A1 Command: Update Record

The A1 command updates records with a hold option.

This chapter covers the following topics:

- Function and Use
- Command: A1
- Control Block
- Buffers
- Additional Considerations
- Examples

Function and Use

It is used to change the value of one or more fields in a record. The record containing the field (or fields) to be updated is identified by the file number in which it is contained and its ISN. The user specifies the fields to be updated in the format buffer and provides the updating values for these fields in the record buffer. Only the fields specified are modified. All other fields in the record remain unchanged.

All necessary updating to the Associator and Data Storage is performed by Adabas.

A hold option is available to place the record in hold status before it is updated.

If the user is operating in multiuser mode, the A1 command will be executed only if the record to be updated is in hold status for the user.

Note:
The A4 command from previous Adabas releases is executed as an A1 command.

Command: A1

User Control Block

Field	Position	Format	Before Adabas Call	After Adabas Call
	1-2	--	--	--
COMMAND CODE	3-4	alphanumeric	F	U
COMMAND ID	5-8	alphanumeric / binary	F	U
FILE NUMBER	9-10	binary	F	U
RESPONSE CODE	11-12	binary	--	A
ISN	13-16	binary	F	U
ISN LOWER LIMIT	17-20	binary	--	A
ISN QUANTITY	21-24	binary	--	A
FORMAT BUFFER LENGTH	25-26	binary	F	U
RECORD BUFFER LENGTH	27-28	binary	F	U
	29-34	--	--	--
COMMAND OPTION 1 / 2	35-36	alphanumeric	F	U
	37-44	--	--	--
ADDITIONS 2	45-48	alphanumeric / binary		A
ADDITIONS 3	49-56	alphanumeric	F	A
ADDITIONS 4	57-64	alphanumeric	F	A
ADDITIONS 5	65-72	alphanumeric	F	--
COMMAND TIME	73-76	binary	--	A
USER AREA	77-80	--	--	U

User Buffer Areas

Buffer	Before Adabas Call	After Adabas Call
FORMAT BUFFER	F	U
RECORD BUFFER	F	U

where:

F	Filled in by user before Adabas Call
A	Filled in by Adabas
U	Unchanged after Adabas call
--	Not used

Control Block

Command Code

A1

Command ID

If a series of records is to be updated by using a series of A1 calls, and the same fields are specified in the format buffer for each call (such as when updating a set of records resulting from a find command), this field should be set to a non-blank, non-zero value. If the A1 command is used in conjunction with an L1/L4, L2/L5, or L3/L6 command, and the same fields within each record are read and updated, the same command ID used for the read command should be used for the A1 calls. In both cases, this reduces the time needed to process each successive A1 call.

If only a single record is to be updated with a single A1 call, or the format buffer is modified between A1 calls, this field should be set to blanks.

The leftmost byte of this field may not be set to hexadecimal ’FF’.

File Number

Specify the binary number of the file to be read in this field. For the physical direct calls, specify the file number as follows:

- For a one-byte file number, enter the file number in the rightmost byte (10); the leftmost byte (9), should be set to binary zero (B’0000 0000’).
- For a two-byte file number, use both bytes (9 and 10) of the field.

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

Adabas returns the response code for the command in this field. Response code 0 indicates that the command was executed successfully. Non-zero response codes, which can also have accompanying subcodes returned in the rightmost half of the additions 2 field, are described in the *Adabas Messages and Codes* documentation.

ISN

The ISN of the record to be updated.

ISN Quantity/Lower Limit

These fields are set to nulls following completion of the A1 operation.

Format Buffer Length

The format buffer length (in bytes). The format buffer area defined in the user program must be as large as (or larger than) the length specified.

Record Buffer Length

The record buffer length (in bytes). The record buffer area defined in the user program must be as large as (or larger than) the length specified.

Command Option 1/2 (Hold Record Option)

Option	Description
H	in either of these fields places the record in hold status before it is updated. If the record is currently being held by another user, the command is placed in wait status until either the record becomes available or the transaction times out-unless option "R" is also specified.
R	if specified, must occur in the command option 1 field and option "H" must occur in the command option 2 field. Option "R" causes Adabas to return response code 145 if the record to be held is not available. The command is *not* placed in wait status.

Additions 2 (Length of Compressed Record)

If the command is processed successfully, the following information is returned in this field:

- If the record buffer contains at least one valid field value, the leftmost two bytes contain the length (in binary form) of the compressed record accessed;
- If the A1 command returns a non-zero response code, the rightmost two bytes may contain a subcode defining the exact response code meaning. Response codes and their subcodes are defined in the *Adabas Messages and Codes* documentation.

Additions 3 (Password)

This field is used to provide an Adabas security or ADAESI password. If the database, file, or fields are security protected, the user must provide a valid security or ADAESI password. Adabas sets the additions 3 field to blanks during command processing to enhance password integrity.

If the accessed file is password protected, Adabas sets this field to blanks during command processing to protect the integrity of the password.

Additions 4 (Cipher Code)

This field is used to provide a cipher code. If the file is ciphered, the user must provide a valid cipher code. If the file is not ciphered, this field should be set to blanks.

Adabas sets any cipher code to blanks during command processing, and returns a version code and database ID in the rightmost (low-order) three bytes of this field. For more information, see the section *Control Block Fields*.

Additions 5 (Format ID, Global Format ID)

This field may be used to provide a separate format ID to identify the internal format buffer used for this command, or to provide a global format ID.

As long as the high order bit of the first byte of the additions 5 field is not set to 1, the value provided in the command ID field will be used as the format ID as well.

If, however, this bit is set to 1, the fifth through eighth bytes of the additions 5 field (additions 5 + 4(4)) is used as the format ID.

If the two high-order (leftmost) bits of the first byte of additions 5 field are set to one (B’11’), the value in all eight bytes of the additions 5 field (additions 5 + 0(8)) is used as a global format ID (that is, the format ID can be used by several users at the same time).

See the section *Command ID, Format ID, Global Format ID* for more information and examples.

Buffers

Format Buffer

The fields to be updated must be specified in this buffer. When performing an A1 command, the format buffer cannot contain any of the following:

- A format selection criteria ("fieldname operator value...");
- An edit mask element;
- A reference to a sub-/superdescriptor field;
- The same field specified more than once (except a multiple-value field);
- An "-N" type of specification for an multiple-value field, or for fields in a periodic group (for example, ABN or AB1-N).

Any of the above in the format buffer cause a nucleus response of 44 for an A1 command.

The syntax and examples of format buffer construction are provided in the section *Adabas Calling Procedure*.

Record Buffer

The values to be used for updating are provided in this buffer according to the length and format as specified in the format buffer.

Additional Considerations

The following additional considerations are applicable for the A1 command:

1. Subdescriptors, superdescriptors, and phonetic descriptors may not be directly updated. To update any of the above, the field(s) used to derive the subdescriptor, superdescriptor, or phonetic descriptor must be updated. All corresponding subdescriptor, superdescriptor, or phonetic descriptor values will

then be updated by Adabas.

2. Theoretically, the maximum record length permitted is 32767 bytes before compression. The actual maximum is limited by block size restrictions. It is also smaller depending on the size of the LU parameter specified for the Adabas session; the maximum is (LU - format buffer length - 108). The maximum record length after compression is equal to the smaller of either the Data Storage block size - 4 bytes, or the Work block size - 110 bytes.

3. A descriptor value cannot be larger than 253 bytes.

4. If a field is updated using a length override that exceeds the standard length (not permitted if the field is defined with the Fixed Storage option), all subsequent references to this field should specify the length that was used. If a subsequent reference uses the standard length, value truncation of an alphanumeric fields, a response code 55 for a numeric field may occur.

5. A multiple-value field or a field within a periodic group may be specified more than once in the format buffer. A multiple-value field may be specified without an index several times, or with different indices. A periodic field must always have different indices.

6. A multiple-value or periodic group count field specified in the format buffer will be ignored by Adabas. The corresponding value in the record buffer will also be ignored. A literal in the format buffer will be ignored by Adabas, and the corresponding positions in the record buffer will also be ignored.

7. If a multiple-value field is updated and the field count must be changed, Adabas updates the multiple-value field count according to the following rules:

   - For a multiple-value field defined with the null suppression (NU) option, the count field is adjusted to reflect the current number of existing non-null values. Null values are completely suppressed.

Field Definition	01,MF,5,A,MU,NU
MF values before update	XXXXX,YYYYY
Format Buffer	MF4
Record Buffer	ZZZZZ
Result after update	XXXXX,YYYYY,ZZZZZ MF count = 3
MF values before update	XXXXX,YYYYY,ZZZZZ
Format Buffer	MF2
Record Buffer	bbbbb (blanks)
Result after update	XXXXX,ZZZZZ MF count = 2
MF values before update	XXXXX,ZZZZZ
Format Buffer	MF1-2
Record Buffer	bbbbbbbbbb (blanks)
Result after update	Values suppressed MF count = 0

- For a multiple-value field defined without the NU option, the count is adjusted to reflect the current number of existing values (including null values).

Field Definition	01,MF,5,A,MU
MF values before update	XXXXX,YYYYY
Format Buffer	MF4
Record Buffer	DDDDD
Result after update	XXXXX,YYYYY,b(blank),DDDDD MF count = 4
MF values before update	XXXXX,YYYYY,ZZZZZ
Format Buffer	MF3
Record Buffer	bbbbb (blanks)
Result after update	XXXXX,YYYYY,b (blank) MF count = 3

A maximum of 191 values is permitted for a multiple-value field.

An exception to the rules stated on the previous page is when the format buffer contains a multiple-value field *without an index specification*. In this case, only the new values specified are used and all other values are deleted regardless of the content of the value.

Field Definition	01,MF,5,A,MU,NU
MF values before update	XXXXX,YYYYY
Format Buffer	MF
Record Buffer	AAAAA
Result after update	AAAAA MF count = 1
MF values before update	XXXXX,YYYYY
Format Buffer	MF1
Record Buffer	AAAAA
Result after update	AAAAA,YYYYY MF count = 2
MF values before update	XXXXX,YYYYY,ZZZZZ
Format Buffer	MF
Record Buffer	bbbbb
Result after update	value suppressed MF count = 0

8. If one or more fields contained in a periodic group are updated, Adabas updates the periodic group count, if necessary, according to the following rule:

   The count is adjusted to reflect the highest occurrence number referenced in the format buffer (provided that this occurrence is higher than the existing highest occurrence number).

Field Definitions	01,GB,PE 02,BA,1,B,DE,NU 02,BB,5,P,NU
GB values before update	GB (1st occurrence) BA = 5 BB = 20 GB (2nd occurrence) BA = 6 BB = 25 GB count = 2
Format Buffer	GB4
Record Buffer	X’08000000500F’
Result after update	GB (1st occurrence) BA = 5 BB = 20 GB (2nd occurrence) BA = 6 BB = 25 GB (3rd occurrence) BA = 0 BB = 0 GB (4th occurrence) BA = 8 BB = 500 GB count = 4
GB values before update	GB (1st occurrence) BA = 5 BB = 20 GB (2nd occurrence) BA = 6 BB = 25 GB count = 2
Format Buffer	GB1
Record Buffer	X’00000000000F’
Result after update	GB (1st occurrence) BA = 0 BB = 0 GB (2nd occurrence) BA = 6 BB = 25 GB count = 2

A maximum of 191 occurrences is permitted for a periodic group.

9. If a field defined with variable length (no standard length) is specified in the format buffer, the corresponding value in the record buffer must be preceded by a one-byte binary length value that includes the length byte itself.

Field Definitions	01,AA,3,A 01,AB,A
Format Buffer	AA,AB
Record Buffer	X’F1F2F306F1F2F3F4F5’

Fields AA and AB are to be updated. The new value for AA is "123". The new value for AB (which is a variable-length field) is "12345".

Examples

For the Adabas file definitions used in all the examples in this section, see *File Definitions Used in Examples*.

Example 1

ISN 4 of file 1 is to be updated with the following values:

Field AA	1234
Field AB	20

Control Block

Command Code	A1	
Command ID	bbbb	only 1 record is to be updated
File Number	1	
ISN	4	
Format Buffer Length	10	or larger
Record Buffer Length	10	or larger
Additions 3	bbbbbbbb	file is not security protected
Additions 4	bbbbbbbb	file is not ciphered

Buffer Areas

Format Buffer	AA,AB,2,U.
Record Buffer	X’F1F2F3F440404040F2F0’

Example 2

A set of records (previously identified with a FIND command) in file 2 are to be updated with the following values:

Field RA	ABCD
Field XB	80
Field XC	0

Control Block

Command Code	A1	
Command ID	ABCD	a non-blank, non-zero command ID is recommended because the same fields in a series of records are being updated
File Number	2	
ISN	n	each ISN resulting from the previous FIND command will be inserted in this field before each call
Format Buffer Length	9	or larger
Record Buffer Length	16	or larger
Additions 3	password	file 2 is security protected
Additions 4	bbbbbbb	file is not ciphered

Buffer Areas

Format Buffer	RA,XB,XC.
Record Buffer	X’C1C2C3C440404040080CF0F0F0F0F0F0’

The A1 call is repeated for each ISN which resulted from the previous find command.