About Adabas Commands

This section provides an overview of the Adabas commands categorized by function: database query, database modification, Data Storage read, Associator read, logical transaction processing, and special commands.

In addition, Adabas command facilities related to data protection, recovery, and user restart are described. The *transaction* concept is introduced and ET logic operations are explained. Competitive updating is discussed for ET logic (record hold/release and avoidance of resource deadlock) and exclusive control users; nonactivity timeouts are described for all user types.

This chapter covers the following topics:

- Command Types

- Transactions and ET Logic

- Competitive Updating

- Non-Activity Time Limit

Command Types

The commands can be categorized into the following functions:

- Database Query Commands (Sx)

- Data Storage Read Commands (L1-L6)

- Associator Read Commands (L9, LF)

- Database Modification Commands (A1, E1, N1/N2)

- Logical Transaction Control Commands (ET/BT)

- Special Commands

Database Query Commands (S*x*)

Database query commands (S1/S4, S2, S5) search for and return the ISNs of specified records or record groups according to specified search criteria. Other commands in this category (S8, S9) sort the resulting ISN lists in preparation for later operations.

The ISN lists resulting from any S*x* command may be saved on the Adabas Work data set for later retrieval during your user session.

In most cases, these commands do not actually read the database; ISNs are read directly from the Associator's inverted lists. Options allow the ISN's record to be placed in hold status to prevent its being updated by other programs until the record is released; if desired, additional field values contained in the first ISN's record can be read from Data Storage.

Note:
The behavior of nondescriptor searches in Adabas databases differs between mainframe and open systems in regards to null suppression in the fields. In open systems, nondescriptor searches do not return records with null values in a field if the field is null-suppressed (NU); on mainframe systems, the null-suppression (NU) of fields is ignored during nondescriptor searches. At this time, to resolve this problem, we recommend that you remove the null suppression option (NU) for open systems fields, if the fields must be used for a nondescriptor search.

This section covers the following database query commands:

- S1/S4
- S2
- S5
- S8
- S9

S1/S4

The S1/S4 command selects the records that satisfy given search criteria.

- If only descriptors are used, the query is resolved using the inverted list alone without accessing Data Storage.

- If no descriptors are included in the search criteria, the S1/4 command reads each record in Data Storage to resolve the query.

- If both descriptors and nondescriptors are used within the search criteria, Adabas first searches the inverted list for the descriptors and then reads the Data Storage for all matching ISNs to check the nondescriptors.

S1 and S4 commands return the count of records that satisfy the search criterion, and a list of the ISNs for those records. An option permits the record identified by the first ISN in the resulting ISN list to be read from Data Storage.

The S4 command may be used to place the record identified by the first ISN in the ISN list in hold status. This prevents another user from updating the record until it is released.

S2

The S2 command is equivalent to the S1 command except that the ISNs of the records selected are returned in the sort sequence of a user-specified descriptor (or descriptors). One to three descriptors may be used. Ascending or descending sequence may be specified.

S5

The S5 command is used to select the ISNs of records in one file which are coupled to a record with a given ISN in another file.

You specify the file and ISN for which coupled ISNs are to be returned and the file in which the coupled records are to be selected. Adabas returns the number of records coupled to the ISN and the list of the coupled ISNs.

S8

The S8 command performs logical (AND, OR, or NOT) operations on two ISN lists previously created by an S1/S4, S5, S8, or S9 command.

- AND results in a single ISN list containing ISNs present in both source lists.

- OR results in a single ISN list containing ISNs present in either source list.

- NOT results in a single ISN list containing ISNs present in the first list but not in the second list.

S9

The S9 command sorts an ISN list created previously by a S1/S4, S2, S5, S8 or S9 command.

The ISN list may be sorted by ISN (ascending sequence only), or by one to three user-specified descriptors (ascending or descending sequence).

Data Storage Read Commands (L1-L6)

The L1 through L6 commands are used to read actual records from Data Storage. Depending on the specified command and its options, records are read individually in the sequence in which they are stored, in the order of an ISN list created by one of the database query commands, or in logical sequence according to a user-specified descriptor.

A hold option allows the database records to be locked until released by a separate command or at transaction end.

This section covers the following data storage read commands:

- L1/L4
- L2/L5
- L3/L6

L1/L4

The L1 command reads a single record from Data Storage. You specify the file number, ISN of the record to be read, and the fields for which values are to be returned. Adabas returns the requested field values.

The L4 command is the same as the L1 command except that the record is placed in hold status. This prevents other users from updating the record until it is released.

The multifetch/prefetch option prefetches records on either a session or command basis. This can reduce the overhead for multiple record fetches. The multifetch option is platform-independent.

The GET NEXT option may be used to read one or more records identified by ISNs contained in an ISN list without requiring that you specify each ISN. Usually, the ISN list is created by a previous S*x* command.

The response code option issues a response code 145 if the L4 command cannot place a record in hold status because it is being held by another user. Otherwise, you are placed in wait status until the ISN (and record) are freed or the waiting user's transaction is timed out.

The ISN sequence option may be used to read records in ISN sequence. The ISN you specified is read, unless it is not present, in which case the record which has the next higher ISN is read.

L2/L5

The L2 command reads the records from a file in the sequence in which they are physically stored in Data Storage. You specify the file to be read and the fields for which values are to be returned. Adabas returns the requested field values.

The L5 command is the same as the L2 command except that the record read is placed in hold status. This prevents other users from updating the record until it is released. The multifetch/prefetch and response code options (see the L1/L4 command description, above) also apply to the L2/L5 commands.

L3/L6

The L3 command reads records from Data Storage in the logical sequence of a user-specified descriptor. You specify the file to be read, the descriptor to be used for sequence control, and the fields for which values are to be returned. Adabas returns the requested field values.

Command Option 2 for the L3/L6 command specifies whether the records are read in ascending or descending order. In addition, Command Option 2 can be used to specify a start value.

The L6 command is the same as the L3 command except that the record read is placed in hold status. This prevents other users from updating the record until it is released.

In addition to the multifetch/prefetch and response code options (see the L1/L4 command description, above), the start value option permits reading to begin at a user-specified value or ISN.

Associator Read Commands (L9, LF)

The L9 and LF commands read information directly from the Associator inverted lists or field definition tables (FDTs), returning either the inverted list values for a specified descriptor or the field definitions for a specified file in the database.

This section covers the following associator read commands:

- L9
- LF

L9

The L9 command returns each value contained in the inverted list for a given descriptor, and the number of records in which the value is contained.

You specify the file and descriptor for which values are to be returned, the value at which the command is to begin, and whether the values are returned in ascending or descending sequence.

LF

The LF command returns the field definitions for a file. You specify the file for which the field definitions are to be returned.

The field definitions for all the fields in the file are returned. Each field definition consists of the field name, level number, standard format, standard length, and definition options.

Database Modification Commands (A1, E1, N1/N2)

Database modification commands (A1, E1, and N1/N2) add, change, or delete database records and update the related Associator lists accordingly. You can assign ISNs to new records or they can be assigned by Adabas.

This section covers the following database modification commands:

- A1
- E1
- N1/N2

A1

The A1 command updates the contents of one or more fields within a record. You specify the file and ISN of the record to be updated, together with the fields to be updated and the values to be used for updating.

Adabas performs all necessary modifications to the Associator and Data Storage. Associator updating is required only if one or more descriptors are updated.

A hold option is available for the purpose of placing the record to be updated in hold status prior to the update.

E1

The E1 command deletes a record or refreshes a file. You specify the file and ISN of the record to be deleted, or specify the file only (without an ISN and command ID) to refresh the file.

Adabas performs all necessary modifications to the Associator and Data Storage.

A hold option is available for the purpose of placing the record to be deleted in hold status prior to the deletion.

N1/N2

The N1/N2 command adds a new record to a file. You specify the file to which the record is to be added together with the fields and field values to be used. Adabas performs all necessary modifications to the Associator and Data Storage.

If the N1 command is used, the ISN for the new record is assigned by Adabas. If N2 is used, you must provide the ISN.

Logical Transaction Control Commands (ET/BT)

An Adabas logical transaction defines the logical start (BT) and end (ET) of the database operation being performed. If the user operation or Adabas itself terminates abnormally, these commands make it possible to restart a user, beginning with the last unsuccessfully processed transaction. ET/BT commands define the transaction start and end, restore pretransaction conditions if a situation occurs that prevents successful completion of the transaction, and read program-specified user data written during the transaction sequence.

Programs that use these commands are called *ET logic programs*. Although not required, Software AG recommends that you use ET logic.

This section covers the following logical transaction control commands:

- BT
- ET

BT

The BT command backs out the current transaction being processed.

All modifications resulting from updates performed during the transaction are removed, and all records placed in hold status during the transaction are released (unless kept in hold status by the multifetch option; see the section *Multifetch Operation Processing*).

ET

The ET command indicates the end of a logical transaction.

An ET command causes Adabas to physically store all data protection information related to the transaction. This information is used to apply all the updates performed during the transaction at the start of the next Adabas session if the current session is terminated before these updates are physically applied to the database.

The ET command releases all the records that have been placed in hold status during the transaction (unless kept in hold status by the multifetch option; see the section *Multifetch Operation Processing*). The ET command may also be used to store user data in an Adabas system file. This data may be retrieved with an OP or RE command.

Special Commands

Special commands perform many of the housekeeping functions required for maintaining the Adabas database environment. Commands in this group allow you to perform the following functions:

- Open and close a session (but not to control a transaction).

- Write data protection information and checkpoints.

- Set and release record hold status.

This section covers the following special commands:

- CL
- C1
- C3
- C5
- HI
- OP
- RC
- RE
- RI

CL

The CL command terminates a user session, releasing all records held for that user.

- It physically writes all current data protection information to the data protection log.

- It releases all records currently in hold status for the user.

- It releases all the command IDs and corresponding ISN lists currently assigned the user.

- It stores user data in an Adabas system file (optional).

C1

The C1 command causes a checkpoint to be taken.

The C1 command physically writes all current data protection information to the data protection log, and writes a checkpoint entry to the data protection log and the system checkpoint file. This checkpoint entry may be used as a reference point for subsequent removal or reapplication of updates. An option allows the C1 command to initiate a buffer flush.

C3

The C3 command, issued only by exclusive control and update users (who are not using ET logic), writes a SYNX-03 checkpoint in the Adabas checkpoint file.

- The checkpoint contains the current data protection log and block number.

- The checkpoint may be used to restore the database (or certain files) to the status in effect at the time the checkpoint was taken. This may be necessary before a program performing exclusive control updating can be rerun or restarted.

If Command Option 2 is specified, the C3 command also stores user data in the Adabas checkpoint file for restart purposes. The stored data may be subsequently read with an OP or RE command.

C5

The C5 command writes user data to the Adabas data protection log.

The data can be read subsequently using the ADASEL utility.

HI

The HI command places a record in hold status. Specify the file and ISN of the record to be placed in hold status.

A record placed in hold status cannot be updated by another user until it is released.

OP

An OP command is mandatory when any of the following apply:

- The nucleus is run with ADARUN parameter OPENRQ=YES

- Exclusive file control (EXF) is to be performed

- User data that was stored in an Adabas system file by a previous ET command is to be read

- User data is to be stored in an Adabas system file, using a C3, CL, or ET command

- You are assigned a special processing priority

- You are an access-only user (no update commands permitted).

- A transaction time limit or a non-activity time limit is to be set for you that differs from that specified by ADARUN parameters TT or TNA*x*, respectively. Your setting must conform to the maximum setting set by the ADARUN parameters MXTT and MXTNA, respectively.

- Special data encoding or architecture is to be specified for your user session.

It may also be used to set the maximum number of records that you can place in hold status at the any given time, and the maximum number of command IDs you may have active at the same time.

RC

The RC command may be used to release one or more command IDs currently assigned to you, or to delete one or all global format IDs.

RE

The RE command reads user data previously stored in an Adabas system file by CL or ET commands.

RI

The RI command releases a record from hold status.

Specify the file and ISN of the record to be released. You may also request that all records currently held by you are released. The RI command should be issued by non-ET users only.

Transactions and ET Logic

This section describes the concept of a *transaction* and an *ET logic user*; it explains ET logic operations including normal and abnormal transaction termination processing and the storage and retrieval of user (ET) data.

This section covers the following topics:

- What Is A Logical Transaction?

- Transaction Sequence Number

- ET Transaction Time Limit

- Back Out Transaction (BT) Command

- Autobackout

- End Transaction (ET) Command

- User (ET) Data

- Adabas User ID

What Is A Logical Transaction?

A *logical transaction* is the smallest unit of work (as defined by the user) that must be performed in its entirety to ensure the logical consistency of the information in the database. Users who use logical transaction commands are referred to as ET logic users.

A logical transaction comprises one or more Adabas commands that read or update the database as required to complete a logical unit of work. A logical transaction begins with the first command that places a record in hold status and ends when an ET, BT, CL, or OP command for the same user is issued.

The RE (read ET data) command can be used to retrieve user restart data stored by the ET or CL command.

Transaction Sequence Number

When a program issues an ET or CL command, Adabas returns a transaction sequence number in the command ID field of the ET or CL command's control block. The transaction sequence number is a count of the total number of ET commands issued thus far during the user session.

The transaction sequence number is set to 1 for the first ET command issued by the user. The first ET command following the OP command returns transaction sequence number 2 in the Command ID field or the Adabas control block. Each subsequent OP command returns the transaction sequence number of the last ET command issued by that same user.

ET Transaction Time Limit

Adabas provides a transaction time limit for programs that use ET logic. The time measurement for a transaction begins when the first command places a record in hold status, and ends when the program issues an ET, BT, or CL command.

The time limit is set with the ADARUN TT parameter; a transaction time limit that overrides this general limit for a specific user can be set with the OP command; this limit is controlled by the ADARUN MXTT parameter.

If a transaction exceeds the prescribed limit, Adabas generates a BT (back out transaction) command. The BT command removes all the updates performed during the transaction and releases all held records.

Adabas returns response code 9 for the next command issued by the user indicating that the current transaction has been backed out. The user can either repeat the backed out transaction from the beginning or begin another transaction.

Back Out Transaction (BT) Command

The BT command removes all updates made during the transaction currently being processed. This may be necessary because of a program error, a timeout, or when Adabas determines that the transaction cannot be completed successfully.

A BT command also performs an implied ET command, which releases all the records held during the transaction unless otherwise specified by the multifetch option; for more information, see the section *Multifetch Operation Processing*.

For example, the command sequence

```
FIND (S4)
  UPDATE (A1) (modify field XX to value 20)
  FIND (S4)
  UPDATE (A1) (modify field YY to value 50)
  END TRANSACTION (ET)
  FIND (S4)
  UPDATE (A1) (modify field XX to value 10)
  BACKOUT TRANSACTION (BT)
```

results in the field values of XX = 20 and YY = 50. The second update to field XX is removed by the BT command.

Autobackout

Autobackout, which is performed only for ET logic users, backs out transactions automatically in sessions that end abnormally. Adabas performs an internal BT (back out transaction) followed by autorestart, and then returns response code 9 to indicate that the last transaction has been backed out.

Autobackout occurs at the end of any nucleus session that was terminated with HALT.

Autobackout also occurs at the beginning of the next Adabas session to remove any updates that were performed in partially completed transactions by ET logic users during the previous terminated session.

When response code 9 indicates that the transaction was backed out, the user has the option of either reissuing the transaction from the start or beginning another transaction.

To restart the backed-out transaction, a terminal operator may need to reenter the data for the transaction, or an internal restart may have to be performed from the beginning of the "update phase" of the transaction; the "update phase" of a transaction begins with the first command that places a record in hold status.

End Transaction (ET) Command

The ET command must be issued at the end of each logical transaction. Successful execution of an ET command ensures that all the updates performed during the transaction will be physically applied to the database regardless of subsequent user or Adabas session interruption.

Updates performed within transactions for which ET commands have not been successfully executed will be backed out by the autobackout routine (see *Autobackout*).

Unless otherwise specified by the multifetch option, the ET command releases all records held by the user during the transaction. Adabas returns a unique transaction sequence number (see *Transaction Sequence Number*) that can be used to identify the transaction for auditing or restart purposes.

The ET command may also be used to store user data in an Adabas system file. This data may be used for user restart purposes, and may be read with an OP or RE command.

User Program	Adabas
FIND (S4), UPDATE (A4)	record updated in Adabas buffer but not necessarily written to the database
FIND (S1), HOLD ISN (HI), UPDATE (A4)	record updated in Adabas buffer but not necessarily written to the database
END TRANSACTION (ET)	data protection information for the transaction is written to the Adabas Work and data protection log, ending the transaction
FIND (S4), UPDATE (A4)	record updated in Adabas buffer but not necessarily written to the database
FIND (S1), HOLD ISN (HI), UPDATE (A4)	record updated in Adabas buffer but not necessarily written to the database
. . . Adabas or user session interruption . . .	

When the next Adabas session is started, or when the user is timed out, both updates for transaction 1 are physically written to the database (if they were not previously written). The updates for transaction 2 are not physically written (or will be backed out) because no ET command was processed for this transaction.

User (ET) Data

User data (ET data) may be stored with an ET or CL command and read with an OP or RE (read ET data) command.

One record of user data is kept for each user ID. The data is maintained until the next ET or CL command is issued in which user data is provided.

Each user data record is also written to the Adabas data protection log with each checkpoint written by the transaction. This data may be subsequently read with the ADASEL utility.

User data is primarily used to perform the following functions:

- Store data needed to restart an operation (e.g., input message data, transaction identification data, transaction summary data).

- Store intermediate data to be used by subsequent transactions (e.g., for audit trail purposes).

- Communicate with other users. The data stored may be read by other users provided that the ID of the user who stored the data is supplied in the OP command.

Software AG also suggests using the user data facility to perform the following functions:

- Establish an installation standard for the user data record format and store data for each update transaction with the ET command.

- Read the user data and display it in a standard message format when the user logs on to an application. The user is thus informed of the last successful updating activity which corresponds to his user ID.

Software AG recommends that you keep ET commands that provide user data separate from those that do not. Combining the two types of ET command may cause significant additional overhead by forcing Adabas to repeatedly copy user data from the Adabas Work data set (where it is temporarily stored) to the protection log.

Adabas User ID

The user ID is an identifier used by Adabas to store and retrieve user (ET) data and to assign a special processing priority to a user. The user ID is specified in the Additions 1 field of the OP command.

A user ID provided by the user must meet both of the following criteria:

- It must begin with the character A (X'C1') through 9 (X'F9')

- It must be unique to ensure that the user (ET) data is related to the user regardless of the terminal used. User (ET) data is maintained until the user's next ET or CL command in which user (ET) data is provided.

If the user either does not provide a user ID or provides an invalid user ID, Adabas establishes a default user ID for the user session, and any user (ET) data stored by the user is deleted when the current session is terminated.

To avoid later limitations, Software AG recommends that you always specify a user ID.

Competitive Updating

Competitive updating is in effect when two or more users (in multiuser mode) are updating the same Adabas file(s).

This section describes the Adabas facilities used to ensure data integrity in a competitive updating environment. These include record hold and release processing and the avoidance of resource deadlock, as well as exclusive control updating.

This section covers the following topics:

- ET User Record Hold

- Avoiding Resource Deadlock

- Exclusive Control Updating

ET User Record Hold

The record hold facility allows the ET user to place a record in hold status for updating without allowing interim updating of the record by another user. Adabas *holds* a record by placing the ISN of the record in the hold queue; as a result, record hold is also called an *ISN hold*.

Record hold is applicable for all ET logic users; Adabas does not place an ISN in hold status for EXU (exclusive control updating) users. Read *Exclusive Control Updating*, for more information.

This section covers the following topics:

- Record Hold Commands
- Record Update Using the Hold Option
- Record Release

Record Hold Commands

A record is held when you use the find-with-hold command (S4), the read-with-hold commands (L4, L5, L6), an A1 or E1 command in which the hold option is specified, or a hold record (HI) command. An N1/N2 command issued by an ET logic user also places the added record in hold status.

The successful completion of any of these commands places the record (ISN) in hold status. If the record is already being held by another user, the user issuing the record hold command is placed in a wait status until the record becomes available, at which time Adabas reactivates the command.

If the R (return code) or O (multifetch/return code) option is used with any of the record hold commands and the record to be held is already being held by another user, Adabas returns response code 145 instead of placing the user in a wait status.

If you issue a find (S1) or read (L1, L2, L3) command, you can access the record regardless of the fact that the record is in hold status for another user.

Record Update Using the Hold Option

You can update or delete any records they have in hold status by issuing an A1 or E1 command.

An A1 command is executed only if the record is either in hold status for the requesting user, or free from hold status and the you specify that the record should be held. If the record is currently held by another user, your hold request is either placed in wait status or, if you request, you will receive response code 145. If the record is not in hold status, response code 144 is returned, indicating that the record was placed in hold status.

If an E1 command (without the hold option) is issued for a record that is not in hold status for you, Adabas places the record in hold status for you provided that the record is not in hold status for another user. If you do not place a record to be deleted in hold status, there is no guarantee that the record will not be updated or deleted by another user before the E1 command is executed.

If an N1/N2 command is issued and there is no available space in the hold queue, response code 145 is returned.

Record Release

An ET logic user releases records from hold status with the ET command following each logical transaction, and with a close (CL) command at the end of the Adabas session. Programs using ET logic should not release records with the release record (RI) command if any updating has been performed during the current transaction, since this could result in a loss of data integrity.

For example, a record is updated and then released with an RI command by ET user 1, and the transaction continues. In the meantime, the same record is updated by user 2, who then ends the session with an ET command. If user 1's transaction is subsequently backed out due to an error or transaction timeout, the updates performed by both users 1 and 2 are removed even though user 2's transaction completed successfully with an ET command.

Therefore, user programs that employ ET logic should only release records at the completion of a logical transaction with the ET command. Non-ET logic users should use A1 or E1 commands to update or delete records and then release them.

The multifetch option allows records (and their ISNs) to be released from hold status selectively if a non-zero count and the file number/ISN is specified in the ISN buffer when the ET (or BT) command is issued. See the section *Multifetch Operation Processing*.

The CL command releases all records in hold status for the issuing user, whether an ET user or not.

Avoiding Resource Deadlock

A resource deadlock can occur if two users are placed in wait status because each had requested a record that was currently in hold status for the other user.

For example:
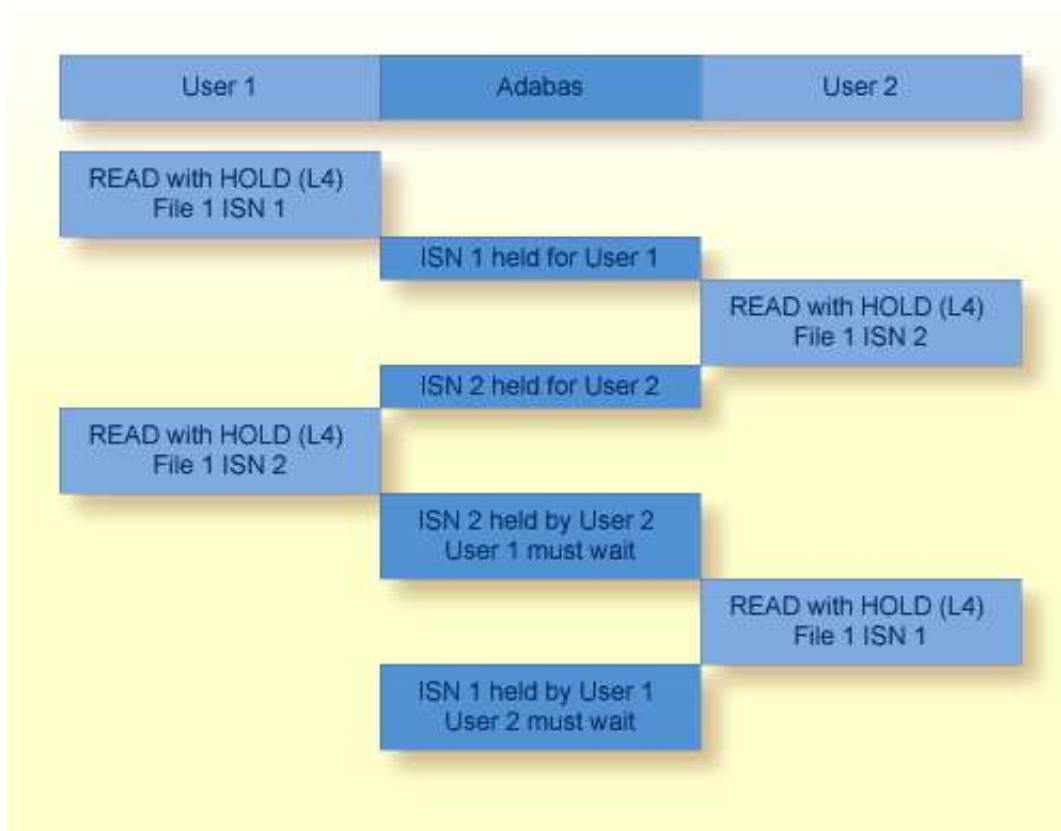
Resource Deadlock Example

Adabas detects the potential deadlock prior to putting user 2 in wait state and returns a response code 145 to the second user.

Exclusive Control Updating

You may request exclusive control of one or more Adabas files to prevent other users from updating the file during the execution of your user session.

Exclusive control is requested with the OP command and is given only if the file for which exclusive control is requested is not already opened for update by another user or utility. If the requested file is not available, the OP command returns response code 48.

In addition to preventing competitive updating of a file, exclusive control may be used to simplify recovery procedures, in that the files may be restored regardless of other user activity.

The record hold commands need not (but may) be used for files for which the user has exclusive control. Adabas disables hold logic processing for files being updated under exclusive file control.

Exclusive control users are assumed to be non-ET logic users, and therefore not controlled by the ET timeout restrictions. However, if a non-ET user issues an ET command, that user automatically becomes an ET logic user and is subject to transaction timeout restrictions if records are put in hold status for any reason. That user retains exclusive control until either finished or timed out.

Users performing exclusive control updating can use the C1 command to request that a checkpoint be taken. The C1 checkpoint acts as a reference point to either remove updates that have been applied after the checkpoint, or to reapply updates that were applied before the checkpoint.

Non-Activity Time Limit

All users (access-only, ET logic, or exclusive control) are subject to a non-activity time limit. Different non-activity time limits may be defined for each user type with the ADARUN TNAA, TNAE, and TNAX parameters (for more information, read *Adabas Initialization (ADARUN Statement)*).

A user-specific non-activity time limit may also be set with the OP command; the maximum is controlled with the ADARUN MXTNA parameter. For more information, read *OP Command: Open User Session*.

The following diagram summarizes the actions taken by Adabas when a user exceeds the non-activity time limit.
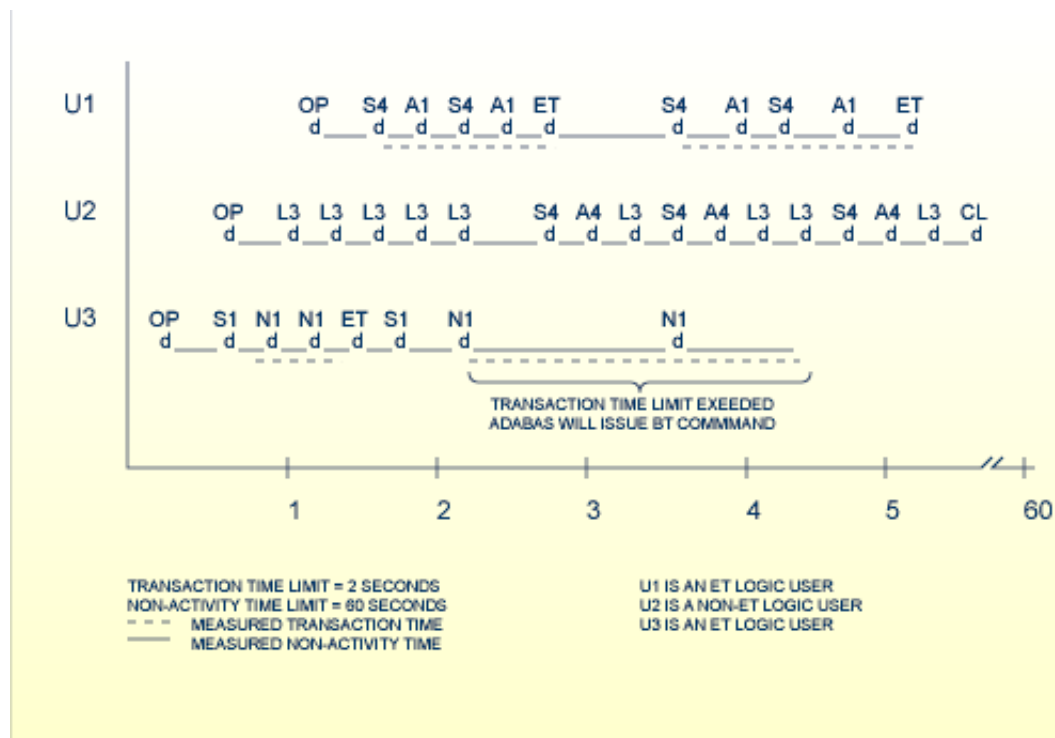
Transaction and Non-Activity Time Limits

This section describes these Adabas actions in the following topics:

- For an ET Logic User

- For an Exclusive File Control User (EXF)

- For an Access Only User

For an ET Logic User

For an ET logic user, Adabas performs the following processing:

- It issues a backout transaction (BT) command for the current user transaction (if necessary).

- It releases all records held during the transaction.

- It deletes the user's file list in the UQE.

- It deletes all command IDs for the user.

Adabas returns response code 9 to the user when the next command is issued if one of the following conditions occur:

- The user was not in ET status when the timeout occurred.

- The user was in ET status when the timeout occurred and provided a non-blank user ID in the OP command.

If the user was at ET status when the timeout occurred and did *not* provide a non-blank user ID in the OP command, the user's UQE is deleted.

For an Exclusive File Control User (EXF)

For an exclusive file control user, Adabas deletes the user's file list in the UQE. As a result, the user loses exclusive control of the file or files for which exclusive control was in effect.

In addition, all command IDs for the user are released and the user is changed to an access-only user.

For an Access Only User

For an access-only user, Adabas deletes the user's file list in the UQE.