

CALLDLI Test Program - DAZZLER

This chapter covers the following topics:

- Introduction
 - DL/I Statements
 - Sets
 - DAZZLER Control Statements
 - z/OS JCL Requirements for DAZZLER
 - z/VSE JCS Requirements for DAZZLER
-

Introduction

DAZZLER is a special test utility provided with the Adabas Bridge for DL/I. It is used to format DL/I calls and to monitor their results. This may be necessary, for example, to compare DL/I and Adabas performance.

DAZZLER handles up to the following maximum values:

Number of PCBs	10
Number of SSAs	15
I/O area length (bytes)	256
SSA length (bytes)	256

You are most likely to want to use DAZZLER for low volume input, and for this reason the eight possible statement types have been divided into two groups of four. The first group contains those statements necessary for creating DL/I calls, and are referred to as DL/I statements. The second group consists of statements provided for the user's benefit, particularly when handling medium-to-large volumes of input. The statements in this second group are known as DAZZLER control statements, and are all optional.

All statements in both groups contain a number of validated fields. The validation rules for each field are explained in the discussion of the individual statements.

You should note that the first three positions of each statement contain an identifying title. Invalid titles, and thus the information on the statement concerned, are immediately rejected.

Blank statements are ignored.

Note:

All PSBs used for the DAZZLER program must have been created with either the PSBGEN statement option LANG=ASSEM or LANG=COBOL. Alternatively, if the PSB had been created with LANG=PLI, you must specify LANG=COBOL as DAZIFP parameter.

Appendix A shows an example of DAZZLER usage.

DL/I Statements

The following table lists the DL/I statements.

Statement	Function
HEADER Statement	A compulsory statement which contains the DL/I call and other important related parameters. It must be the first statement within the set (see below).
IOA Statement	An optional "input/output area" statement which, when present, causes the existing I/O area to be overwritten.
SSA Statement	An optional "segment search argument" statement which, when present, causes a specified segment search argument to be overwritten.
SUMMARY Statement	An optional statement which, when present, produces a tabular summary of all segment names and how many occurrences there are in a given PCB.

Sets

Each individual DL/I call requires that a set of statements be input — a set being defined as a group of statements beginning with a HEADER statement and ending with either the statement preceding the next HEADER or SUMMARY DL/I statement, JUMP DAZZLER control statement, or with the end of the file, as appropriate. A SUMMARY statement is considered to be a set in itself.

In other words, a set can consist of:

- only a HEADER statement,
- a HEADER statement followed by any combination of other DL/I or DAZZLER control statements (any COMMENT statements encountered will be ignored), or
- only a SUMMARY statement.

If the HEADER statement of a set is rejected by the validation routines, all the remaining statements in the same set will be rejected as well.

The following pages describe the individual statements, their layouts and the validation routines appropriate in each case.

HEADER Statement

```
.....1.....2.....3.....4.....5.....6.....+
*HD call nnn cc pp ss iiiii +++++ ssslssslssslssslssslssslsssl
```

The following pages explain the individual entries on the statement in detail.

Statement Field	Position	Status*	Description
*HD	1-3	C	The identifier for the HEADER statement.
call	5-8	C	The kind of DL/I call. This is validated against an internal table and must be four characters long, plus trailing blanks if appropriate. e.g. "DLET", "REPL", "GU", "GHNP", etc.
nnn	10-12	O	The number of times the set is to be run. The range is 1-999, with leading zeros as appropriate. The default is 001. See also the "cc" field.
cc	14-15	O	Any non-blank entry overrides any value entered in the "nnn" field. The set is repeated until the status code returned by DL/I corresponds exactly to the entry made here. An entry of "??" instructs DAZZLER to stop repeating the call as soon as a non-blank status code is returned.
pp	17-18	C	The PCB number to be used for this call.
ss	20-21	O	The number of SSAs in the call. The range is 00 to 15 (a leading zero must be provided if necessary). The default is 00.
iiii	23-27	O	If this parameter is omitted, or if "00000" is entered, no further validation is performed and any further coding in this statement is treated as a printable comment. If a five-figure number (including leading zeros) is entered, this value is used as the initial value for an iterative overwrite of the IOA. All remaining fields are then used to specify this overwrite.
+++++	29-33	O	A five-figure number with leading zeros which will be used as the increment for the iterative overwrite initiated by the entry in "iiii". 00001 is the default.
sssl...etc	35-66	O/C **	Each occurrence of "sssl" represents the start position (SSS) and the length ("l") of the iterative overwrite of the IOA. The "sss" entry must include leading zeros if necessary and be in the range 1-256. The "l" entry must be in the range 1-5. In addition, the end position calculable from the two entries must be less than 257. A minimum of 1 and a maximum of 8 "sssl" entries may be provided. Blank entries are ignored.

* C = compulsory, O = optional

** Optional unless an entry has been made in "iiii", in which case compulsory.

Example of the Iterative Overwrite Facility

Where *nnn* =004, *iiii* contains 00100, +++++ contains 00035, and the following *sssl* entries are provided

```

ssslssslssslssslssslssslssslsssl
+....4....+....5....+....6....+
02040352    0615    0013    1111
    
```

then the results of the coding are:

IOA Positions	1st Iteration	2nd Iteration	3rd Iteration	4th Iteration
20,21,22,23	0100	0135	0170	0205
35,36	00	35	70	05
61,62,63,64,65	00100	00135	00170	00205
1,2,3	100	135	170	205
111	0	5	0	5

BACK Call

In addition to DL/I calls you can specify BACK as the call in the HEADER statement. This results in an Adabas BT (Backout Transaction) call which removes all data base modifications performed during the user's current logical transaction. The BACK call can only be used, if DAZZLER runs as an online batch application, i.e. with BMP, MPS or SDB as first DAZIFP parameter. To prevent conflicts with the automatic ET functionality, it is recommended to specify ET=NO as DAZIFP parameter or in the ADL Parameter Module. The BACK call can not be used while DAZZLER runs against DL/I.

IOA and SSA Statements

```
.....1.....2.....3.....4.....5.....6.....+
```

```
tit form sss ll ddddddddddddddddddddddddddddddddddddddddddddddd
```

Statement Field	Position	Status*	Description
tit	1-3	C	Two settings are possible: IOA - Indicates that the I/O area is to be overwritten. Snn - Indicates that an SSA is to be overwritten. The value "nn" specifies the SSA concerned and must be in the range 1-15, with a leading zero if necessary. The entry in "nn" is validated for being not greater than the number of SSAs coded on the HEADER statement.
form	5-8	O	A four-character entry with a trailing space if necessary. Indicates the format for the rewrite to the program. The possible values are: CHAR Characters HEX Hexadecimal data PCK Positive packed decimal number PCK+ Positive packed decimal number PCK- Negative packed decimal number The default is CHAR.
sss	10-12	C	A three-figure number with leading zeros, if necessary, which specifies the start position for the overwrite. It must be in the range 1-256.
ll	14-15	C	Two-digit string, with a leading zero if necessary, specifying the length of the overwrite. It is dependent on the entry made under "form" . If this specifies character format, the "ll" entry determines the number of bytes to be overwritten. If it specifies non-character format, it specifies the number of digits to be read from the statement. The entry must be in the range 1-50, except for packed decimal data, in which case the maximum is 18. The calculable end position for the overwrite must be less than 257.
dddd ...	17-66	C	The data to be used in the overwrite are entered here, left-justified. Overwrites in hexadecimal format cause the program to validate the data for being in the ranges 0-9 or A-F and for being an even number of digits. Packed decimal format causes validation for the range 0-9.

* C = compulsory, O = optional

The I/O area and the SSAs are only overwritten when this is specifically requested. If a subsequent call requires some or all of the same SSAs or the same data in the I/O area, it is not necessary for these to be recoded. You should note that the I/O area may well change as the result of a DL/I call.

SUMMARY Statement

....+.1....

SUMMARY pp

Statement Field	Position	Status	Description
pp	9-10	Compulsory	The required PCB number. The range is 01-10. A leading zero must be provided if necessary.

This statement causes the program to produce a table showing, for any one PCB, all the retrievable segment names and how many occurrences of each there are. The statement is NOT a part of a set as defined earlier, but rather can be considered a set in itself. When a SUMMARY statement has been encountered and validated as correct, the program issues an unqualified GU to establish position on the first segment and then issues successive GN calls until a status code of "GB" is returned. The table is then printed out and the program continues with the next statement. A SUMMARY statement may appear anywhere in the job stream and will also be recognized as ending a previous HEADER statement set.

DAZZLER Control Statements

Statement	Function
PRINT CONTROL PARAMETER	An optional statement which provides run control parameters. It comes at the front of the input stream.
COM(MENT)	This statement enables users to make comments which will not subsequently be printed. It can appear anywhere in the job stream, and is ignored as soon as it is encountered. It is used to help understand the test stream.
JUMP	This statement tells DAZZLER to ignore all following statements until the corresponding LABEL statement is read. It allows users to jump down the stream and skip some or all statements.
LABEL	This statement is the partner of a JUMP statement. It places a label in the job stream to indicate the point at which the program is to restart processing statements. If no prior associated JUMP statement exists, the LABEL statement is ignored.

PRINT CONTROL PARAMETER Statement

If the PRINT CONTROL PARAMETER statement is omitted, dates are printed with the format "DD/MM/YY" and a page throw occurs before each new set of statements.

The layout of the statement is as follows:

....+.1....+....

PARddtccsssssppee

Statement Field	Position	Status*	Description
PAR	1-3	C	The identifier for the statement.
dd	4-5	C	The date print field. Two values are possible: US - Dates are printed in the format MM/DD/YY. AM - Dates are printed in the format DD/MM/YY. If the PRINT CONTROL PARAMETER statement is not used, then the date will automatically have the format DD/MM/YY.
t	6	O	If an entry is made here, the automatic page throw for every new call is suppressed.
cc	7-8	O	Specifies what is to be printed. Permissible entries are: PA Print all. PN Print no results from the calls. PF Print from a specified call number onwards. PC Print no results until a non- blank status code is returned: after this, print the number of calls specified by the "pp" field (see below). PS Print slots (e.g. print every 20th call). The default is PA.
sssss	9-13	O	Specifies the start call number when PF or PS have been entered in the "cc" field. If entered, the start call number must be a five-figure number with leading zeros if necessary.
pp	14-15	O	Two uses are possible: Specifies the number of calls to be printed after a PC or a PF. The value entered must be a two-figure number with a leading zero if necessary. An entry of 99 means that all remaining calls will be printed. For the PS parameter, specifies the number of calls either side of the selected call number to be printed.
eee	16-18	O	Used only with the PS parameter. Specifies the number of calls to be skipped when slots are to be printed. It is the repeating increment and must be a three-figure number (including leading zeros where necessary).

* C = compulsory, O = optional

COMMENT Statement

```
....+....1....+...
```

```
COM
```

Positions 1-3 must contain COM. This statement will be ignored by the program and is provided so that users can increase the legibility of their test streams.

JUMP and LABEL Statements

```
....+....
```

```
JUMP 111
LABEL111
```

This pair of statements is provided so that users can ignore calls in the test stream. As soon as the program reads the JUMP statement ("JUMP" in columns 1-5), it ignores all subsequent statements until the matching LABEL statement ("LABEL" in columns 1-5) is found. The match is performed on columns 6-8 (inclusive) of both statements. The three characters must match exactly for the program to restart. If no LABEL statement exists for a JUMP statement, no further calls will take place. If no JUMP statement exists for a LABEL statement, then the latter is ignored. Note that a JUMP statement forces an end of set condition (see earlier in this section).

z/OS JCL Requirements for DAZZLER

The following table lists the data sets used by the Test utility, DAZZLER.

DDname	Medium	Description
CFILE	Reader	Control input for DAZZLER.
PRNTR	Printer	Report.

Example

```
//          EXEC PGM=DAZIFP,PARM='DLI,DAZZLER,psbname'
//STEPLIB DD DSN=ADL.LOAD,DISP=SHR
//          DD DSN=ADABAS.LOAD,DISP=SHR
//DDCARD  DD *
ADARUN PROGRAM=USER,...
//DAZOUT1 DD SYSOUT=X
//*
//*  DAZZLER DATASETS
//*
//PRNTR   DD SYSOUT=X
//CFILE   DD *
*HD GU           01
*HD GN           GB 01
/*
```


z/VSE JCS Requirements for DAZZLER

The following table lists the files used by the Test utility, DAZZLER.

DTF	Logical Unit	Medium	Description
CFILE	SYS007	Reader	Control input for DAZZLER.
PRNTR	SYS006	Printer	Report.

The control input for the batch monitor (DAZIFP) and for ADARUN is read from SYSIPT. Where the control input for DAZZLER itself is also to be read from SYSIPT (by assigning SYS007 to SYSIPT), the control statements must be specified in the following order:

```
DLI,DAZZLER,psbname,...                               input for DAZIFP
/*
ADARUN DB=dbid,MO=MULTI,PROGRAM=USER,...             input for ADARUN
/*
*HD GU ....                                         input for DAZZLER
.
.
/*
```

Example

```
// ASSGN SYS006,SYSLST
// ASSGN SYS007,SYSIPT
// EXEC DAZIFP
DLI,DAZZLER,psbname
/*
ADARUN PROGRAM=USER,...
/*
*HD GU          01
*HD GN          GB 01
/*
```