DBCLOSE

INTRINSIC NUMBER 403

Terminates access to a database or a data set, or rewinds a data set. `DBCLOSE` is used in conjunction with `DBOPEN` to establish and terminate access to a database.

OPENTURBO Specific

100%

OPENTURBO Performance Enhancement

N/A.

OPENTURBO Additional Feature

N/A.

Syntax

DBCLOSE, base,dset,mode,status

Parameters

`base` is the name of an array used as the *base* parameter when opening the database. The first element of the array must contain the base ID returned by `DBOPEN`. (Refer to `DBOPEN` for more information about the base ID.)

`dset` is the name of an array containing the left-justified name of the data set to be closed, or is an integer referencing the data set by number if *mode* equals 2 or 3. If *mode* equals 1, this parameter is ignored. The data set name can be up to 16 characters long. If shorter, it must be terminated by a semicolon or a blank.

`mode` is an integer equal to 1, 2, or 3 indicating the type of termination desired as follows:

Mode 1: Access to the database is terminated. Any locks held by this user for this base ID are released. If `DBCLOSE` mode 1 is called while a dynamic transaction is still active, an error is returned, the transaction is aborted, and the database is closed automatically. You do not need to call DBXEND or DBXUNDO.

Mode 2: The data set referenced by the *dset* array is closed, but locks held in the data set are not released. If `DBCLOSE` mode 2 is called while a dynamic transaction is still active, an error is returned. You must check the error and decide to use `DBXEND, DBXUNDO`, or continue with the transaction even if `DBCLOSE` mode 2 failed. `DBXUNDO` will abort the entire dynamic transaction. `DBXEND` will terminate the dynamic transaction; the modifications completed thus far within the transaction will remain in the database.

Mode 3: If *mode* equals 3, the data set referenced by the *dset* array is reinitialized but not closed.

If your database is enabled for third-party indexing (TPI), refer to your vendor documentation for additional DBCLOSE mode information. The section on DBUTIL in chapter 8 of this book has a brief description of the TPI option.

status is the name of an array of 10 halfwords in which TurboIMAGE/XL returns status information about the procedure. If the procedure executes successfully, the status array contents are:

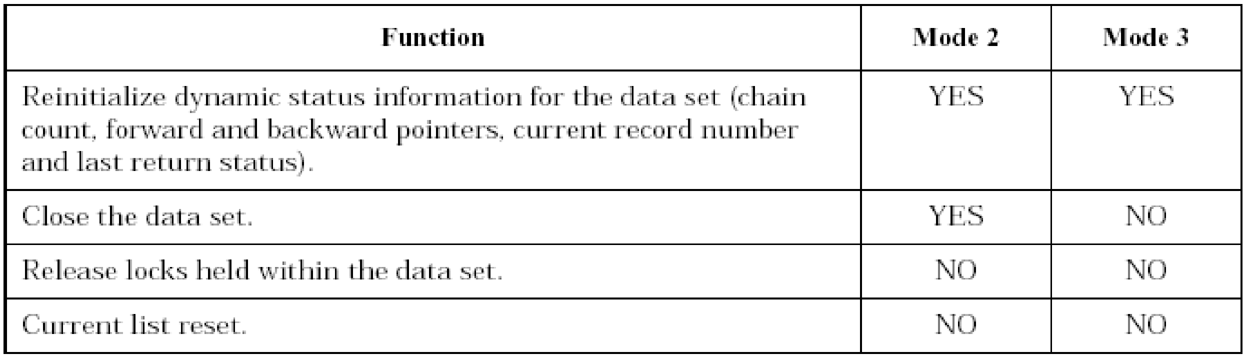
Element	Contents
1	If the procedure succeeds, the return status is 0. Table 5-5. describes the contents of element 1 when the procedure does not succeed.
2-4	Unchanged from previous procedure call using this array.
5-10	Procedure call information. Refer to “Library Procedure Error Messages” in appendix A for a description of this information.

Discussion

You must call DBCLOSE mode 1 to terminate access to the database when you have completed all the tasks you want to perform. If a process has issued multiple calls to DBOPEN for the same database, only the access path specified in the DBCLOSE base parameter is affected by the call to DBCLOSE.

The capability to reset and close a data set is provided to perform functions such as reinitializing dynamic status information for a process accessing a particular data set and returning system resources. In both modes 2 and 3, status information is reinitialized, but system resources are returned in mode 2 only. The current list is not reset. Table 5-5. summarizes the functions performed in each mode.

Table 5-5. DBCLOSE Modes 2 and 3 Functions



Function	Mode 2	Mode 3
Reinitialize dynamic status information for the data set (chain count, forward and backward pointers, current record number and last return status).	YES	YES
Close the data set.	YES	NO
Release locks held within the data set.	NO	NO
Current list reset.	NO	NO

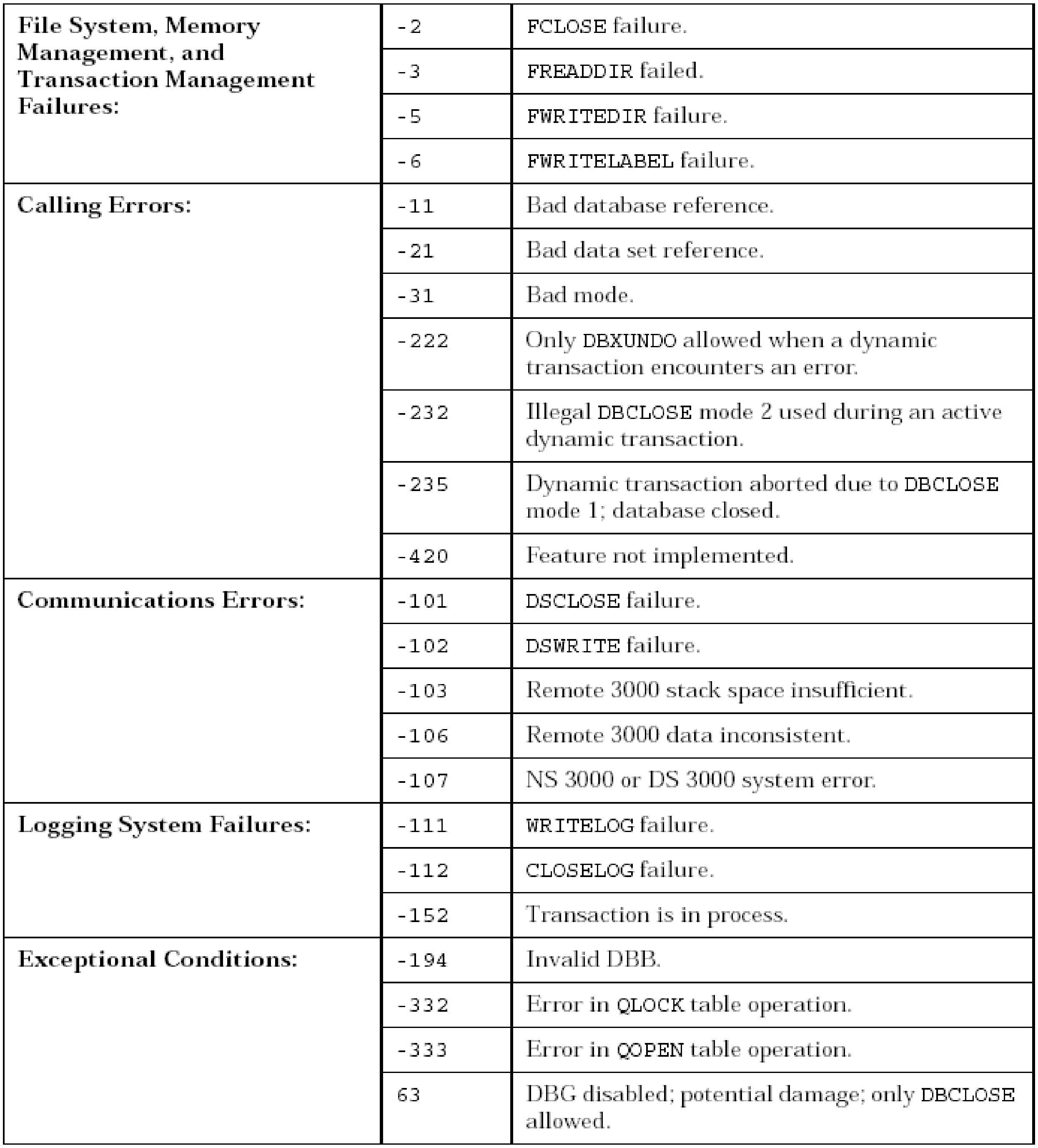
Because mode 3 does not close and reopen a data set, it is more efficient than mode 2 if the data set is to be accessed again before the database is closed.

Only mode 3 is allowed within a dynamic transaction; mode 2 will return an error, and mode 1 will abort the transaction.

If the process is logging, a mode 1 DBCLOSE causes a DBCLOSE log record to be written to the log file. DBCLOSE log records contain such information as the time, date, and user log identification number. A DBCLOSE log record is also written if the process aborts or terminates without closing the database. If the process aborts before completing an active transaction, a special DBEND log record is written prior to the DBCLOSE.

DBCLOSE returns an error condition if the process has not completed an active transaction; in other words, the process has called DBBEGIN without a matching call to DBEND. Transactions that abort in this manner are not automatically suppressed by DBRECOV during recovery in order to salvage as many subsequent transactions that may depend on the aborted transaction as possible.

Table 5-6. DBCLOSE Return Status Values



File System, Memory Management, and Transaction Management Failures:	-2	FCLOSE failure.
	-3	FREADDIR failed.
	-5	FWRITEDIR failure.
	-6	FWRITELABEL failure.
Calling Errors:	-11	Bad database reference.
	-21	Bad data set reference.
	-31	Bad mode.
	-222	Only DBXUNDO allowed when a dynamic transaction encounters an error.
	-232	Illegal DBCLOSE mode 2 used during an active dynamic transaction.
	-235	Dynamic transaction aborted due to DBCLOSE mode 1; database closed.
	-420	Feature not implemented.
Communications Errors:	-101	DSCLOSE failure.
	-102	DSWRITE failure.
	-103	Remote 3000 stack space insufficient.
	-106	Remote 3000 data inconsistent.
	-107	NS 3000 or DS 3000 system error.
Logging System Failures:	-111	WRITELOG failure.
	-112	CLOSELOG failure.
	-152	Transaction is in process.
Exceptional Conditions:	-194	Invalid DBB.
	-332	Error in QLOCK table operation.
	-333	Error in QOPEN table operation.
	63	DBG disabled; potential damage; only DBCLOSE allowed.

Consult appendix A for more information about these conditions.