N1 and N2 Commands: Adding Records

The N1 and N2 commands are used to add a new record to a file.

The N1 command adds a new database record with an ISN assigned by Adabas. The N2 command adds a new database record with an ISN assigned by the user.

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: N1 and N2 Commands
- ACBX Interface Direct Call: N1 and N2 Commands
- Buffers

Function and Use

The user specifies the file to which the record is to be added, and the fields for which values are being provided. Any fields not specified will contain a null value in the record added.

Adabas assigns the record an ISN, adds the record to Data Storage, and performs any Associator updating which may be required.

The N2 command is used if the ISN to be assigned to the record is being provided by the user. To keep the ISN assigned to the same record, the unloaded file must be reloaded with the USERISN=YES option.

If the user is an ET logic user and is operating in multiuser mode, the record added is placed in hold status.

ACB Interface Direct Call: N1 and N2 Commands

This section describes ACB interface direct calls for N1 and N2 commands. It covers the following topics:

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

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
Command ID	5-8	alphanumeric	F	U
File Number	9-10	binary	F	U
Response Code	11-12	binary	--	A
ISN	13-16	binary	F	A/U [1]
ISN Lower Limit	17-20	binary	--	A [2]
ISN Quantity	21-24	binary	--	A [2]
Format Buffer Length	25-26	binary	F	U
Record Buffer Length	27-28	binary	F	U
	29-44	--	--	--
Additions 2	45-48	alphanumeric	--	A
Additions 3	49-56	alphanumeric	F	A
Additions 4	57-64	alphanumeric	F	A
Additions 5	65-72	alphanumeric	F	U
Command Time	73-76	binary	--	A
User Area	77-80	--	--	U

Notes

1. Supplied by Adabas for N1; unchanged for N2.
2. These fields are used and not reset by Adabas if coupled files are used.

Buffer Areas

Buffer	Before Adabas Call	After Adabas Call
Format	F	U
Record	F	U

where:

F	Supplied by user before Adabas call
A	Supplied by Adabas
U	Unchanged after Adabas call
--	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)

N1 or N2

Command ID (ACBCID)

If a series of records is being added using a series of N1 or N2 calls, and the same fields are specified in the format buffer for each call, this field should be set to a non-blank, non-zero value. This results in a reduction in the time required to process each N1 or N2 call.

If only a single record is being added, or if the format buffer is modified between N1 or N2 calls, this field should be set to blanks.

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

File Number (ACBFNR)

Specify the binary number of the file to be read in this field. For 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 (ACBRSP)

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 rightmost half of the Additions 2 field, are described in the *Adabas Messages and Codes Manual* documentation.

ISN (ACBISN)

If the N1 command is being executed, Adabas returns the ISN assigned to the record in this field.

If the N2 command is being executed, the ISN to be assigned to the record must be provided in this field. The ISN provided must not already be assigned to a record in the file, and must be within the limit (MAXISN) in effect for the file. MAXISN is set by the DBA when the file is loaded.

Note:
You cannot assign an ISN that is greater than the value specified by the MAXISN parameter for the file.

ISN Quantity/Lower Limit (ACBISQ and ACBISL)

These fields are set to nulls following completion of the N1 or N2 command operation, unless hard-coupled files are used. If coupled files are used, these fields are used by N1 or N2 command processing and are not reset.

Format Buffer Length (ACBFBL)

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 (ACBRBL)

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.

Additions 2: Length of Compressed Record (ACBADD2)

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 newly added compressed record;
- If the N1 or N2 command returns a nonzero 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 Manual* documentation.

Additions 3: Password (ACBADD3)

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

Additions 4: Cipher Code (ACBADD4)

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 (ACBADD5)

Use this field to specify a separate format ID that identifies the internal format buffer used for this command, or to provide a global format ID allowing use of the internal format buffer by all users.

As long as the leftmost bit of the Additions 5 field is set to 0, 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 are 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’), all eight bytes of the Additions 5 field are 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, Format, and Global Format IDs* for more information and examples.

ACB Examples

Example 1

A record is to be added to file 1. The ISN of the record is to be assigned by Adabas. The field values which are to be provided are as follows:

Field	**Value**
AA	ABCD
MF (value 1)	AAA
MF (value 2)	BBB
BA (1st occurrence)	5
BA (2nd occurrence)	6

Control Block

Command Code	N1	
Command ID	*bbbb* (blanks)	only 1 record being added
File Number	1	
Format Buffer Length	15	or larger
Record Buffer Length	16	or larger
Additions 3	*bbbbbbbb* (blanks)	file 1 is not security-protected
Additions 4	*bbbbbbbb* (blanks)	file is not ciphered

Buffer Areas

Format Buffer	AA,MF1-2,BA1-2.
Record Buffer	X’C1C2C3C440404040C1C1C1C2C2C20506’

Example 2

A record is to be added to file 2. The ISN of the record is to be provided by the user. The field values to be provided are as follows:

Field	**Value**
RA	12345678
RB	ABCD

Control Block

Command Code	N2	
Command ID	*bbbb* (blanks)	only 1 record is to be added
File Number	2	
ISN	20	ISN 20 is to be assigned to the record
Format Buffer Length	6	or larger
Record Buffer Length	18	or larger
Additions 3	password	file 2 is security-protected
Additions 4	*bbbbbbbb* (blanks)	file is not ciphered

Buffer Areas

Format Buffer	RA,RB.
Record Buffer	X’F1F2F3F4F5F6F7F8C1C2C3C4404040404040’

ACBX Interface Direct Call: N1 and N2 Commands

This section describes ACBX interface direct calls for N1 and N2 commands. 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
Command ID	13-16	alphanumeric/ binary	F	U
Database ID	17-20	numeric	F	U
File Number	21-24	numeric	F	U
	25-28	---	---	---
ISN	29-32	binary	F	A/U[1]
	33-36	---	---	---
ISN Lower Limit	37-40	binary	---	A[2]
	41-44	---	---	---
ISN Quantity	45-48	binary	---	A[2]
	49-64	---	---	---
Additions 2	65-68	binary	---	A
Additions 3	69-76	alphanumeric/ binary	F	A
Additions 4	77-84	alphanumeric	F	A
Additions 5	85-92	alphanumeric/ binary	F	U
	93-114	---	---	---
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	---	---

Notes

1. Supplied by Adabas for N1; unchanged for N2.
2. These fields are used and not reset by Adabas if coupled files are used.

ABDs and Buffers

ABD and Buffer	Before Adabas Call	After Adabas Call
Format	F	U
Record	F	U

where:

F	Supplied by user before Adabas call
A	Supplied by Adabas
U	Unchanged after Adabas call
---	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)

N1 or N2

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.

Command ID (ACBXCID)

If a series of records is being added using a series of N1 or N2 calls, and the same fields are specified in the format buffer for each call, this field should be set to a non-blank, non-zero value. This results in a reduction in the time required to process each N1 or N2 call.

If only a single record is being added, or if the format buffer is modified between N1 or N2 calls, this field should be set to blanks.

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

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 to which the Adabas call should be directed.

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.

ISN (ACBXISN)

If the N1 command is being executed, Adabas returns the ISN assigned to the record in this field.

If the N2 command is being executed, the ISN to be assigned to the record must be provided in this field. The ISN provided must not already be assigned to a record in the file, and must be within the limit (MAXISN) in effect for the file. MAXISN is set by the DBA when the file is loaded.

Note:
You cannot assign an ISN that is greater than the value specified by the MAXISN parameter for the file.

The ACBXISN field is a four-byte binary field embedded in the eight-byte ACBXISNG field, which is not yet used. Set the high-order part of the ACBXISNG field to binary zeros.

ISN Lower Limit (ACBXISL)

This field is set to nulls following completion of the N1 or N2 command operation, unless hard-coupled files are used. If coupled files are used, this field is used by N1 or N2 command processing and is not reset.

The ACBXISL field is a four-byte binary field embedded in the eight-byte ACBXISLG field, which is not yet used. Set the high-order part of the ACBXISLG field to binary zeros.

ISN Quantity (ACBXISQ)

This field is set to nulls following completion of the N1 or N2 command operation, unless hard-coupled files are used. If coupled files are used, this field is used by N1 or N2 command processing and is not reset.

Additions 2: Length of Compressed Record (ACBXADD2)

If the command is processed successfully and if the record buffer contains at least one valid field value, the leftmost two bytes contain the length (in binary form) of the newly added compressed record;

Additions 3: Password (ACBXADD3)

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

Additions 4: Cipher Code (ACBXADD4)

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 (ACBXADD5)

Use this field to specify a separate format ID that identifies the internal format buffer used for this command, or to provide a global format ID allowing use of the internal format buffer by all users.

As long as the leftmost bit of the Additions 5 field is set to 0, 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 are 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’), all eight bytes of the Additions 5 field are 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, Format, and Global Format IDs* for more information and examples.

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

- Format Buffer
- Record Buffer

Format Buffer

The fields for which values are being provided in the record buffer must be specified in this buffer. When performing an N1 command, the format buffer cannot contain any of the following:

- format selection criteria ("field-name 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 record format specification (for example, ABN or AB1 - N).

Any of the above in the format buffer cause a nucleus response of 44 for an N1 command. Any fields that are not specified in the format buffer will contain a null value in the record being added.

The following rules control the processing of non-NU descriptors which are not specified in the format buffer for an N1 or N2 command:

- Any omitted non-NU descriptor whose definition in the field definition table (FDT) is both the farthest from the beginning of the FDT, and *precedes* the FDT definition of the last field specified in the format buffer *will* have null values entered in the inverted list for the descriptor;
- Any omitted non-NU descriptor whose definition in the field definition table (FDT) is both the farthest from the beginning of the FDT, and *follows* the FDT definition of the last field specified in the format buffer will *not* have null values entered in the inverted list for the descriptor.

Therefore, the format buffer entry should reference either all non-NU descriptors, or at least one field following (in FDT order) all non-NU descriptor fields. This ensures that null values are correctly inserted in the inverted lists for all non-NU descriptors.

For non-NU descriptors that are contained in a periodic group, null values are entered in their inverted lists only for null occurrences that precede the highest occurrence number specified in the format buffer.

The following additional format buffer considerations are applicable for the N1 or N2 command:

1. Subdescriptors, superdescriptors, hyperdescriptors, and phonetic descriptors may not be specified in the format buffer. Adabas automatically creates the correct value for any of the above if a field from which such a descriptor is derived is specified in the format buffer.
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. If a field is specified 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 for alphanumeric fields or a non-zero response code for numeric fields may occur.
4. Only a multiple-value field may be specified more than once in the format buffer.
5. A multiple-value count field, periodic group count field, or literal value specified in the format buffer is ignored by Adabas. The corresponding value in the record buffer is also ignored.

6. If a multiple-value field is specified in the format buffer, Adabas sets the multiple-value field count according to the following rules:

   - For a multiple-value field defined with the NU option, the count field is adjusted to reflect the number of existing nonblank values. Blank values are completely suppressed.

Field definition	01,MF,5,A,MU,NU
Format buffer	MF1-3
Record buffer	XXXXXYYYYYZZZZZ
Result after add	XXXXX,YYYYY,ZZZZZ MF count = 3
Format buffer	MF1-3
Record buffer	XXXXX*bbbbb*ZZZZZ
Result after add	XXXXX,ZZZZZ MF count = 2
Format buffer	MF1-3
Record buffer	*bbbbbbbbbbbbbbb* (blanks)
Result after add	values suppressed MF count = 0

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

Field definition	01,MF,5,A,MU
Format buffer	MF1-3
Record buffer	XXXXXYYYYY*bbbbb*
Result after add	XXXXX,YYYYY,b(blank) MF count = 3
Format buffer	MF1
Record buffer	*bbbbb* (blanks)
Result after add	b (blank) MF count = 1

   Up to 191 values are permitted for a multiple-value field.

7. If a periodic group or a field within a periodic group is specified in the format buffer, Adabas sets the periodic group count equal to the highest occurrence number specified in the format buffer. If the highest occurrence suppresses null values, the count is adjusted accordingly.

Field definitions	01,GB,PE 02,BA,1,B,DE,NU 02,BB,5,P,NU
Format buffer	GB1-2.
Record buffer	X’08000000500F09000000600F’
Result after add	GB (1st occurrence) BA = 8 BB = 500 GB (2nd occurrence) BA = 9 BB = 600 GB count = 2
Format buffer	GB1-2.
Record buffer	X’00000000000F00000000000F’
Result after add	GB (1st occurrence) values suppressed GB (2nd occurrence) values suppressed GB count = 0

Up to 191 occurrences are permitted for a periodic group.

8. 8. 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 1-byte binary number that represents the length of the value (including the length byte).

Field definitions	01,AA,3,A 01,AB,A
Format buffer	AA,AB.
Record buffer	X’F1F2F306F1F2F3F4F5’

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

Record Buffer

The value for each field specified in the format buffer must be provided in the record buffer.

Each value must be provided according to the standard length and format of the field for which the value is being provided, unless a different length and/or format is specified in the format buffer.

If the field is defined as a variable-length field (no standard length), a 1-byte binary field containing the length of the field (including the length byte) must be provided immediately before the value.

If the field for which the value is being provided is defined as a unique descriptor, the value provided must not already exist for the descriptor; otherwise, the command will be rejected.