

ET Command: End Transaction

The ET command ends and saves the current transaction.

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: ET Command
 - ACBX Interface Direct Call: ET Command
 - Buffers
-

Function and Use

The ET command is used to indicate the end of a logical transaction. Each logical transaction should be secured by issuing an ET command. Subsequent restoring or backing out of any later transaction returns the database status to that set by the last successful ET command.

The ET command:

- writes all current data protection information to the Adabas data protection log and Adabas Work for all update commands executed successfully during the transaction. If the current session ends abnormally, Adabas uses this protection information to reapply the updates for this transaction to the Associator and Data Storage during the next session;
- releases all records held by the user during the current transaction. ISN lists which were saved by the user as a result of find commands are not released (an option is also available by which these ISNs can be returned to hold status);
- optionally stores user data in an Adabas system file. This data may be read subsequently with an OP or RE command as part of a program restart procedure;
- returns a unique sequence number for the transaction in the command ID field of this and any following OP or CL commands issued by the same user. This sequence number may be used to identify the last successfully processed transaction for the user.

The successful execution of an ET command guarantees that all the updates performed during the transaction will be applied to the database, regardless of any subsequent user or Adabas session interruption.

If an ET logic user issues an OP command after a system failure or in the middle of a transaction, Adabas automatically issues a BT command.

ACB Interface Direct Call: ET Command

This section describes ACB interface direct calls for the ET command. 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	binary	--	A
File Number****	9-10	binary	F	U
Response Code	11-12	binary	--	A
	13-16	--	--	--
ISN Lower Limit	17-20	binary	F	U
ISN Quantity	21-24	binary	--	A
	25-26	--	--	
Record Buffer Length	27-28	binary	F *	U
	29-32	--	--	--
ISN Buffer Length **	33-34	binary	F **	U
Command Option 1	35	alphanumeric	F	U
Command Option 2	36	alphanumeric	F	U
	37-72	--	--	--
Command Time	73-76	binary	--	A
User Area	77-80	--	--	U

Buffer Areas

Buffer	Before Adabas Call	After Adabas Call
Format *	***	--
Record *	F	U
ISN **	F	U

where:

- F Supplied by user before Adabas call
- A Supplied by Adabas
- U Unchanged after Adabas call
- * Required only if ET data is to be stored
- ** Required for hold ISN option; optional for multifetch option
- *** Not used but must be included in parameter list of call statement if ET data is to be stored
- **** A database ID is only necessary if you are accessing a database other than the application's default database (read in by ADARUN DBID parameter, provided in the loaded link globals table, or linked with the link routine).
- 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)

ET

Command ID (ACBCID)

Adabas returns either binary zeros or the transaction sequence number in this field.

If the ET command completes a transaction without update commands (that is, A1, E1, N1, N2), Adabas returns binary zeros.

Otherwise, Adabas returns the transaction sequence number in binary format. These numbers are assigned in ascending sequence during a given user session, starting with 1.

File Number (ACBFNR)

A database ID is only necessary if you are accessing a database other than the application's default database (read in by ADARUN DBID parameter, provided in the loaded link globals table, or linked with the link routine).

Response Code (ACBRSP)

Adabas returns the response code for the command in this field. Response code 0 indicates that the command was executed successfully. If the ET command returns a non-zero response code, the rightmost two bytes of the 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.

ISN Lower Limit (ACBISL)

If the hold ISNs option is specified, this field must contain the count of six-byte ISN buffer entries.

ISN Quantity (ACBISQ)

In this field, Adabas returns the transaction duration time specified in timer units (each unit equals 1.05 seconds).

Record Buffer Length (ACBRBL)

If user data is to be stored in an Adabas system file, the number of bytes of user data to be stored must be specified in this field.

Adabas will store the number of bytes specified in this field. The maximum number of bytes which may be specified is 2000 bytes.

If no user data is to be stored, this field is not used.

ISN Buffer Length (ACBIBL)

The ISN buffer length (in bytes). This length is required only if the hold ISNs or multifetch option is used (see the Command Option 1 field description).

Command Option 1: Hold ISNs Option (ACBCOP1)

Note:

If multifetch is set with ADARUN PREFETCH=YES, the "P" option is automatically used for ET/BT commands (the "M" option is automatically used for all other commands). You *cannot* override this option setting by using this field.

By default as part of ET command execution, Adabas releases all ISNs currently held by the user.

Option	Description
P	Places all or a portion of the ISNs back into hold status. The ISNs to be returned to hold status must be provided in the ISN buffer, and the ISN count must be specified in the ISN lower limit field.
M (command-level multifetch)	Releases only a subset of the records held by the current transaction. The records to be released from hold status are specified in the ISN buffer. The first fullword in the ISN buffer specifies the number of eight-byte elements and each following eight-byte group is interpreted as one file number/ISN identifier of records to be released from hold status.

Command Option 2: Store User Data (ACBCOP2)

Option	Description
E	Indicates that user data is to be stored in an Adabas system file.

ACB Examples**Example 1: ET without User Data****Control Block**

Command Code	ET	
Command Option 2	b	no user data is to be stored

Example 2: ET with User Data**Control Block**

Command Code	ET	
Record Buffer Length	25	25 bytes of user data to be stored
Command Option 2	E	user data to be stored

Buffer Areas

Record Buffer	USER DATA FOR TRANSACTION
---------------	---------------------------

Example 3: ET with Hold ISN Option**Control Block**

Command Code	ET	
ISN Lower Limit	3	
ISN Buffer Length	18	
Command Option 1	P	place ISNs in hold status

Buffer Areas

ISN Buffer	
000600000001	ISN 1
000600000002	ISN 2
000600000003	ISN 3

ACBX Interface Direct Call: ET Command

This section describes ACBX interface direct calls for the ET 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
Command ID	13-16	alphanumeric/ binary	---	A
Database ID****	17-20	numeric	F	U
	21-36	---	---	---
ISN Lower Limit	37-40	binary	F	U
	41-44	---	---	---
ISN Quantity	45-48	binary	---	A
Command Option 1	49	alphanumeric	F	U
Command Option 2	50	alphanumeric	F	U
	51-114	---	---	---
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 *	F	U
ISN **	F	U

where:

- F Supplied by user before Adabas call
- A Supplied by Adabas
- U Unchanged after Adabas call
- * Required only of ET data to be stored
- ** Required for hold ISN option; optional for multifetch option
- *** Not used but should be included in Adabas call if ET data is to be stored. If not specified, one will be automatically generated.
- **** A database ID is only necessary if you are accessing a database other than the application's default database (read in by ADARUN DBID parameter, provided in the loaded link globals table, or linked with the link routine).
- 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)

ET

Response Code (ACBXRSP)

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 Error Subcode (ACBXERRC) field, are described in the *Adabas Messages and Codes Manual* documentation.

Command ID (ACBXCID)

Adabas returns either binary zeros or the transaction sequence number in this field.

If the ET command completes a transaction without update commands (that is, A1, E1, N1, N2), Adabas returns binary zeros.

Otherwise, Adabas returns the transaction sequence number in binary format. These numbers are assigned in ascending sequence during a given user session, starting with 1.

Database ID (ACBXDBID)

Use this field to specify the database ID only if you are accessing a database other than the application's default database (read in by ADARUN DBID parameter, provided in the loaded link globals table, or linked with the link routine). 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.

ISN Lower Limit (ACBXISL)

If the hold ISNs option is specified, this field must contain the count of six-byte ISN buffer entries.

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)

In this field, Adabas returns the transaction duration time specified in timer units (each unit equals 1.05 seconds).

Command Option 1: Hold ISNs Option (ACBXCOP1)

By default as part of ET command execution, Adabas releases all ISNs currently held by the user.

Option	Description
P	places all or a portion of the ISNs back into hold status. The ISNs to be returned to hold status must be provided in the ISN buffer, and the ISN count must be specified in the ISN lower limit field.
M (command-level multifetch)	Releases only a subset of the records held by the current transaction. The records to be released from hold status are specified in the ISN buffer. The first fullword in the ISN buffer specifies the number of eight-byte elements and each following eight-byte group is interpreted as one file number/ISN identifier of records to be released from hold status.

Command Option 2: Store User Data (ACBXCOP2)

Option	Description
E	Indicates that user data is to be stored in an Adabas system file.

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

The following buffers apply to an ET command:

- Format Buffer
- Record Buffer
- ISN Buffer

Format Buffer

A format buffer is not used by the ET command, but should be included in the Adabas call if ET data is to be stored. 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

The user data to be stored in an Adabas system file is provided in this buffer, if ET data is to be stored.

The data is retained until you issue the next ET or CL command containing ET data. The user data will be retained when the user session terminates only if you issued an OP command in which a unique, non-blank user ID was provided.

ISN Buffer

If the Command Option 1 field is set to "P", each ISN to be returned to hold status must be provided as a six-byte binary entry in which the first two bytes contain the number of the file in which the record is contained, and the next four bytes contain the ISN of the record to be held. Neither the file numbers nor the ISNs are checked for validity.

If the Command Option 1 field is set to "M", only a subset of the records held by the current transaction are to be released. The first fullword in the ISN buffer specifies the number of eight-byte elements, and each following eight-byte group is interpreted as one file number/ISN identifier of records to be released from hold status. For more information, read *BT / ET Multifetch Processing*.