

# Statement Logging

The Natural Database Log utility DBLOG allows you to ascertain which SQL statements are issued by a Natural program.

This section covers the following topics:

- Invoking DBLOG
- The DBLOG Menu for DB2
- The Trace Function
- The Snapshot Function

## Invoking DBLOG

The Natural DBLOG utility is used to log SQL statements issued to DB2.

### ▶ To invoke DBLOG for DB2

- Enter the command TEST DBLOG Q at the Natural NEXT prompt:

```

TEST DBLOG Q      [ ON
                   OFF
                   SHOW
                   CONT
                   MENU
                   ?
                   * ]
  
```

Parameter	Explanation
Q	Starts logging of SQL statements.
ON	Starts the trace function for non-selective logging.
OFF	Terminates logging.
SHOW	Terminates logging and displays the current log.
CONT	Continues logging if a log already exists and displays the current log.
MENU / ? / *	Selective logging of SQL statements. The DBLOG menu is displayed for selection. Any existing log is deleted.

The statements are logged in the Natural debug buffer, which has to be specified via the DSIZE profile parameter, as described in the Natural Parameter Reference documentation.

If the Natural debug buffer is too small to contain all statements to be logged, only the most recent statements are held in the debug buffer.

If you enter the TEST DBLOG Q command without any parameters, logging of SQL statements is activated, or - if it has already been activated - the current log is displayed.

## The DBLOG Menu for DB2

If you enter command TEST DBLOG Q with the parameter "MENU", "?", or "\*", the DBLOG Menu for DB2 is displayed, which offers the possibility of selective logging of SQL statements:

```

12:36:57          ***** NATURAL Test Utilities *****          1999-02-08
User SAG              - DBLOG Menu -                          Library SYSSQL

          Code  Function
          ----  -
          B     Begin Logging of SQL Statements
          E     End and Display Log Records
          S     Snapshot of Specific SQL Statement
          .     Exit
          ----  -

Code .. _

Statement .. _____ Skip ..... _____ Program .... _____
Line from .. 0000

Low SQLC .. ____ High SQLC .. 999 Line to .... 0000

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Print Exit  Begin End  Snap                               Canc
    
```

On the DBLOG Menu for DB2, you can specify which SQL statements are to be logged. Each statement is logged after it has been processed by DB2.

Code	Explanation
B	Corresponds to the command TEST DBLOG Q ON, which means that the trace function for selective logging is started and all SQL statements matching the selection criteria are logged.
E	Corresponds to the commands TEST DBLOG Q OFF/SHOW, which means that logging of SQL statements is stopped and the resulting log records are displayed.
S	Is used to log and display a specific SQL statement only.

The following selection criteria are available:

<b>Parameter</b>	<b>Explanation</b>
Statement	Only statements with the specified statement code are to be logged (for example SELECT).
Skip	Number of statements to be skipped before logging is to start (only applicable with function code "S").
Program	Only statements issued by the specified program are to be logged.
Line from/ Line to	Only statements within the range of the specified line numbers are to be logged.
Low SQLC/ High SQLC	Only statements which result in an SQL return code within the specified range are to be logged.

# The Trace Function

## To invoke the Trace Function for SQL Statements

- Enter function code "B" and corresponding selection criteria on the DBLOG Menu, or enter the TEST DBLOG Q command without parameters or with parameter "ON".  
The trace function is active until you enter the TEST DBLOG Q command again (or TEST DBLOG Q OFF respectively), or enter function code "E" on the DBLOG Menu.
- Once you have entered function code "E" (or the command TEST DBLOG Q SHOW), the DBLOG Trace screen is displayed as shown in the following example.

### Example of the Trace Function for SQL Statements

1. Write the following program:

```

DEFINE DATA
LOCAL USING SQL-AUTO
LOCAL USING SQL-EMPL
END-DEFINE
FIND (1) SQL-AUTO WITH OWNER > 'ID----01'
  FIND (1) SQL-EMPL WITH PERS_ID = OWNER(0050)
    MOVE 'NAME3333333333333347' TO NAME
    UPDATE
  END-FIND
  MOVE 'MAKE3333333334' TO MAKE
  UPDATE
END-FIND
END TRANSACTION
FIND (1) SQL-AUTO WITH MAKE = 'MAKE3333333334'
  FIND (1) SQL-EMPL WITH PERS_ID = OWNER(0140)
    DISPLAY PERS_ID NAME MAKE
  END-FIND
END-FIND
END

```

2. Enter "TEST DBLOG Q" at the Natural NEXT prompt.  
The message "DBLOG started now" is displayed.
3. Enter "RUN".
4. Enter "TEST DBLOG Q" again.

The following screen is displayed:

14:08:14		***** NATURAL Test Utilities *****						1999-09-08			
User SAG		- DBLOG Trace -						Library SYSSQL			
M No	R	SQL Statement (truncated)	CU	SN	SREF	M	Typ	SQLC/W	Program	Line	LV
1		SELECT MAKE,MODEL,BODY_TYPE,H	01	01	0050	D	ASQ		DBTEST	0050	01
2		FETCH CURSOR	01	01	0050	D	ASQ		DBTEST	0050	01
3		SELECT PERS_ID,NAME FROM SAG.	02	02	0060	D	ASQ		DBTEST	0060	01
4		FETCH CURSOR	02	02	0060	D	ASQ		DBTEST	0060	01
5		UPDATE SAG.EMPLOYEES SET NAME	02	03	0060	D	ASQ		DBTEST	0080	01
6		CLOSE CURSOR	02	02	0060	D	ASQ		DBTEST	0060	01
7		UPDATE SAG.AUTOMOBILES SET MA	01	04	0050	D	ASQ		DBTEST	0120	01
8		CLOSE CURSOR	01	01	0050	D	ASQ		DBTEST	0050	01
9		COMMIT	00	00	0000	D	ASQ		DBTEST	0140	01
10		SELECT MAKE,MODEL,BODY_TYPE,H	01	01	0150	D	ASQ		DBTEST	0150	01
11		FETCH CURSOR	01	01	0150	D	ASQ		DBTEST	0150	01
12		SELECT PERS_ID,NAME FROM SAG.	02	02	0160	D	ASQ		DBTEST	0160	01
13		FETCH CURSOR	02	02	0160	D	ASQ		DBTEST	0160	01
14		CLOSE CURSOR	02	02	0160	D	ASQ		DBTEST	0160	01
15		CLOSE CURSOR	01	01	0150	D	ASQ		DBTEST	0150	01
Command ==>											
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---											
Help Print Exit Top Posi Bot - + Canc											

On the DBLOG Trace screen for DB2, the following information is displayed:

Item	Explanation
M	Executes the commands EXPLAIN (function code "E") or LISTSQL (function code "L") for the corresponding SQL statement, which is identified by the library name, program name, and line number taken from the debug buffer. <b>Important:</b> Since both commands obtain their information from the Natural system file, unwanted results may occur if the corresponding Natural program has been recataloged after the trace was performed. See the commands LISTSQL and EXPLAIN.
No	Sequence number; the statements are displayed in the sequence in which they were executed.
R	Indicates by an asterisk in front of the corresponding statement that a reselection has been performed; if not, the column is left blank; see also the Concept of the File Server.
SQL Statement	The first 29 characters of the logged SQL statement.
CU	Cursor number.
SN	Internal statement number.
SREF	Statement reference number.
M	Mode ("D" for dynamic or "S" for static).
Typ	Database type (in this case: DB2).
SQLC/W	Either the SQL return code in the SQLCODE field of the SQLCA, or the warning in the SQLWARN0 field of the SQLCA if SQLCODE is 0.
Pgm	Natural program name.
Line	Source code line number.
LV	Program level.

## The Snapshot Function

When you enter function code "S" and the appropriate selection criteria on the DBLOG Menu for DB2, a "snapshot" of the specified statement is made as soon as it occurs.

A snapshot does not interrupt the program flow. The snapped data are displayed only if logging is terminated by the user.

In contrast to the statements displayed with the Trace function, the snapshot statement is shown in its entirety (limited to 13 lines).

Once you have entered function code "E" on the DBLOG menu (or the command TEST DBLOG Q SHOW in the Natural NEXT line), the Snapshot Report screen is displayed as shown in the following example:

### Example of Snapshot Report for SQL Statements:

```

14:09:00          ***** NATURAL Test Utilities *****          1999-09-08
User SAG              - Snapshot Report -                          Library SYSSQL

CU SN M Typ R SQLC/W      Library  Program  Store Clock Value   Line LV CID(Hex)
01 01 D ASQ              SYSSAG   DBTEST   1993/02/08 09:03:33 0050 01 00500101

SQL Statement
SELECT MAKE,MODEL,BODY_TYPE,HORSEPOWER,COLOR,SERIAL_NO,OWNER FROM SAG.AUTOMOBIL
ES WHERE OWNER > 'ID----01' FOR UPDATE OF MAKE

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Print Exit                                     Canc

```

On the DBLOG Snapshot Report screen for DB2, the following information is displayed:

<b>Item</b>	<b>Explanation</b>
CU	Cursor number.
SN	Internal statement number.
M	Mode ("D" for dynamic or "S" for static).
Typ	Database type (in this case: DB2).
R	Indicates by an asterisk in front of the corresponding statement that a reselection has been performed; if not, the column is left blank; see also the Concept of the File Server.
SQLC/W	Either the SQL return code in the SQLCODE field of the SQLCA, or the warning in the SQLWARN0 field of theSQLCA if SQLCODE is 0.
Library	The library where the Natural program with the logged statement was cataloged.
Program	The name of the Natural program which contains the logged statement.
Store Clock Value	The time stamp of the Natural program which contains the logged statement.
Line	The source code line number of the logged statement.
LV	The call level of the Natural program which contains the logged statement.
CID (hex)	The command ID of the logged statement (in hexadecimal format)