DBLOG Snapshot Function

The snapshot function provides detailed information on one particular Adabas command, DL/I call or SQL statement.

This section covers the following topics:

- Snapshot Function for Adabas Commands

- Snapshot Function for DL/I Calls

- Snapshot Function for SQL Statements

Snapshot Function for Adabas Commands

This snapshot function interrupts program execution after executing the first Adabas command that matches the selection criteria specified in the **DBLOG Menu**. The **Snapshot Report** (see the following example screen) generated for the specified Adabas command is displayed immediately after program interruption.

The snapshot function automatically logs *all* Adabas buffers. Therefore, you do not have to mark any of the optional buffers in the **DBLOG Menu** before you start the snapshot function. The default **Snapshot Report** displays the control block (CB), which is either the classic control block (ACB) or the extended Adabas control block (ACBX).

This section covers the following topics:

- Invoking Snapshot Report for Adabas Commands

- Displaying Buffers on Snapshot Report

Invoking Snapshot Report for Adabas Commands

▶ **To invoke the Snapshot Report screen for Adabas commands**

1. In the **DBLOG Menu**, specify an Adabas command and additional criteria, if desired, and enter function code `S`.

   The message `DBLOG snapshot facility started now` is displayed.

2. Execute a Natural program which contains the Adabas command specified in the **DBLOG Menu**.

   The program stops executing and a **Snapshot Report** screen similar to the example below appears:

```
16:36:39                  ***** NATURAL TEST UTILITIES *****                 2006-12-12
                           - Snapshot Report -

Command Code : L3          Command ID   :  ??? 00200101 File Number  : 013C
Response Code:      0       ISN          :            1300
ISN Low Limit: 00000000     ISN Quantity :          0
FB Length    : 0009         RB Length    : 0014        SB Length    : 0008
VB Length    : 0014         IB Length    : 0000        Com. Option 1:
Com. Option 2: V            Additions 1  : AE]?        Additions 2  :  ? ?
Additions 3  :              Additions 4  :
Global FID   : 0000000000000000 Command Time : 00000019 Pgm: SAGTEST  Lin: 0020
Control Block
0000 * 30D5D3F3 00200101 013C0000 00000514 * ?NL3 ?????    ?? * 0000
0010 * 00000000 00000000 00090014 00080014 *        ? ? ? ? * 0010
0020 * 000000E5 C1C5BBCA 40404040 00120014 *    VAE]?     ? ? * 0020
0030 * 00000000 00000000 00000000 00000000 *              * 0030
0040 * 00000000 00000000 00000019 00000000 *          ?   * 0040
0050 * 00000000 00000000 00000000 00000000 *              * 0050
0060 * 00000000 00000000 00000000 00000000 *              * 0060
0070 * 00000000 00000000 00000000 00000000 *              * 0070



Command ===> CB_____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help          Exit CB    FB    RB    -     +     SB    VB    IB    Canc
```

Displaying Buffers on Snapshot Report

The **Snapshot Report** screen shows the control block (CB) by default. If you want to display different Adabas buffers or scroll through a report, choose the appropriate PF key or, in the Command line, enter its equivalent direct command described in this section.

The availability of a PF key depends on the buffer currently displayed. If a buffer extends beyond the screen or contains multiple format/record buffers, PF keys required to scroll through the buffer are provided on the screen.

PF Key	Direct Command	Buffer
PF4	CB	Displays the control block. This is the default.
PF5	FB	Displays the format buffer.
PF6	RB or --	Displays the record buffer (RB) or scrolls (--) to the beginning of a long buffer.
PF7	-	Scrolls up one page in a long buffer.
PF8	+	Scrolls down one page in a long buffer
PF9	SB or ++	Displays the search buffer (SB) or scrolls (++) to the end of a long buffer.
PF10	VB or <	Displays the value buffer (VB). For multiple format/record buffers, shows the previous (<) record/format buffer.
PF11	IB or >	Displays the ISN buffer (IB). For multiple format/record buffers, shows the next (>) record/format buffer.

For information on the fields displayed in a control block or buffer, see *Displaying Adabas Buffers*.

Snapshot Function for DL/I Calls

This snapshot function generates the **Snapshot Report** (see the following example screen) of the first DL/I call that matches the selection criteria specified in the **DBLOG Menu**. A snapshot does not interrupt the program flow. The snapshot data is kept in the Natural DBLOG buffer to be displayed only if the user enters the appropriate DBLOG command as described below.

This section covers the following topics:

- Invoking Snapshot Report for DL/I Calls

- Snapshot Report Information for DL/I Calls

Invoking Snapshot Report for DL/I Calls

▶ **To invoke the Snapshot Report screen for DL/I calls**

1. In the **DBLOG Menu**, specify a DL/I call and additional criteria, if desired, and enter function code S.

The message `DBLOG snapshot facility started now` is displayed.

2. Execute a Natural program which contains the DL/I call specified in the **DBLOG Menu**. (Log data is written to the Natural DBLOG buffer.)

3. Display the snapshot data by entering the following command:

```
TEST DBLOG D
```

Or:
In the **DBLOG Menu**, enter function code `E`.

A **Snapshot Report** screen for DL/I calls similar to the example below appears:

```
16:33:27                    ***** NATURAL DBA Utility *****              2006-12-12
User SAG                       - Snapshot Report -           Library SAG
        DL/I Call ......... : GHNP           Pgm: SAGTEST      Line: 0110
        PSB Name .......... : PBNDL01
        PCB Number ........ : 1                out of 3
        Contents of this PCB                   Contents of NDB/NSB
        Database Name ..... : DNDL01           DBD is physical
        Segment Level ..... : 2                Segment Level .. : 2
        Statuscode ........ : BLANK
        Processing Options  : A
        Segment Name ...... : OFFERING         DBID/FNR ....... : 246   12
                                               Min/Max Length . : 41    41
        Length of KFBA .... : 9
        Number of SENSEGs . : 10  <WARNING>  Number of SENSEGs: 5
        Number of SSAs .... : 2

IOA:    Char :   ?010791DARMSTADT
        Zone : 02FFFFFFCCDDEECCE44444444444444444444444440000000000000000
        Digit: 09010791419423143000000000000000000000000000000000000000000

KFBA:   Char : 004010791
        Zone : FFFFFFFFF00000000000000000000000000000000000000000000000000
        Digit: 00401079100000000000000000000000000000000000000000000000000
MORE

SSA 1 : Char : COURSE  *- (COURSEN  =004)
        Zone : CDEDEC445644CDEDECD447FFF500000000000000000000000000000000
        Digit: 36492500C00D364925500E004D0000000000000000000000000000000

SSA 2 : Char : OFFERING*F-
        Zone : DCCCDCDC5C6400000000000000000000000000000000000000000000000
         Digit: 66659957C60000000000000000000000000000000000000000000000000
```

Snapshot Report Information for DL/I Calls

The following information is provided on the **Snapshot Report** screen for DL/I calls:

- the PSB (Program Specification Block) Name

- the PCB (Program Communication Block) Number

- the PCB mask, which consists of the following:

    ○ DBD (Database Description) including Database Name

    ○ Segment Level number

    ○ Statuscode

    ○ Processing Options

    ○ Segment Name

    ○ Length of KFBA (Key Feedback Area)

    ○ Number of SENSGs (Sensitive Segments)

    ○ KFBA:
      Key Feedback Area

    ○ Number of SSAs (Segment Search Argument)

- all SSAs

- the I/O Area

The first 120 bytes of the Key Feedback Area, of all SSAs (up to 15 SSAs are possible) and of the I/O area are displayed, both in decimal and hexadecimal format.

The DBD Name in the PCB is used to read the corresponding NDB (Natural equivalent of DBD) from the Natural FDIC system file. In this NDB, the segment whose name is given in the PCB is located and its minimum/maximum length and segment level number are displayed. The segment level number should match the number in the PCB. In this way, it is possible to detect inconsistencies between Natural NDBs and DL/I DBDs.

The PSB name is used to read the corresponding NSB (Natural equivalent of PSB) from the Natural FDIC system file. From this NSB, the number of sensitive segments is displayed. This number should match the number in the PCB. In this way, it is possible to detect inconsistencies between Natural NSBs and DL/I PSBs.

The snapshot function checks whether the DL/I DBD/PSB and the Natural NDB/NSB contain the same values in the fields **Level Number** and **Number of SENSEGs**. The same values, however, do not necessarily ensure that the DL/I DBD/PSB and the Natural NDB/NSB are fully consistent.

In the example above, the values in the **Number of SENSEGs** fields are different, because the Natural NATPSB procedure was not executed after the PSB had been changed by the DL/I PSBGEN procedure.

Snapshot Function for SQL Statements

The snapshot function generates the **Snapshot Report** (see the following example screen) of the first SQL statement that matches the selection criteria specified in the **DBLOG Menu**. A snapshot does not interrupt the program flow.

Unlike the statements displayed with the DBLOG trace function, the snapshot shows the statement in its entirety (limited to 13 lines).

The snapshot data is kept in the Natural DBLOG buffer to be displayed only if the user enters the appropriate DBLOG command as described below.

This section covers the following topics:

- Invoking Snapshot Report for SQL Statements

- Snapshot Report Information for SQL Statements

Invoking Snapshot Report for SQL Statements

▶ **To invoke the Snapshot Report screen for SQL statements**

1. In the **DBLOG Menu**, specify an SQL statement and additional criteria, if desired, and enter function code S.

   The message DBLOG snapshot facility started now is displayed.

2. Execute a Natural program which contains the SQL statement specified in the **DBLOG Menu**. (Log data is written to the Natural DBLOG buffer.)

3. Display the snapshot data by entering the following command:

   ```
   TEST DBLOG Q
   ```

   Or:
   In the **DBLOG Menu**, enter function code E.

   A **Snapshot Report** screen for SQL statements similar to the example below appears:

   ```
   10:59:28                 ***** NATURAL Test Utilities *****              2006-12-12
   User SAG                        - Snapshot Report -           Library SAG

   CU SN M Typ R SQLC/W     Library  Program  Store Clock Value   Line LV CID(Hex)
   01 01 D DB2              SAG      SAGTEST  2002/04/03 14:23:06 0150 01 01500101


   SQL Statement
   SELECT EMPNO,FIRSTNME,MIDINIT,LASTNAME,EDLEVEL,SALARY FROM DSN8510.EMP WHERE EM
   PNO > '000300' FOR UPDATE OF EDLEVEL




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

Snapshot Report Information for SQL Statements

The following information is provided on the **Snapshot Report** screen for SQL statements:

Column	Explanation
CU	Cursor number.
SN	Internal statement number.
M	Mode: D for dynamic or S for static.
Typ	Database type: DB2 or SQL/DS.
R	Only applicable if the Natural File Server for DB2 is in use. 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 *Concept of the File Server* in the *Natural for DB2* documentation.
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.
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.