

# LF Command: Read Field Definitions

The LF command reads the characteristics of all fields in a file.

We recommend that you set unused ACB and ACBX fields to binary zeros before the direct call is initiated.

This chapter covers the following topics:

- Function and Use
  - ACB Interface Direct Call: LF Command
  - ACBX Interface Direct Call: LF Command
  - Buffers
- 

## Function and Use

The LF command is used to read the field definition information for a file. This command is used primarily by Adabas subsystems; it is normally not used by an application program.

The user specifies the file number for which the field definitions are to be returned.

Adabas provides the field information in the record buffer in one of four formats, according to the setting of the Command Option 2 field. The timestamp when the field definitions were created or last updated is returned in the record buffer with the Command Option 2 field is set to X or F. If a file is stored with ADAORD or restored with ADASAV, the timestamp of the store or restore is not stored; the original timestamp of the file when it was reordered or saved is kept.

## ACB Interface Direct Call: LF Command

This section describes ACB interface direct calls for the LF command. It covers the following topics:

- Control Block and Buffer Overview
- Control Block Field Descriptions
- ACB Example

### Control Block and Buffer Overview

#### Control Block

Field	Position	Format	Before Adabas Call	After Adabas Call
	1-2	--	--	--
Command Code	3-4	alphanumeric	F	U
	5-8	--	--	--
File Number	9-10	binary	F	U
Response Code	11-12	binary	--	A
	13-26	--	--	--
Record Buffer Length	27-28	binary	F	U
	29-35	--	--	--
Command Option 2	36	alphanumeric	F	U
	37-48	--	--	--
Additions 3	49-56	alphanumeric	F	A
	57-72	--	--	--
Command Time	73-76	binary	--	A
User Area	77-80	--	--	U

### Buffer Areas

Buffer	Before Adabas Call	After Adabas Call
Format	*	--
Record	--	A

where:

- F Supplied by user before Adabas call
- A Supplied by Adabas
- U Unchanged after Adabas call
- \* Not used but must be included in parameter list of call statement
- Not used

### Control Block Field Descriptions

We recommend that you set unused ACB fields to binary zeros before the direct call is initiated.

#### Command Code (ACBCMD)

LF

**File Number (ACBFNR)**

The number of the file for which the field definition information is to be returned.

**Note:**

When using two-byte file numbers and database IDs, a X'30' must be coded in the first byte of the control block.

**Response Code (ACBRSP)**

Adabas returns the response code for the command in this field. Response code 0 (ADARSP000) indicates that the command was executed successfully. If the LF command returns a nonzero response code, the rightmost two bytes of Adabas control block bytes 45-48 (Additions 2 field) may contain a subcode defining the exact response code meaning. Response codes and their subcodes are defined in the *Adabas Messages and Codes Manual* documentation.

**Record Buffer Length (ACBRBL)**

The record buffer length (in bytes). The length specified must be large enough to contain all field definition information for the file, but not larger than the size of the record buffer area defined in the user program. If you specify the internal format (I) Command Option, the maximum possible output is four Associator blocks.

**Command Option 2: Type of Information to Be Displayed (ACBCOP2)**

The setting of the Command Option 2 field determines the format and type of field information to be returned in the record buffer.

Option	Description
F	<p>This option delivers the same information as the "X" option (below), including information on logically deleted fields and descriptors.</p> <p>Logically deleted fields will be returned with a specific indicator. All entries with logically deleted descriptors will be returned with a specific indicator.</p>
I	Returns all field information in Adabas internal format.
S	<p>Returns all field information, including collation descriptor, subfield and superfield, subdescriptor, superdescriptor, hyperdescriptor, and phonetic descriptor information.</p> <p>However, this information may not be complete; in particular, new features introduced beginning with Adabas Version 8.2 are not included.</p> <p><b>Note:</b> Very large FDTs cannot be read with the ACB interface if the required space exceeds the record buffer size limit of 32 KB (minus 1 byte) length.</p>
X	<p>Returns extended field information, including collation descriptor, subfield, superfield, subdescriptor, superdescriptor, hyperdescriptor, and phonetic descriptor information with extended date and time formats.</p> <p>For logically deleted fields, no entry will be returned. For logically deleted descriptors on fields, subdescriptors, and superdescriptors, an entry will be returned with the descriptor options reset. For collation descriptors, hyperdescriptors, and phonetic descriptors, no entry will be returned.</p> <p><b>Note:</b> Very large FDTs cannot be read with the ACB interface if the required space exceeds the record buffer size limit of 32 KB (minus 1 byte) length.</p>

If this field is left blank or contains binary zero, the LF command returns field information *excluding* sub-/super-/hyper-/phonetic or collation descriptor information. This is the same format as provided in Adabas version 4.

### Additions 3: Password (ACBADD3)

This field is used to provide a security password. If the file to be used is not security-protected, this field should be set to blanks. If the file is security-protected, the user must provide a valid password.

If the accessed file is password-protected, Adabas replaces the password with blanks during command processing to protect password integrity.

## ACB Example

The field definition information for file 1 is to be read.

### Control Block

<b>Command Code</b>	LF	
<b>File Number</b>	1	field definitions requested for file 1
<b>Record Buffer Length</b>	100	
<b>Command Option 2</b>	S	information for all types of descriptors and sub/superfields is to be returned
<b>Additions 3</b>	<i>bbbbbbb</i> (blanks)	file is not security-protected

## ACBX Interface Direct Call: LF Command

This section describes ACBX interface direct calls for the LF command. It covers the following topics:

- Control Block and Buffer Overview
- Control Block Field Descriptions

### Control Block and Buffer Overview

#### Control Block

Field	Position	Format	Before Adabas Call	After Adabas Call
	1-2	---	---	---
Version Indicator	3-4	binary	F	U
	5-6	---	---	---
Command Code	7-8	alphanumeric	F	U
	9-10	---	---	---
Response Code	11-12	binary	---	A
	13-16	---	---	---
Database ID	17-20	numeric	F	U
File Number	21-24	numeric	F	U
	25-49	---	---	---
Command Option 2	50	alphanumeric	F	U
	51-68	---	---	---
Additions 3	69-76	alphanumeric/ binary	F	A
	77-114	---	---	---
Error Subcode	115-116	binary	---	A
	117-144	---	---	---
Command Time	145-152	binary	---	A
User Area	153-168	not applicable	---	U
---	169-193	do not touch	---	---

### ABDs and Buffers

ABD and Buffer	Before Adabas Call	After Adabas Call
Format	*	--
Record	--	A

where:

- F      Supplied by user before Adabas call
- A      Supplied by Adabas
- U      Unchanged after Adabas call
- \*      Not used but should be included in Adabas call or one will be automatically generated.
- Not used

## Control Block Field Descriptions

We recommend that you set unused ACBX fields to binary zeros before the direct call is initiated.

### Version Indicator (ACBXVER)

F2

### Command Code (ACBXCMD)

LF

### Response Code (ACBXRSP)

Adabas returns the response code for the command in this field. Response code 0 (ADARSP000) indicates that the command was executed successfully. Non-zero response codes, which can also have accompanying subcodes returned in the Error Subcode (ACBXERRC) field, are described in the *Adabas Messages and Codes Manual* documentation.

### Database ID (ACBXDBID)

Use this field to specify the database ID. The Adabas call will be directed to this database.

This field is a four-byte binary field, but at this time only two-byte database IDs are supported. Therefore, the database ID should be specified in the low-order part (rightmost bytes) of the field, with leading binary zeros.

If this field is set to binary zeros, the Adabas API uses either the database ID from the ADARUN cards provided in DDCARD input data or the default database ID value provided in the LNKGBLS module linked with or loaded by the link routine.

### File Number (ACBXFNR)

Use this field to specify the number of the file for which the field definition information is to be returned.

This field is a four-byte binary field, but the file number should be specified in the low-order part (rightmost bytes) of the field, with leading binary zeros.

### Command Option 2: Type of Information to Be Displayed (ACBXCOP2)

The setting of the Command Option 2 field determines the format and type of field information to be returned in the record buffer.

Option	Description
F	<p>This option delivers the same information as the "X" option (below), including information on logically deleted fields and descriptors.</p> <p>Logically deleted fields will be returned with a specific indicator. All entries with logically deleted descriptors will be returned with a specific indicator.</p>
I	Returns all field information in Adabas internal format.
S	<p>Returns all field information, including collation descriptor, subfield and superfield, subdescriptor, superdescriptor, hyperdescriptor, and phonetic descriptor information.</p> <p><b>Note:</b> Beginning with Adabas 8.2, new field options (e.g. date-time and system fields) are not returned when Command Option 2 is set to "S", due to the fixed length of the field information output element. We recommend you set Command Option 2 to "X" instead.</p>
X	<p>Returns extended field information, including collation descriptor, subfield, superfield, subdescriptor, superdescriptor, hyperdescriptor, and phonetic descriptor information with extended date and time formats.</p> <p>For logically deleted fields, no entry will be returned. For logically deleted descriptors on fields, subdescriptors, and superdescriptors, an entry will be returned without descriptor options set. For collation descriptors, hyperdescriptors, and phonetic descriptors, no entry will be returned.</p>

If this field is left blank or contains binary zero, the LF command returns field information *excluding* sub-/super-/hyper-/phonetic or collation descriptor information. This is the same format as provided in Adabas version 4.

### Additions 3: Password (ACBXADD3)

This field is used to provide a security password. If the file to be used is not security-protected, this field should be set to blanks. If the file is security-protected, the user must provide a valid password.

If the accessed file is password-protected, Adabas replaces the password with blanks during command processing to protect password integrity.

### Error Subcode (ACBXERRC)

If the command returns a nonzero response code, this field contains a subcode defining the exact response code meaning. Response codes and their subcodes are defined in the *Adabas Messages and Codes Manual* documentation.

## Buffers

A format buffer is not used by the LF command, but should be included in the Adabas call.

All field definition information is returned in the record buffer, in one of four formats, depending on the setting of the Command Option 2 field.

This section covers the following topics:

- Format Buffer
- Record Buffer When Command Option 2=S
- Record Buffer When Command Option 2=F or X
- Record Buffer When Command Option 2=I
- Record Buffer When Command Option 2 Not Set

### Format Buffer

A format buffer is not used by the LF command, but should be included in the Adabas call. If this is an ACB interface direct call and a format buffer is not specified, a processing error will occur; ACB interface direct calls expect buffers to be specified in a set sequence. If this is an ACBX interface direct call and a format buffer is not specified, one will be automatically generated.

### Record Buffer When Command Option 2=S

If the Command Option 2 field is set to "S", all field information, including collation descriptor, subdescriptor, superdescriptor, hyperdescriptor, and phonetic descriptor and sub-/superfield information is returned in the following format:

**Note:**

If you are using Adabas 8, and an LF command with Command Option 2 set to "S" is run, and large object (LB) fields are encountered, the LB field description is returned in an F-type FDT field definition. Bit 6 in the second format byte (at offset 7 or byte 8 in the element) is set to indicate that the LB (large object) option is set for the field. In addition, bit 1 of the second format byte indicates whether the LB field is defined with the NB (no blank compression) option.

Bytes	Usage
1-2	total length of information
3-4	number of fields in the FDT (including SDTs, as described below)
5-N	field entries; each entry is 8 bytes, maximum number of entries is 3214
(N+1) - M	special descriptor table (SDT) including <ul style="list-style-type: none"> <li>● sub-/superdescriptors (or sub-/superfields)</li> <li>● phonetic descriptors</li> <li>● hyperdescriptors</li> <li>● collation descriptors</li> </ul> The length of a sub-/phonetic/collation descriptor element is eight bytes. Superdescriptor or hyperdescriptor elements are two or more 8-byte entries long.

The following sections provide details of this format:

- FDT Field Entries
- SDT Field Definitions
- Collation Descriptor Entries
- Hyperdescriptor Entries
- Phonetic Descriptor Entries
- Subfield/Subdescriptor Entries
- Superdescriptor/Superfield Entries

### FDT Field Entries

The syntax of FDT field entry is:

```
'F' field-name option level length format
```

The following table describes this syntax:

Notation	Bytes	Usage
F	1	"F" indicates an FDT field entry
<i>field-name</i>	2-3	field name

<b>Notation</b>	<b>Bytes</b>	<b>Usage</b>
<i>option</i>	4	definition options: <ul style="list-style-type: none"><li>● bit 1=1: descriptor</li><li>● bit 2=1: fixed length</li><li>● bit 3=1: multiple-value field</li><li>● bit 4=1: null suppression</li><li>● bit 5=1: periodic group field</li><li>● bit 6=1: parent of phonetic descriptor</li><li>● bit 7=1: parent of subdescriptor or superdescriptor</li><li>● bit 8=1: unique descriptor</li></ul>
<i>level</i>	5	level number (in binary)
<i>length</i>	6	length

Notation	Bytes	Usage
<i>format</i>	7	type of data: <ul style="list-style-type: none"> <li>● A -- alphanumeric</li> <li>● B -- binary</li> <li>● F -- fixed point</li> <li>● G -- floating point</li> <li>● P -- packed decimal</li> <li>● U -- unpaced decimal</li> <li>● W -- wide-character</li> </ul>
	8	options (continued): <ul style="list-style-type: none"> <li>● bit 1=1: NB (no blank compression) option</li> <li>● bit 2=1: NV (not converted) option</li> <li>● bit 3=1: unused</li> <li>● bit 4=1: XI (exclude PE occurrence number from UQ) option</li> <li>● bit 5=1: LA (long alpha) option</li> <li>● bit 6=1: LB (large object) option</li> <li>● bit 7=1: NN option</li> <li>● bit 8=1: NC option</li> </ul>

**Note:**

A field within a periodic group has the following characteristics: - an option field (byte 4) with bit 5=1; and - a level field (byte 5) with level number *greater than* 1. The periodic group field itself always has option bit 5=1 and a level number of 1.

**SDT Field Definitions**

*x* *SDT-definition*

where *x* is one of the following:

Value of <i>x</i>	Description
C	Collation descriptor; see the section <i>Collation Descriptor Entries</i>
H	Hyperdescriptor; see the section <i>Hyperdescriptor Entries</i>
P	Phonetic descriptor; see the section <i>Phonetic Descriptor Entries</i>
S	Subfield or subdescriptor; see the section <i>Subfield/Subdescriptor Entries</i>
T	Superfield or superdescriptor; see the section <i>Superdescriptor/Superfield Entries</i>
X'00'	Element continuation

### Collation Descriptor Entries

```
'C' name option exit length p-field-name
```

Notation	Bytes	Usage
'C'	1	'C' indicates collation descriptor
<i>name</i>	2-3	collation descriptor name
<i>option</i>	4	definition options: bit 1=1     descriptor bit 2=1     exclude PE occurrence number from UQ bit 3=1     multiple-value format bit 4=1     null-value suppression bit 5=1     periodic-group field bits 6-7    (unused) bit 8=1     unique descriptor
<i>exit</i>	5	collation exit number (binary, values 1-8 permitted)
<i>length</i>	6	length
<i>p-field-name</i>	7-8	parent-field name

### Hyperdescriptor Entries

```
'H' name option exit length format X'00'  
X'00' X'00' p-fieldname-list ...
```

Notation	Bytes	Usage
'H'	1	'H' indicates a hyperdescriptor definition
<i>name</i>	2-3	hyperdescriptor name
<i>option</i>	4	definition options: bit 1 (unused) bit 2=1 fixed length bit 3=1 multiple value bit 4=1 null-value suppression bit 5=1 periodic group bit 6-7 (unused) bit 8=1 unique descriptor
<i>level</i>	5	hyperdescriptor exit number (binary; values 1-31 permitted)
<i>length</i>	6	length
<i>format</i>	7	format: A alphanumeric B binary F fixed point G floating point P packed decimal U unpacked decimal
X'00'	8	options (continued) bits 1-3 (unused) bit 4=1 XI (exclude PE occurrence number from UQ) option bits 5-8 (unused)

A hyperdescriptor parent-field name list is an extension of a hyperdescriptor definition. It has the following format for all eight-byte groups after the first:

Notation	Bytes	Explanation
X'00'	1	X'00' indicates continuation
X'00'	2	(unused)
<i>p-fieldname-list...</i>	3-8	parent-field name list: each name is two bytes; six bytes total (that is, three names. If fewer than three names are provided, the additional bytes are filled with X'00').

### Phonetic Descriptor Entries

```
'P' desc-name option p-field-name X'0000'
```

Notation	Bytes	Explanation
'P'	1	'P' indicates phonetic descriptor
<i>desc-name</i>	2-3	phonetic descriptor name
<i>option</i>	4	(unused)
<i>p-field-name</i>	5-6	parent-field name
X'0000'	7-8	(unused; set to nulls)

### Subfield/Subdescriptor Entries

```
'S' s-name option p-field-name from to
```



Notation	Bytes	Usage
'T'	1	'T' indicates superdescriptor/superfield
<i>sup-name</i>	2-3	superdescriptor name
<i>option</i>	4	definition options: bit 1=1            descriptor bit 2=1            exclude PE occurrence number from UQ bit 3=1            multiple-value format bit 4=1            null-value suppression bit 5=1            periodic-group field bit 6-7            (unused) bit 8=1            unique descriptor
<i>p-field-name</i>	5-6	parent-field name
<i>from</i>	7	starting (inclusive) byte
<i>to</i>	8	ending (inclusive) byte

Extension of a superdescriptor or superfield definition has the following format on all eight-byte groups after the first:

Notation	Bytes	Explanation
X'00'	1	indicates continuation
X'000000'	2-4	(unused)
<i>p-field-name</i>	5-6	parent-field name
<i>from</i>	7	starting (inclusive) byte
<i>to</i>	8	ending (inclusive) byte

## Record Buffer When Command Option 2=F or X

If the Command Option 2 field is set to "F" or "X", extended field information, including collation descriptor, subdescriptor, superdescriptor, hyperdescriptor, and phonetic descriptor and sub-/superfield information is returned in the following format:

The "F" option delivers the same information as the "X" option, including information on logically deleted fields and descriptors. Logically deleted fields will be returned with a specific indicator; entries with logically deleted descriptors will be returned with the specific indicator.

Bytes	Usage
1-4	Total length of information
5	Structure level
6	Flag byte for future use
7-8	Number of field definition entries in the FDT (including SDTs, as described below). The maximum number of entries is 3214.
9-16	UNIX timestamp (the number of microseconds since 1970)
17-N	FDT Field entries: each entry is 16 bytes long. Future versions of Adabas may introduce additional, larger entries. Therefore, unknown entry types should not be considered in error.
(N+1) - M	Special descriptor table (SDT) entries including: <ul style="list-style-type: none"> <li>● sub-/superdescriptors (or sub-/superfields)</li> <li>● phonetic descriptors</li> <li>● hyperdescriptors</li> <li>● collation descriptors</li> </ul> <p>For SDT entries, an SDT element has an integral length and is a multiple of 8 bytes.</p>
(M+1) - L	Referential integrity constraint table (RIT). This table is not applicable to mainframe systems; it is used in Adabas open systems environments only.

The following sections provide format details of the field and descriptor entries included in the record buffer:

- FDT Field Entries
- SDT Field Entries
- Collation Descriptor Entries
- Hyperdescriptor Entries
- Phonetic Descriptor Entries
- Subfield/Subdescriptor Entries
- Superdescriptor/Superfield Entries

These entries may be larger with each Adabas version. Therefore, you should always use the entry field length (in byte 2) to skip to the next entry. The length of each entry is aligned to four bytes so you can access 4-byte integer values without alignment problems.

### FDT Field Entries

The format of FDT field entries varies, depending on whether any of the following conditions are met:

- The DT (date/time) field option is not set.
- The SY (system-generated) option is not set. This option is currently only applicable to Adabas in open systems environments.
- The length of the FDT field entry is less than 255 bytes.

If any of these conditions are met, the FDT field entry format matches that produced when the Command Option 2 field is set to "S", as described in *FDT Field Entries*.

The following table describes the FDT field entry format when Command Option 2 is set to "F" or "X" and if the DT or SY options are specified or if the length of the FDT field entry is greater than 255 bytes.

Bytes	Usage
1	"F" indicates an FDT field definition
2	Total length of FDT field entry
3-4	Field name
5	Field format: <ul style="list-style-type: none"> <li>● A: alphanumeric</li> <li>● B: binary</li> <li>● F: fixed point</li> <li>● G: floating point</li> <li>● P: packed decimal</li> <li>● U: unpaced decimal</li> <li>● W: wide-character</li> </ul>
6	Definition options: <ul style="list-style-type: none"> <li>● bit 1=1: descriptor</li> <li>● bit 2=1: fixed length</li> <li>● bit 3=1: multiple-value field</li> <li>● bit 4=1: null suppression</li> <li>● bit 5=1: periodic group field</li> <li>● bit 6=1: parent of phonetic descriptor</li> <li>● bit 7=1: parent of subdescriptor or superdescriptor</li> <li>● bit 8=1: unique descriptor</li> </ul>

Bytes	Usage
7	Additional options: <ul style="list-style-type: none"> <li>● bit 1=1: NB (no blank compression) option</li> <li>● bit 2=1: NV (not converted) option</li> <li>● bit 3=1: unused</li> <li>● bit 4=1: XI (exclude PE occurrence number from UQ) option</li> <li>● bit 5=1: LA (long alpha) option</li> <li>● bit 6=1: LB (large object) option</li> <li>● bit 7=1: NN option</li> <li>● bit 8=1: NC option</li> </ul>
8	Level number (in binary)
9	Date/time edit masks: <ul style="list-style-type: none"> <li>● 1: E(DATE)</li> <li>● 2: E(TIME)</li> <li>● 3: E(DATETIME)</li> <li>● 4: E(TIMESTAMP)</li> <li>● 5: E(NATDATE)</li> <li>● 6: E(NATTIME)</li> <li>● 7: E(UNIXTIME)</li> <li>● 8: E(XTIMESTAMP)</li> </ul>
10	For the datetime/edit masks, the following additional information is supplied: <ul style="list-style-type: none"> <li>● bit 0x01: TZ (timezone) option</li> </ul> For the SY option, the following additional information is supplied: <ul style="list-style-type: none"> <li>● bit 0x40: CR option in use</li> </ul>

Bytes	Usage
11	SY function options (not applicable on mainframe systems) or the user exit number: <ul style="list-style-type: none"> <li>● 1: TIME</li> <li>● 2: SESSIONID</li> <li>● 3: OPUSER</li> <li>● 4: SESSIONUSER</li> <li>● 5: JOBNAME (mainframe only)</li> </ul>
12	Additional field status information * : <ul style="list-style-type: none"> <li>● bit 0x01: Field logically deleted</li> <li>● bit 0x02: Descriptor logically deleted (mainframe only)</li> </ul> <p>*: This information will only be given when Command Option 2 is set to "F"; other descriptor-related flags also remain set. When Command Option 2 is set to "X", the field entry for the logically deleted field will be suppressed entirely; descriptor-related flags for logically deleted descriptors will be reset.</p>
13-16	Field length

**Note:**

A field within a periodic group has the following characteristics: - an option field (byte 6) with bit 5=1; and - a level field (byte 8) with level number *greater than* 1. The periodic group field itself always has option bit 5=1 and a level number of 1.

**SDT Field Entries**

<i>x</i> <i>SDT-definition</i>
--------------------------------

where *x* is one of the following:

Value of <i>x</i>	Description
C	Collation descriptor entry; see the section <i>Collation Descriptor Entries</i>
H	Hyperdescriptor entry; see the section <i>Hyperdescriptor Entries</i>
P	Phonetic descriptor entry; see the section <i>Phonetic Descriptor Entries</i>
S	Subfield or subdescriptor entry; see the section <i>Subfield/Subdescriptor Entries</i>
T	Superfield or superdescriptor entry; see the section <i>Superdescriptor/Superfield Entries</i>
X'00'	Element continuation

## Collation Descriptor Entries

The following table describes the format of collation descriptor entries:

Bytes	Usage
1	"C" indicates a collation descriptor
2	Total length of collation descriptor entry.
3-4	Collation descriptor name
5	Parent field format (only "W" is possible in Adabas open systems environments; both "A" and "W" are possible in Adabas mainframe environments).
6	Definition options: <ul style="list-style-type: none"> <li>● bit 1=1: descriptor</li> <li>● bit 2=1: XI (exclude PE occurrence number from UQ) option</li> <li>● bit 3=1: multiple-value field</li> <li>● bit 4=1: null suppression</li> <li>● bit 5=1: periodic group field</li> <li>● bit 6=1: parent of phonetic descriptor</li> <li>● bit 7=1: parent of subdescriptor or superdescriptor</li> <li>● bit 8=1: unique descriptor</li> </ul>
7-8	Standard length
9-10	Parent field name
11-12	Maximum internal length

Bytes	Usage
13	<p>Additional options:</p> <ul style="list-style-type: none"> <li>● 0x01: NC (open systems)</li> <li>● 0x02: Descriptor logically deleted (mainframe systems only)*</li> <li>● 0x04: LA (open systems only)</li> <li>● 0x08: LB (open systems only)</li> <li>● 0x80: Collation defined via collation exit</li> </ul> <p>When this bit is not set, the collation is defined via ICU</p> <ul style="list-style-type: none"> <li>● Other bits are unused</li> </ul> <p>* : This information will only be given when Command Option 2 is set to "F"; descriptor-related flags also remain set. When Command Option 2 is set to "X", the entry for the logically deleted descriptor will be suppressed entirely.</p>
14	String length of the collation attribute string (length byte and termination null character not included)
15-(14+byte14)	<p>Collation attribute string as null terminated string. For example: " 'de' , PRIMARY"</p> <p>If the collation is defined via the collation exit, the exit number as a null terminated string is placed here. For example: " 1 ".</p>

**Notes:**

1. The collation descriptor entry length is aligned to four bytes.
2. The format of the parent field is relevant if you specify a parent field value in the value buffer. In Adabas on open systems, you can also specify the internal collation descriptor values if the collation descriptor is defined without the HE option. These values are values of format A with option NV.

**Hyperdescriptor Entries**

The following table describes the format of hyperdescriptor entries:

Bytes	Usage
1	"H" indicates a hyperdescriptor definition
2	Total length of the hyperdescriptor entry
3-4	Hyperdescriptor name
5	Field format
6	Definition options: <ul style="list-style-type: none"> <li>● bit 1=1: descriptor</li> <li>● bit 3=1: multiple-value field</li> <li>● bit 4=1: null suppression</li> <li>● bit 5=1: periodic group field</li> <li>● bit 6=1: parent of phonetic descriptor</li> <li>● bit 7=1: parent of subdescriptor or superdescriptor</li> <li>● bit 8=1: unique descriptor</li> </ul>
7-8	Hyperdescriptor length
9	Exit number
10	Additional field status information: * <ul style="list-style-type: none"> <li>● bit 0x02: Descriptor logically deleted (mainframe only)</li> </ul> <p>*: This information will only be given when Command Option 2 is set to "F"; descriptor-related flags remain set. When Command Option 2 is set to "X", the field entry for the logically deleted field will be suppressed entirely.</p>
11	Unused
12	Number of parent fields
13-(12+2* byte-12)	Parent field names

**Note:**

The hyperdescriptor entry length is aligned to four bytes.

**Phonetic Descriptor Entries**

The following table describes the format of phonetic descriptor entries:

Bytes	Usage
1	"P " identifies a phonetic descriptor
2	Total length of phonetic descriptor entry
3-4	Phonetic descriptor name
5	Field format (currently only "A"-format supported)
6	Additional field status information: * *: This information will only be given when Command Option 2 is set to "F". When Command Option 2 is set to "X", the field entry for the logically deleted field will be suppressed entirely.
7-8	Phonetic descriptor length
9-10	unused
11-12	Phonetic descriptor parent field name

### Subfield/Subdescriptor Entries

The following table describes the format of subfield or subdescriptor entries:

Bytes	Usage
1	"S" identifies a subdescriptor or subfield
2	Total length of the subfield or subdescriptor entry.
3-4	Subdescriptor or subfield name
5	Field format
6	Definition options: <ul style="list-style-type: none"> <li>● bit 1=1: descriptor</li> <li>● bit 2=1: XI (exclude PE occurrence number from UQ) option</li> <li>● bit 3=1: multiple-value field</li> <li>● bit 4=1: null suppression</li> <li>● bit 5=1: periodic group field</li> <li>● bit 6=1: parent of phonetic descriptor</li> <li>● bit 7=1: parent of subdescriptor or superdescriptor</li> <li>● bit 8=1: unique descriptor</li> </ul>
7-8	Descriptor length
9	Additional field status information: * <ul style="list-style-type: none"> <li>● bit 0x02: Descriptor logically deleted (mainframe systems only)</li> </ul> <p>*: This information will only be given when Command Option 2 is set to "F"; descriptor-related flags also remain set. When Command Option 2 is set to "X", the entry for the logically deleted descriptor will appear with all descriptor related flags reset.</p>
10	Number of parent fields ("1" for every subfield)
11-(10+ 6 * <i>byte10</i> )	Parent field entries. For each parent field, the following format is used: <ul style="list-style-type: none"> <li>● Parent field name (2 bytes)</li> <li>● From byte (2 bytes)</li> <li>● To byte (2 bytes)</li> </ul>

**Note:**

The subdescriptor entry length is aligned to four bytes.

**Superdescriptor/Superfield Entries**

The following table describes the format of superfield or superdescriptor entries:

Bytes	Usage
1	"T" identifies a superdescriptor or superfield
2	Total length of the superfield or superdescriptor entry.
3-4	Superdescriptor or superfield name
5	Field format
6	Definition options: <ul style="list-style-type: none"> <li>● bit 1=1: descriptor</li> <li>● bit 2=1: XI (exclude PE occurrence number from UQ) option</li> <li>● bit 3=1: multiple-value field</li> <li>● bit 4=1: null suppression</li> <li>● bit 5=1: periodic group field</li> <li>● bit 6=1: parent of phonetic descriptor</li> <li>● bit 7=1: parent of subdescriptor or superdescriptor</li> <li>● bit 8=1: unique descriptor</li> </ul>
7-8	Descriptor length
9	Additional field status information: * <ul style="list-style-type: none"> <li>● bit 0x02: Descriptor logically deleted (mainframe systems only)</li> </ul> <p>*: This information will only be given when Command Option 2 is set to "F"; descriptor-related flags also remain set. When Command Option 2 is set to "X", the entry for the logically deleted descriptor will appear with all descriptor related flags reset.</p>
10	Number of parent fields ("1" for every superdescriptor)
11-(10+ 6 * byte10)	Parent field entries. For each parent field, the following format is used: <ul style="list-style-type: none"> <li>● Parent field name (2 bytes)</li> <li>● From byte (2 bytes)</li> <li>● To byte (2 bytes)</li> </ul>

### Record Buffer When Command Option 2=I

If "I" is set for Command Option 2, field information is returned in Adabas internal format:

Bytes	Contents
1	X'80'
2	B'00000xyz' where xyz are ciphering bits: 1=yes; 0=no x     user y     new z     old
3-4	binary zeros
5-8	total inclusive length of FDT, including field definition table (FDT proper), FDT index, and special descriptor table (SDT) [= p - 4]
9-12	Inclusive length of field definition table (FDT proper) [= n - 8]
13-n	FDT field descriptor elements (20 bytes per element; see the following description)
n+1 to n+4	Inclusive length of FDT index [= m - n]
n+5 to m	FDT index
m+1 to m+4	Inclusive length of special descriptor table (SDT) [= p - m]
m+5 to p	Special descriptor table (SDT)

The format of each FDT field descriptor elements is described in the following table:

Offset	Contents
0	field level
1 - 2	field name
3	special field options
4 - 6	reserved
7	default field length
8	field format
9	descriptor definition options
10	special descriptor parent options
11	periodic group count field
12-13	FDT element chain pointer
14	field security levels
15 - 19	reserved

The meaning of FDT elements is described in the Adabas architecture training information.

The FDT is stored in up to four Associator blocks. Therefore, the maximum possible record buffer length is the length of four Associator blocks.

## Record Buffer When Command Option 2 Not Set

### Note:

Command option 2 may be set to values other than "I" or "S" to support older programs; these values do not support newer features. Software AG recommends "I" or "S" for new programs.

If the Command Option 2 field contains neither "I" nor "S", the field information returned *excludes* collation descriptor and sub-/super-/hyper-/phonetic descriptor information. The information is provided in the same format as provided in Adabas version 4:

*n field-def*

where:

- n* is the number of fields in the file. The number is provided as a four-byte binary number in the first four bytes of the record buffer.
- field-def* is the field definition information for each field within the file. The information for each field is provided in six bytes according to the following format:

Bytes	Usage																
1	level number (binary)																
2 - 3	name (alphanumeric)																
4	standard length (binary)																
5	standard format (alphanumeric):																
	<table> <tr> <td>A</td> <td>alphanumeric</td> </tr> <tr> <td>B</td> <td>binary</td> </tr> <tr> <td>F</td> <td>fixed point</td> </tr> <tr> <td>G</td> <td>floating point</td> </tr> <tr> <td>P</td> <td>packed decimal</td> </tr> <tr> <td>U</td> <td>unpacked decimal</td> </tr> <tr> <td>W</td> <td>wide-character</td> </tr> </table>	A	alphanumeric	B	binary	F	fixed point	G	floating point	P	packed decimal	U	unpacked decimal	W	wide-character		
A	alphanumeric																
B	binary																
F	fixed point																
G	floating point																
P	packed decimal																
U	unpacked decimal																
W	wide-character																
6	definition options:																
	<table> <tr> <td>bit 1=1</td> <td>descriptor</td> </tr> <tr> <td>bit 2=1</td> <td>fixed storage</td> </tr> <tr> <td>bit 3=1</td> <td>multiple-value field</td> </tr> <tr> <td>bit 4=1</td> <td>null-value suppression</td> </tr> <tr> <td>bit 5=1</td> <td>periodic group field</td> </tr> <tr> <td>bit 6=1</td> <td>phonetic source field</td> </tr> <tr> <td>bit 7=1</td> <td>sub-/superdescriptor source field</td> </tr> <tr> <td>bit 8=1</td> <td>unique descriptor</td> </tr> </table>	bit 1=1	descriptor	bit 2=1	fixed storage	bit 3=1	multiple-value field	bit 4=1	null-value suppression	bit 5=1	periodic group field	bit 6=1	phonetic source field	bit 7=1	sub-/superdescriptor source field	bit 8=1	unique descriptor
bit 1=1	descriptor																
bit 2=1	fixed storage																
bit 3=1	multiple-value field																
bit 4=1	null-value suppression																
bit 5=1	periodic group field																
bit 6=1	phonetic source field																
bit 7=1	sub-/superdescriptor source field																
bit 8=1	unique descriptor																

The information for the next field immediately follows the information for the preceding field with no intervening spaces.