = 22000)
		ADAFISN	1	ISN of record
		ADAFNUM	1	Logical record number
		ADAFNUM	1	Physical record number
		ADAFEOFF	5000	Error offset in logical record
		ADAFPEX	0	PE index
		ADAFFN	ZA	Field name
		ADAFRSP	41	Response code
		ADAFSUB	2	Subcode
		Reserved	0	Reserved
ADAFDATA	'Record 1 -- Payload Data'	Rejected input data		

Rejected Record	FEHL Records	FEHL Fields		Description
		Field	Value	
2	ADAF	ADAFEYE	ADAF	ADAF header eye-catcher
		ADAFLEN	72	ADAF header length
		ADAFTYPE	R	Type. Valid values are: H: ADAH header P: ADAH header and payload R: Logical record
		ADAFIND	C	Continuation indicator. Valid values are: C: Continuation record (ADAN) to follow E: End of logical record (last segment)
		Reserved	0	Reserved
		ADAFSLEN	27962	Segment length (length of payload data following the header)
		ADAFOTL	50000	Total length. Note: The sum of the values of all ADANSLEN fields and ADAFSLEN should equal the value of ADAFTOTL (for record 2 in this example, 22038 + 27962 = 50000)
		ADAFISN	2	ISN of record
		ADAFNUM	2	Logical record number
		ADAFNUM	3	Physical record number
		ADAFEOFF	35000	Error offset in logical record
		ADAFPEX	0	PE index
		ADAFN	ZA	Field name
		ADAFRSP	41	Response code
		ADAFSUB	2	Subcode
	Reserved	0	Reserved	
	ADAFDATA	'Record 2 -- Payload data part 1'	Rejected input data	
	ADAN	ADANEYE	ADAN	ADAN header eye-catcher
		ADANLEN	24	ADAN header length
		ADANTYPE	R	Type. Valid values are: C: ADAC header D: ADAC header and payload P: ADAH record segment R: Logical record segment
		ADANIND	E	Continuation indicator. Valid values are: C: Continuation record (ADAN) to follow E: End of logical record (last segment)
		Reserved	0	Reserved
		ADANSLEN	22038	Segment length (length of payload data following the header)
ADANOFF		27962	Error offset in logical record.	
ADANDATA		'Record 2 -- Payload data part 2'	Continued rejected input data	

Rejected Record	FEHL Records	FEHL Fields		Description
		Field	Value	
3	ADAF	ADAFEYE	ADAF	ADAF header eye-catcher
		ADAFLEN	72	ADAF header length
		ADAFATYPE	P	Type. Valid values are: H: ADAH header P: ADAH header and payload R: Logical record
		ADAFIND	E	Continuation indicator. Valid values are: C: Continuation record (ADAN) to follow E: End of logical record (last segment)
		Reserved	0	Reserved
		ADAFSLEN	20000	Segment length (length of payload data following the header)
		ADAFOTL	20000	Total length. Note: The sum of the values of all ADANSLEN fields and ADAFSLEN should equal the value of ADAFOTL (for record 3 in this example, 0 + 20000 = 20000)
		ADAFISN	3	ISN of record
		ADAFNUM	3	Logical record number
		ADAFNUM	4	Physical record number
		ADAFEOF	0	Error offset in logical record
		ADAFPEX	0	PE index
		ADAFN		Field name
		ADAFRSP	233	Response Code
		ADAFSUB	1	Subcode
		Reserved	0	Reserved
ADAFDATA	ADAH and payload data	Rejected input data		

Rejected Record	FEHL Records	FEHL Fields		Description
		Field	Value	
4	ADAF	ADAFEYE	ADAF	ADAF header eye-catcher
		ADAFLEN	72	ADAF header length
		ADAFATYPE	H	Type. Valid values are: H: ADAH header P: ADAH header and payload R: Logical record
		ADAFIND	C	Continuation indicator. Valid values are: C: Continuation record (ADAN) to follow E: End of logical record (last segment)
		Reserved	0	Reserved
		ADAFSLEN	32	Segment length (length of payload data following the header)
		ADAFOTL	10064	Total length. Note: The sum of the values of all ADANSLEN fields and ADAFSLEN should equal the value of ADAFTOTL (for record 4 in this example, 32 + 32 + 10000 = 10064)
		ADAFISN	4	ISN of record
		ADAFNUM	4	Logical record number
		ADAFNUM	8	Physical record number
		ADAFEOFF	0	Error offset in logical record
		ADAFPEX	0	PE index
		ADAFFN		Field name
		ADAFRSP	234	Response Code
		ADAFSUB	3	Subcode
	Reserved	0	Reserved	
	ADAFDATA	ADAH header	Rejected input data	
	ADAN	ADANEYE	ADAN	ADAN header eye-catcher
		ADANLEN	24	ADAN header length
		ADANTYPE	C	Type. Valid values are: C: ADAC header D: ADAC header and payload P: ADAH record segment R: Logical record segment
		ADANIND	C	Continuation indicator. Valid values are: C: Continuation record (ADAN) to follow E: End of logical record (last segment)
		Reserved	0	Reserved
		ADANSLEN	32	Segment length (length of payload data following the header)
		ADANOFF	32	Error offset in logical record.
	ADANDATA	ADAC header	Continued rejected input data	
	ADAN	ADANEYE	ADAN	ADAN header eye-catcher
		ADANLEN	24	ADAN header length
		ADANTYPE	D	Type. Valid values are: C: ADAC header D: ADAC header and payload P: ADAH record segment R: Logical record segment
		ADANIND	E	Continuation indicator. Valid values are: C: Continuation record (ADAN) to follow E: End of logical record (last segment)
		Reserved	0	Reserved
		ADANSLEN	10000	Segment length (length of payload data following the header)
		ADANOFF	64	Error offset in logical record.
	ADANDATA	ADAC and payload data	Continued rejected input data	

ADACMP Report

ADACMP calculates the approximate amount of space (in both blocks and cylinders) required for Data Storage for the compressed records. This information is printed as a matrix which contains the required space for the different device types requested by the DEVICE parameter for various Data Storage padding factors between 5 and 30 percent.

The following is an example of ADACMP report output:

PARAMETERS:

```
ADACMP COMPRESS NUMREC=1000
ADACMP FNDEF='01,AA,8,B,DE'
ADACMP FNDEF='01,BA,6,A,NU'
ADACMP FNDEF='01,BB,8,P,NU'
ADACMP FNDEF='01,AD,1,A,FI'
ADACMP SUBDE='CA=BA(1,3)'
```

COMPRESS PROCESSING STATISTICS:

```
NUMBER OF RECORDS READ          1,000
NUMBER OF INCORRECT RECORDS      0
NUMBER OF COMPRESSED RECORDS    1,000

RAW DATA                        24,000 BYTES
COMPRESSED DATA                 16,656 BYTES
COMPRESSION RATE                 31.9 %
LARGEST COMPRESSED RECORD        20 BYTES
```

DATASTORAGE SPACE REQUIREMENTS:

I	DEVICE	I	PADDING	I	BLOCKSIZE	I	NUMBER OF		I
I		I	FACTOR	I	BYTES	I	BLOCKS	CYLS	I
I	3380	I		I	4,820	I			I
I		I	5%	I	4,578	I	4	1	I
I		I	10%	I	4,337	I	4	1	I
I		I	15%	I	4,096	I	5	1	I
I		I	20%	I	3,856	I	5	1	I
I		I	25%	I	3,615	I	5	1	I
I		I	30%	I	3,373	I	5	1	I
I		I		I		I			I

TEMP SPACE ESTIMATION:

I	DEVICE	I	BLOCKSIZE	I	NUMBER OF		I
I		I	BYTES	I	BLOCKS	CYLS	I
I	3380	I	7,476	I	5	1	I

THE LARGEST DESCRIPTOR IS AA, IT WILL OCCUPY 1 TEMP BLOCKS

SORT SPACE ESTIMATION:

I	DEVICE	I	BLOCKSIZE	I	LWP	I	NR OF		I
I		I	(BYTES)	I	(BYTES)	I	BLOCKS	CYLS	I

```

-----
I  3380  I    7476  I  139264 (MINIMUM) I    2    1 I
I      I      I  1048576 (DEFAULT) I    2    1 I
I      I      I   139264 (OPTIMUM) I    2    1 I
I-----I-----I-----I-----I-----I

```

The compression rate is computed based on the real amount of data used as input to the compression routine. Fields skipped by a format element "nX" (used to fill a field with blanks) are not counted.

If SPAN was specified in an ADACMP COMPRESS run, statistics about the spanned Data Storage records are also printed:

Spanned Record Statistics:

```

-----
Number of Non-Spanned records          2
Number of Spanned records              8
Min Number of Segments                 2
Max Number of Segments                 5
Avg Number of Segments                 4

Max MU Count                           0
Max PE Count                           300

```

If large object (LB) fields are compressed, statistics about the LB fields (listed as "LOBs" in the report) are printed:

Large Objects Statistics:

```

-----
Number of bytes for largest LOB        10,124,996
Total number of LOBs                   5
Total number of outsourced LOBs        5
Number of LOB file records             10,085

```