J E S 2 J O B L O G

```
18.01.10 JOB  145  IEF677I WARNING MESSAGE(S) FOR JOB VSTESTE2 ISSUED
18.01.10 JOB  145  $HASP373 VSTESTE2 STARTED - INIT  1 - CLASS A - SYS HMVS
18.01.10 JOB  145  IEF403I VSTESTE2 - STARTED - TIME=18.01.10
18.01.11 JOB  145  CCI001C PL1L    /IEMAA   /00:00:00.16/        /00004/SYS     /VSTESTE2
18.01.11 JOB  145  CCI001C LKED    /IEWL    /00:00:00.05/        /00000/SYS     /VSTESTE2
18.01.11 JOB  145  CCI001C GO      /PGM=*.DD/00:00:00.02/        /00000/SYS     /VSTESTE2
18.01.11 JOB  145  IEF404I VSTESTE2 - ENDED - TIME=18.01.11
18.01.11 JOB  145  $HASP395 VSTESTE2 ENDED

------ JES2 JOB STATISTICS ------

 07 JUL 20 JOB EXECUTION DATE

       23 CARDS READ

    1,043 SYSOUT PRINT RECORDS

        0 SYSOUT PUNCH RECORDS

     0.00 MINUTES EXECUTION TIME
```

```
 1      //VSTESTE2 JOB (SYS),'VSAMIOP IVP ESDSLOAD',CLASS=A,MSGCLASS=X,         JOB  145
        //             REGION=4096K
        ***
        ********************************************************************
        *** PL/1 MODULE: ESDSLOAD     VSAM DATASET: VSTESTES.CLUSTER (ESDS)
        ***
        *** SEQUENTIALLY LOADS ENTRY SEQUENCED DATASET
        ********************************************************************
        ***
 2      //PL1F    EXEC PL1LFCLG,
        //             PARM='LOAD,NODECK,ATR,XREF,CHAR60,MACRO'
 3      XXPL1L   EXEC  PGM=IEMAA,PARM='LOAD,NODECK',REGION=52K                  00000100
 4      XXSTEPLIB  DD DSN=SYSC.LINKLIB,DISP=SHR                                 00000200
 5      //PL1L.SYSPRINT DD SYSOUT=*
        X/SYSPRINT DD  SYSOUT=A                                                 00000300
 6      XXSYSLIN   DD  DSNAME=&&LOADSET,DISP=(MOD,PASS),UNIT=SYSSQ,            *00000400
        XX             SPACE=(80,(250,100))                                     00000500
 7      XXSYSUT3  DD  DSNAME=&&SYSUT3,UNIT=SYSDA,SPACE=(80,(250,250)),         *00000600
        XX             DCB=BLKSIZE=80                                           00000700
 8      XXSYSUT1   DD  DSNAME=&&SYSUT1,UNIT=SYSDA,SPACE=(1024,(60,60),,CONTIG),*00000800
        XX             SEP=(SYSUT3,SYSLIN),DCB=BLKSIZE=1024                     00000900
 9      //PL1L.SYSLIB DD DSN=SYSC.VSAMIOP.MACLIB,DISP=SHR
10      //PL1L.SYSIN DD  DSN=SYSC.VSAMIOP.SOURCE(ESDSLOAD),DISP=SHR
11      XXLKED   EXEC  PGM=IEWL,PARM='XREF,LIST',COND=(9,LT,PL1L),              *00001000
        XX             REGION=96K                                               00001100
12      //LKED.SYSLIB DD
        X/SYSLIB   DD  DSNAME=SYSC.PL1LIB,DISP=SHR                              00001201
13      //         DD  DSN=SYSC.LINKLIB,DISP=SHR
14      XXSYSLMOD  DD  DSNAME=&&GOSET(GO),DISP=(MOD,PASS),                      *00001300
        XX             UNIT=SYSDA,SPACE=(1024,(50,20,1),RLSE)                   00001400
15      XXSYSUT1   DD  DSNAME=&&SYSUT1,UNIT=SYSDA,SPACE=(1024,(200,20)),        *00001500
        XX             SEP=(SYSLMOD,SYSLIB),DCB=BLKSIZE=1024                    00001600
16      //LKED.SYSPRINT DD SYSOUT=*
        X/SYSPRINT DD  SYSOUT=A                                                 00001700
17      XXSYSLIN   DD  DSNAME=&&LOADSET,DISP=(OLD,DELETE)                       00001800
18      XX         DD  DDNAME=SYSIN                                             00001900
19      XXGO     EXEC  PGM=*.LKED.SYSLMOD,COND=((9,LT,LKED),(9,LT,PL1L))       00002000
20      //GO.STEPLIB DD DSN=SYSC.PL1LIB,DISP=SHR
        X/STEPLIB  DD  DSN=SYSC.LINKLIB,DISP=SHR                                00002102
21      XX         DD  DSN=SYSC.PL1LIB,DISP=SHR                                 00002202
22      XXSYSPRINT DD  SYSOUT=A                                                 00002300
23      //GO.PRINTR DD SYSOUT=*
24      //GO.IMAGES DD DSN=PUB001.VSAMTEST.DATA,DISP=SHR
25      //GO.SYSUDUMP DD SYSOUT=*
26      //GO.SYSPRINT DD SYSOUT=*
27      //GO.ESDSF01 DD DSN=PUB001.VSTESTES.CLUSTER,DISP=OLD
```

```
 STMT NO. MESSAGE
-
   19     IEF686I DDNAME REFERRED TO ON DDNAME KEYWORD IN PRIOR STEP WAS NOT RESOLVED
IEF236I ALLOC. FOR VSTESTE2 PL1L PL1F
IEF237I 253  ALLOCATED TO STEPLIB
IEF237I 253  ALLOCATED TO SYS00294
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 380  ALLOCATED TO SYSLIN
IEF237I 251  ALLOCATED TO SYSUT3
IEF237I 370  ALLOCATED TO SYSUT1
IEF237I 253  ALLOCATED TO SYSLIB
IEF237I 253  ALLOCATED TO SYSIN
IEF142I VSTESTE2 PL1L PL1F - STEP WAS EXECUTED - COND CODE 0004
IEF285I   SYSC.LINKLIB                                KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   UCSYSCPK                                    KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   JES2.JOB00145.SO0101                        SYSOUT
IEF285I   SYS20189.T180110.RA000.VSTESTE2.LOADSET      PASSED        *------210
IEF285I   VOL SER NOS= MVS380.
IEF285I   SYS20189.T180110.RA000.VSTESTE2.SYSUT3       DELETED       *------261
IEF285I   VOL SER NOS= WORK00.
IEF285I   SYS20189.T180110.RA000.VSTESTE2.SYSUT1       DELETED       *--------0
IEF285I   VOL SER NOS= MVS370.
IEF285I   SYSC.VSAMIOP.MACLIB                         KEPT          *-------27
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   SYSC.VSAMIOP.SOURCE                         KEPT          *--------3
IEF285I   VOL SER NOS= SYSCPK.
IEF373I STEP /PL1L    / START 20189.1801
IEF374I STEP /PL1L    / STOP  20189.1801 CPU    0MIN 00.16SEC SRB    0MIN 00.04SEC VIRT  4096K SYS   212K
**** JOBCARD READ 20189 18:01:10 ************************************************************************************
*                                      PRC-CCI  370/148 VS2 R03.8  HMVS  STEP STATISTICS                              *
*  STEP NAME  PL1L        USER CORE        4096K   TAPES USED/IO 000/000000000   START   TIME  18:01:10    TCB TIME  00:00:00.16  *
*  PGM  NAME  IEMAA       SYSTEM CORE       212K   DISKS USED/IO 004/000000501   STOP    TIME  18:01:11    SRB TIME  00:00:00.04  *
*   COND CODE    0004     PRIVATE AREA SZ   4096K       ALLOC TIME  18:01:10      ELAPSED TIME                PGM LOAD  18:01:10   *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
*   004          2717   00:00:00.23            0             0           0             0              0             0            0  *
************************************************************************************************************************
*  CPU $ (     0.05) + EXCP $ (     0.67) + MEMORY $ (     1.86) = TOTAL $ (     2.58)                                  *
************************************************************************************************************************
IEF236I ALLOC. FOR VSTESTE2 LKED PL1F
IEF237I 253  ALLOCATED TO SYSLIB
IEF237I 253  ALLOCATED TO
IEF237I 253  ALLOCATED TO SYS00296
IEF237I 251  ALLOCATED TO SYSLMOD
IEF237I 370  ALLOCATED TO SYSUT1
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 380  ALLOCATED TO SYSLIN
IEF237I DMY  ALLOCATED TO
IEF142I VSTESTE2 LKED PL1F - STEP WAS EXECUTED - COND CODE 0000
IEF285I   SYSC.PL1LIB                                 KEPT          *------106
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   SYSC.LINKLIB                                KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   UCSYSCPK                                    KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   SYS20189.T180110.RA000.VSTESTE2.GOSET        PASSED        *-------57
IEF285I   VOL SER NOS= WORK00.
IEF285I   SYS20189.T180110.RA000.VSTESTE2.SYSUT1       DELETED       *--------0
IEF285I   VOL SER NOS= MVS370.
IEF285I   JES2.JOB00145.SO0102                        SYSOUT
```

```
IEF285I   SYS20189.T180110.RA000.VSTESTE2.LOADSET      DELETED       *------211
IEF285I   VOL SER NOS= MVS380.
IEF373I STEP /LKED    / START 20189.1801
IEF374I STEP /LKED    / STOP  20189.1801 CPU    0MIN 00.05SEC SRB    0MIN 00.01SEC VIRT   260K SYS   168K
********************************************************************************************************
*                                      PRC-CCI  370/148 VS2 R03.8  HMVS  STEP STATISTICS                 *
*  STEP NAME  LKED        USER CORE          260K   TAPES USED/IO 000/000000000   START   TIME  18:01:11    TCB TIME  00:00:00.05  *
*  PGM  NAME  IEWL        SYSTEM CORE        168K   DISKS USED/IO 004/000000374   STOP    TIME  18:01:11    SRB TIME  00:00:00.01  *
*   COND CODE    0000     PRIVATE AREA SZ   4096K       ALLOC TIME  18:01:11      ELAPSED TIME                PGM LOAD  18:01:11   *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
*   004          1916   00:00:00.08            0             0         0              0              0             0  *
********************************************************************************************************
*  CPU $ (     0.01) + EXCP $ (     0.50) + MEMORY $ (     0.03) = TOTAL $ (     0.54)                    *
********************************************************************************************************
IEF236I ALLOC. FOR VSTESTE2 GO PL1F
IEF237I 251  ALLOCATED TO PGM=*.DD
IEF237I 253  ALLOCATED TO STEPLIB
IEF237I 253  ALLOCATED TO
IEF237I 253  ALLOCATED TO SYS00298
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I JES2 ALLOCATED TO PRINTR
IEF237I 190  ALLOCATED TO IMAGES
IEF237I 190  ALLOCATED TO SYS00300
IEF237I JES2 ALLOCATED TO SYSUDUMP
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 190  ALLOCATED TO ESDSF01
IEF142I VSTESTE2 GO PL1F - STEP WAS EXECUTED - COND CODE 0000
IEF285I   SYS20189.T180110.RA000.VSTESTE2.GOSET        KEPT          *--------0
IEF285I   VOL SER NOS= WORK00.
IEF285I   SYSC.PL1LIB                                  KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   SYSC.PL1LIB                                  KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   UCSYSCPK                                     KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   JES2.JOB00145.SO0103                         SYSOUT
IEF285I   JES2.JOB00145.SO0104                         SYSOUT
IEF285I   PUB001.VSAMTEST.DATA                         KEPT          *-------11
IEF285I   VOL SER NOS= PUB001.
IEF285I   UCPUB001                                     KEPT          *--------0
IEF285I   VOL SER NOS= PUB001.
IEF285I   JES2.JOB00145.SO0105                         SYSOUT
IEF285I   JES2.JOB00145.SO0106                         SYSOUT
IEF285I   PUB001.VSTESTES.CLUSTER                      KEPT          *--------2
IEF285I   VOL SER NOS= PUB001.
IEF373I STEP /GO      / START 20189.1801
IEF374I STEP /GO      / STOP  20189.1801 CPU    0MIN 00.02SEC SRB    0MIN 00.00SEC VIRT   100K SYS   180K
********************************************************************************************************
*                                      PRC-CCI  370/148 VS2 R03.8  HMVS  STEP STATISTICS                 *
*  STEP NAME  GO          USER CORE          100K   TAPES USED/IO 000/000000000   START   TIME  18:01:11    TCB TIME  00:00:00.02  *
*  PGM  NAME  PGM=*.DD    SYSTEM CORE        180K   DISKS USED/IO 003/000000013   STOP    TIME  18:01:11    SRB TIME  00:00:00.00  *
*   COND CODE    0000     PRIVATE AREA SZ   4096K       ALLOC TIME  18:01:11      ELAPSED TIME                PGM LOAD  18:01:11   *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
*   004            92   00:00:00.02            0             0         0              0              0             0  *
********************************************************************************************************
*  CPU $ (     0.00) + EXCP $ (     0.01) + MEMORY $ (     0.00) = TOTAL $ (     0.01)                    *
********************************************************************************************************
IEF237I 251  ALLOCATED TO SYS00001
IEF285I   SYS20189.T180111.RA000.VSTESTE2.R0000001     KEPT          *--------0
IEF285I   VOL SER NOS= WORK00.
IEF285I   SYS20189.T180110.RA000.VSTESTE2.GOSET        DELETED
```

```
IEF285I   VOL SER NOS= WORK00.
IEF375I  JOB /VSTESTE2/ START 20189.1801
IEF376I  JOB /VSTESTE2/ STOP  20189.1801 CPU    0MIN 00.23SEC SRB    0MIN 00.05SEC
```

PL/I F COMPILER OPTIONS SPECIFIED ARE AS FOLLOWS--

```
     LOAD,NODECK,ATR,XREF,CHAR60,MACRO
```

THE COMPLETE LIST OF OPTIONS USED DURING THIS COMPILATION IS--

```
  EBCDIC
  CHAR60
  MACRO
  SOURCE2
NOMACDCK
  COMP
  SOURCE
  ATR
  XREF
NOEXTREF
NOLIST
  LOAD
NODECK
  FLAGW
NOSTMT
  SIZE=4154608
  LINECNT=050
  OPT=01
  SORMGIN=(002,072)
NOEXTDIC
NONEST
OPLIST
SYNCHKT
```

```
*OPTIONS IN EFFECT*     EBCDIC,CHAR60,MACRO,SOURCE2,NOMACDCK,COMP,SOURCE,ATR,XREF,NOEXTREF,NOLIST,LOAD,
*OPTIONS IN EFFECT*     NODECK,FLAGW,NOSTMT,SIZE=4154608,LINECNT=050,OPT=01,SORMGIN=(002,072),NOEXTDIC,
*OPTIONS IN EFFECT*     NONEST,OPLIST,SYNCHKT
```

COMPILE-TIME MACRO PROCESSOR
MACRO SOURCE2 LISTING

```
  1    /*****************************************************************01770000
  2                                                                      01780000
  3      ESDSLOAD - TESTS THE VSAMIO ROUTINE BY LOADING AN ESDS CLUSTER WITH  01790000
  4                 RECORDS FROM A SEQUENTIAL DATASET.                   01800000
  5                                                                      01810000
  6     ******************************************************************/01820000
  7    ESDSLOD:                                                          01830000
  8       PROCEDURE OPTIONS(MAIN);                                       01840000
  9                                                                      01850000
 10          ON ERROR                                                   01860000
 11             BEGIN;                                                  01870000
 12                ON ERROR SYSTEM;                                     01880000
 13                PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');     01890000
 14                PUT SKIP DATA;                                       01900000
 15                PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');     01910000
 16             END;                                                    01920000
 17                                                                     01930000
 18          OPEN                                                       01940000
 19             FILE(IMAGES),                                           01950000
 20             FILE(PRINTR) LINESIZE(121);                             01960000
 21                                                                     01970000
 22          ON ENDFILE(IMAGES)                                         01980000
 23             MORE_RECORDS = NO;                                      01990000
 24                                                                     02000000
 25          PRINT_AREA = 'ESDSLOAD: WRITE ESDS SEQUENTIALLY';          02010000
 26          WRITE FILE(PRINTR) FROM(PRINT_LINE);                       02020000
 27          PRINT_AREA = '-------------------------------';            02030000
 28          WRITE FILE(PRINTR) FROM(PRINT_LINE);                       02040000
 29          PRINT_AREA = ' ';                                          02050000
 30          WRITE FILE(PRINTR) FROM(PRINT_LINE);                       02060000
 31                                                                     02070000
 32          MORE_RECORDS = YES;                                        02080000
 33                                                                     02090000
 34     /*****************************************************************02100000
 35       ESTABLISH PARAMETERS FOR VSAM DATASET AND CALL ROUTINE TO OPEN    02110000
 36      ******************************************************************/02120000
 37          VSFB_DDNAME = 'ESDSF01';                                   02130000
 38          VSFB_ORGANIZATION = VSIO_ESDS;                             02140000
 39          VSFB_ACCESS = VSIO_SEQUENTIAL;                             02150000
 40          VSFB_MODE = VSIO_OUTPUT;                                   02160000
 41          VSFB_RECORD_LENGTH = 80;                                   02170000
 42          VSFB_KEY_POSITION = 0;                                     02180000
 43          VSFB_KEY_LENGTH = 0;                                       02190000
 44          VSIO_COMMAND = VSIO_OPEN;                                  02200000
```

MACRO SOURCE2 LISTING

```
 45          CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                          02210000
 46                        VSIO_FILE_BLOCK,                               02220000
 47                        RECORD_IMAGE);                                 02230000
 48          IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN                02240000
 49             DO;                                                       02250000
 50                CALL VSIO_ERROR;                                       02260000
 51                RETURN;                                                02270000
 52             END;                                                      02280000
 53                                                                       02290000
 54          DO WHILE(MORE_RECORDS);                                      02300000
 55             READ FILE(IMAGES) INTO(RECORD_IMAGE);                     02310000
 56             IF (MORE_RECORDS) THEN                                    02320000
 57                CALL WRITE_ES;                                         02330000
 58          END;                                                         02340000
 59                                                                       02350000
 60          CLOSE FILE(IMAGES);                                          02360000
 61                                                                       02370000
 62     /*****************************************************************02380000
 63       CALL ROUTINE TO CLOSE VSAM DATASET                              02390000
 64      *****************************************************************/02400000
 65          VSIO_COMMAND = VSIO_CLOSE;                                   02410000
 66          CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                          02420000
 67                        VSIO_FILE_BLOCK,                               02430000
 68                        RECORD_IMAGE);                                 02440000
 69          IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN                02450000
 70             CALL VSIO_ERROR;                                          02460000
 71                                                                       02470000
 72          COUNTER_EDIT = RECORD_COUNTER;                               02480000
 73          PRINT_AREA = COUNTER_EDIT ||                                 02490000
 74                       ' RECORDS WERE LOADED SUCCESSFULLY';            02500000
 75          WRITE FILE(PRINTR) FROM(PRINT_LINE);                         02510000
 76                                                                       02520000
 77          RETURN;                                                      02530000
 78                                                                       02540000
 79     WRITE_ES:                                                         02550000
 80        PROCEDURE;                                                     02560000
 81                                                                       02570000
 82     /*****************************************************************02580000
 83       CALL ROUTINE TO WRITE RECORD INTO VSAM DATASET                  02590000
 84      *****************************************************************/02600000
 85          VSIO_COMMAND = VSIO_WRITE;                                   02610000
 86          CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                          02620000
 87                        VSIO_FILE_BLOCK,                               02630000
 88                        RECORD_IMAGE);                                 02640000
 89          IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN                02650000
```

MACRO SOURCE2 LISTING

```
 90            CALL VSIO_ERROR;                                          02660000
 91         ELSE                                                         02670000
 92            RECORD_COUNTER = RECORD_COUNTER + 1;                      02680000
 93                                                                      02690000
 94         RETURN;                                                      02700000
 95                                                                      02710000
 96       END WRITE_ES;                                                  02720000
 97                                                                      02730000
 98    VSIO_ERROR:                                                       02740000
 99       PROCEDURE;                                                     02750000
100         PRINT_AREA = 'VSAMIO ERROR OCCURRED DURING ' ||             02760000
101                      VSIO_COMMAND;                                  02770000
102         WRITE FILE(PRINTR) FROM(PRINT_LINE);                        02780000
103         PRINT_AREA = 'VSIO_RETURN_CODE = ' ||                       02790000
104                      VSIO_RETURN_CODE;                              02800000
105         WRITE FILE(PRINTR) FROM(PRINT_LINE);                        02810000
106         PRINT_AREA = 'VSIO_VSAM_RETURN_CODE = ' ||                  02820000
107                      VSIO_VSAM_RETURN_CODE;                         02830000
108         WRITE FILE(PRINTR) FROM(PRINT_LINE);                        02840000
109         PRINT_AREA = 'VSIO_VSAM_FUNCTION_CODE = ' ||                02850000
110                      VSIO_VSAM_FUNCTION_CODE;                       02860000
111         WRITE FILE(PRINTR) FROM(PRINT_LINE);                        02870000
112         PRINT_AREA = 'VSIO_VSAM_FEEDBACK_CODE = ' ||                02880000
113                      VSIO_VSAM_FEEDBACK_CODE;                       02890000
114         WRITE FILE(PRINTR) FROM(PRINT_LINE);                        02900000
115         PRINT_AREA = ' ';                                           02910000
116                                                                     02920000
117         RETURN;                                                     02930000
118                                                                     02940000
119      END VSIO_ERROR;                                                02950000
120                                                                     02960000
121      DECLARE                                                        02970000
122         IMAGES FILE INPUT RECORD SEQUENTIAL EXTERNAL                02980000
123            ENV(F),                                                  02990000
124         PRINTR FILE OUTPUT RECORD SEQUENTIAL EXTERNAL               03000000
125            ENV(F CTLASA);                                           03010000
126                                                                     03020000
127      DECLARE                                                        03030000
128         COUNTER_EDIT                  PICTURE 'ZZ,ZZZ,ZZ9V',      03040000
129         MORE_RECORDS                  BIT(1),                     03050000
130         NO                            BIT(1)    INIT('0'B),       03060000
131         RECORD_COUNTER                FIXED BINARY(15,0),         03070000
132         YES                           BIT(1)    INIT('1'B);       03080000
133                                                                     03090000
134      DECLARE                                                        03100000
```

MACRO SOURCE2 LISTING

```
135          1 RECORD_IMAGE,                                                03110000
136             2 RECORD_FIELDS              CHAR(80);                      03120000
137                                                                         03130000
138       DECLARE                                                           03140000
139          1 PRINT_LINE,                                                  03150000
140             2 CARRIAGE_CONTROL           CHAR(1)   INIT(' '),           03160000
141             2 PRINT_AREA                 CHAR(120);                     03170000
142                                                                         03180000
143    %INCLUDE (VSAMIO);                                                   03190000
144    %INCLUDE (VSAMIOFB);                                                 03200000
145                                                                         03210000
146       END ESDSLOD;                                                      03220000
```

INCLUDED TEXT FOLLOWS FROM DD.MEMBER = SYSLIB .VSAMIO

```
147    /*****************************************************************31100000
148                                                                        31110000
149            VV   VV    SSSSS       A      M     M    IIII      OOOOO       31120000
150            VV   VV   SS   SS     AAA     MM   MM     II      OO   OO      31130000
151            VV   VV   SS         AA AA    MMM MMM     II      OO   OO      31140000
152            VV   VV    SSSSS    AA   AA   MMMMMMM     II      OO   OO      31150000
153            VV   VV        SS   AA   AA   MM M MM     II      OO   OO      31160000
154             VV VV    SS   SS   AAAAAAA   MM   MM     II      OO   OO      31170000
155              VVV     SS   SS   AA   AA   MM   MM     II      OO   OO      31180000
156               V       SSSSS    AA   AA   MM   MM    IIII      OOOOO       31190000
157                                                                        31200000
158     ******************************************************************31210000
159      THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS  31220000
160      ROUTINE.                                                          31230000
161                                                                        31240000
162      THE VSIO_PARAMETER_VALUES SUPPLY THE VALUES USED TO MOVE INTO     31250000
163      PARAMETER ENTRIES TO TAILOR THE ROUTINE TO A SPECIFIC DATASET AND  31260000
164      TO PROVIDE COMMANDS TO DRIVE THE ROUTINE.                         31270000
165     ******************************************************************/31280000
166                                                                        31290000
167            DECLARE                                                     31300000
168               1 VSIO_PARAMETER_VALUES   STATIC,                        31310000
169                  2 VSIO_OPEN            CHAR(8)   INIT('OPEN    '),    31320000
170                  2 VSIO_CLOSE           CHAR(8)   INIT('CLOSE   '),    31330000
171                  2 VSIO_READ            CHAR(8)   INIT('READ    '),    31340000
172                  2 VSIO_WRITE           CHAR(8)   INIT('WRITE   '),    31350000
173                  2 VSIO_REWRITE         CHAR(8)   INIT('REWRITE '),    31360000
174                  2 VSIO_DELETE          CHAR(8)   INIT('DELETE  '),    31370000
175                  2 VSIO_START_EQUAL     CHAR(8)   INIT('STARTEQ '),    31380000
```

MACRO SOURCE2 LISTING

```
                  2 VSIO_START_NOTLESS   CHAR(8)   INIT('STARTGE '),      31390000
                  2 VSIO_KSDS            CHAR(4)   INIT('KSDS'),          31400000
                  2 VSIO_ESDS            CHAR(4)   INIT('ESDS'),          31410000
                  2 VSIO_RRDS            CHAR(4)   INIT('RRDS'),          31420000
                  2 VSIO_SEQUENTIAL      CHAR(10)  INIT('SEQUENTIAL'),    31430000
                  2 VSIO_DIRECT          CHAR(10)  INIT('DIRECT    '),    31440000
                  2 VSIO_DYNAMIC         CHAR(10)  INIT('DYNAMIC   '),    31450000
                  2 VSIO_INPUT           CHAR(6)   INIT('INPUT '),        31460000
                  2 VSIO_OUTPUT          CHAR(6)   INIT('OUTPUT'),        31470000
                  2 VSIO_INPUT_OUTPUT    CHAR(6)   INIT('UPDATE'),        31480000
                  2 (VSIO_RC_SUCCESS                INIT(0),              31490000
                     VSIO_RC_LOGIC_ERROR            INIT(8),              31500000
                     VSIO_RC_END_OF_FILE            INIT(9999),           31510000
                     VSIO_RC_UNKNOWN_COMMAND        INIT(20),             31520000
                     VSIO_RC_DATASET_ALREADY_OPEN   INIT(21),             31530000
                     VSIO_RC_DATASET_NOT_OPEN       INIT(22),             31540000
                     VSIO_RC_ORGANIZATION_UNKNOWN   INIT(23),             31550000
                     VSIO_RC_ACCESS_UNKNOWN         INIT(24),             31560000
                     VSIO_RC_ORG_ACCESS_MISMATCH    INIT(25),             31570000
                     VSIO_RC_MODE_UNKNOWN           INIT(26),             31580000
                     VSIO_RC_MODE_UNSUPPORTED       INIT(27),             31590000
                     VSIO_RC_DDNAME_BLANK           INIT(28))             31600000
                                          FIXED BINARY(15,0),             31610000
                  2 (VSIO_FB_DUPLICATE_RECORD       INIT(8),              31620000
                     VSIO_FB_KEY_SEQUENCE           INIT(12),             31630000
                     VSIO_FB_RECORD_NOT_FOUND       INIT(16),             31640000
                     VSIO_FB_NO_MORE_SPACE          INIT(28),             31650000
                     VSIO_FB_READ_WITHOUT_START     INIT(88))             31660000
                                          FIXED BINARY(15,0),             31670000
    /*****************************************************************31680000
      THE VSIO_PARAMETER_BLOCK IS THE COMMUNICATION INTERFACE TO THE     31690000
      THE ROUTINE.                                                       31700000
     *****************************************************************/31710000
                                                                         31720000
               1 VSIO_PARAMETER_BLOCK    STATIC,                         31730000
                  2 VSIO_COMMAND         CHAR(8)   INIT(' '),            31740000
                  2 (VSIO_RETURN_CODE,                                   31750000
                     VSIO_VSAM_RC,                                       31760000
                     VSIO_VSAM_FUNCTION,                                 31770000
                     VSIO_VSAM_FEEDBACK) FIXED BINARY(15,0) INIT(0);     31780000
                                                                         31790000
    /*****************************************************************31800000
                            END OF VSAMIO COPY BOOK                      31810000
     *****************************************************************/31820000
```

MACRO SOURCE2 LISTING

INCLUDED TEXT FOLLOWS FROM DD.MEMBER = SYSLIB .VSAMIOFB

```
220    /*****************************************************************00000100
221                                                                      00000200
222       VV   VV   SSSSS      A     M     M  IIII   OOOOO   FFFFFFF  BBBBBB  00000300
223       VV   VV  SS   SS    AAA    MM   MM   II   OO   OO  FF       BB   BB 00000400
224       VV   VV  SS        AA AA   MMM MMM   II   OO   OO  FF       BB   BB 00000500
225       VV   VV   SSSSS   AA   AA  MMMMMMM   II   OO   OO  FFFFF    BBBBBB  00000600
226       VV   VV       SS  AA   AA  MM M MM   II   OO   OO  FF       BB   BB 00000700
227        VV VV   SS   SS  AAAAAAA  MM   MM   II   OO   OO  FF       BB   BB 00000800
228         VVV    SS   SS  AA   AA  MM   MM   II   OO   OO  FF       BB   BB 00000900
229          V      SSSSS   AA   AA  MM   MM  IIII   OOOOO   FF       BBBBBB  00001000
230                                                                      00001100
231     *****************************************************************00001200
232       THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS 00001300
233       ROUTINE, AND ARE USED TO COMMUNICATE CHARACTERISTICS FOR A SINGLE   00001400
234       VSAM DATASET.                                                      00001500
235                                                                          00001600
236       WITH THE 2 EXCEPTIONS FOR RECORD LENGTH (TO ACCOMODATE VARIABLE     00001700
237       LENGTH RECORDS) AND RELATIVE RECORD (TO ACCOMODATE RELATIVE RECORD  00001800
238       DATASETS), THESE DATA NAMES MUST BE POPULATED PRIOR TO CALLING THE  00001900
239       ROUTINE TO OPEN THE DATASET AND MUST NOT THEN BE CHANGED UNTIL THE  00002000
240       DATASET HAS BEEN CLOSED.                                           00002100
241     *****************************************************************/00002200
242                                                                          00002300
243           DECLARE                                                        00002400
244              1 VSIO_FILE_BLOCK            STATIC,                        00002500
245                 2 VSFB_DDNAME             CHAR(8)   INIT(' '),           00002600
246                 2 VSFB_ORGANIZATION       CHAR(4)   INIT(' '),           00002700
247                 2 VSFB_ACCESS             CHAR(10)  INIT(' '),           00002800
248                 2 VSFB_MODE               CHAR(6)   INIT(' '),           00002900
249                 2 (VSFB_RECORD_LENGTH,                                   00003000
250                    VSFB_KEY_POSITION,                                    00003100
251                    VSFB_KEY_LENGTH)       FIXED BINARY(15,0) INIT(0),    00003200
252                 2 VSFB_FILE_STATUS        CHAR(1)   INIT('C'),           00003300
253                 2 VSFB_RESERVED           CHAR(161);                     00003400
254                                                                          00003500
255     /*****************************************************************00003600
256                            END OF VSAMIOFB COPY BOOK                     00003700
257      *****************************************************************/00003800
```

NO ERROR OR WARNING CONDITION HAS BEEN DETECTED FOR THIS MACRO PASS.

SOURCE LISTING.

```
                /*****************************************************************    1
                                                                                       1
                  ESDSLOAD - TESTS THE VSAMIO ROUTINE BY LOADING AN ESDS CLUSTER WITH   1
                             RECORDS FROM A SEQUENTIAL DATASET.                        1
                                                                                       1
                 *****************************************************************/   1
                                                                                       6
   1             ESDSLOD:                                                             7
                    PROCEDURE OPTIONS(MAIN);                                           8
                                                                                       9
   2                   ON ERROR                                                       10
   3                      BEGIN;                                                      11
   4                         ON ERROR SYSTEM;                                         12
   5                         PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');   13
   6                         PUT SKIP DATA;                                           14
   7                         PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');   15
   8                      END;                                                        16
                                                                                      17
   9                   OPEN                                                         18
                          FILE(IMAGES),                                             19
                          FILE(PRINTR) LINESIZE(121);                               20
                                                                                    21
  10                   ON ENDFILE(IMAGES)                                           22
  11                      MORE_RECORDS = NO;                                        23
                                                                                    24
  12                   PRINT_AREA = 'ESDSLOAD: WRITE ESDS SEQUENTIALLY';            25
  13                   WRITE FILE(PRINTR) FROM(PRINT_LINE);                         26
  14                   PRINT_AREA = '---------------------------------';            27
  15                   WRITE FILE(PRINTR) FROM(PRINT_LINE);                         28
  16                   PRINT_AREA = ' ';                                            29
  17                   WRITE FILE(PRINTR) FROM(PRINT_LINE);                         30
                                                                                    31
  18                   MORE_RECORDS = YES;                                          32
                                                                                    33
                  /*****************************************************************   34
                    ESTABLISH PARAMETERS FOR VSAM DATASET AND CALL ROUTINE TO OPEN       34
                   *****************************************************************/   34
                                                                                       36
  19                   VSFB_DDNAME = 'ESDSF01';                                     37
  20                   VSFB_ORGANIZATION = VSIO_ESDS;                               38
  21                   VSFB_ACCESS = VSIO_SEQUENTIAL;                               39
  22                   VSFB_MODE = VSIO_OUTPUT;                                     40
  23                   VSFB_RECORD_LENGTH = 80;                                     41
```

```
24                    VSFB_KEY_POSITION = 0;                                    42
25                    VSFB_KEY_LENGTH = 0;                                      43
26                    VSIO_COMMAND = VSIO_OPEN;                                 44
27                    CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                       45
                                    VSIO_FILE_BLOCK,                            46
                                    RECORD_IMAGE);                              47
28                    IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN             48
29                       DO;                                                    49
30                          CALL VSIO_ERROR;                                    50
31                          RETURN;                                             51
32                       END;                                                   52
                                                                                53
33                    DO WHILE(MORE_RECORDS);                                   54
34                       READ FILE(IMAGES) INTO(RECORD_IMAGE);                  55
35                       IF (MORE_RECORDS) THEN                                 56
36                          CALL WRITE_ES;                                      57
37                    END;                                                      58
                                                                                59
38                    CLOSE FILE(IMAGES);                                       60
                                                                                61
                   /*************************************************************  62
                     CALL ROUTINE TO CLOSE VSAM DATASET                         62
                    *************************************************************/  62
                                                                                64
39                    VSIO_COMMAND = VSIO_CLOSE;                                65
40                    CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                       66
                                    VSIO_FILE_BLOCK,                            67
                                    RECORD_IMAGE);                              68
41                    IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN             69
42                       CALL VSIO_ERROR;                                       70
                                                                                71
43                    COUNTER_EDIT = RECORD_COUNTER;                            72
44                    PRINT_AREA = COUNTER_EDIT ||                              73
                                   ' RECORDS WERE LOADED SUCCESSFULLY';         74
45                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                      75
                                                                                76
46                    RETURN;                                                   77
                                                                                78
47                 WRITE_ES:                                                    79
                      PROCEDURE;                                                80
                                                                                81
                   /*************************************************************  82
                     CALL ROUTINE TO WRITE RECORD INTO VSAM DATASET             82
                    *************************************************************/  82
                                                                                84
48                    VSIO_COMMAND = VSIO_WRITE;                                85
49                    CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                       86
```

```
                                VSIO_FILE_BLOCK,                                  87
                                RECORD_IMAGE);                                    88
50                   IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN                89
51                      CALL VSIO_ERROR;                                          90
52                   ELSE                                                         91
52                      RECORD_COUNTER = RECORD_COUNTER + 1;                      92
                                                                                  93
53                   RETURN;                                                      94
                                                                                  95
54                END WRITE_ES;                                                   96
                                                                                  97
55              VSIO_ERROR:                                                       98
                   PROCEDURE;                                                     99
56                    PRINT_AREA = 'VSAMIO ERROR OCCURRED DURING ' ||            100
                                   VSIO_COMMAND;                                  101
57                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                       102
58                    PRINT_AREA = 'VSIO_RETURN_CODE = ' ||                      103
                                   VSIO_RETURN_CODE;                              104
59                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                       105
60                    PRINT_AREA = 'VSIO_VSAM_RETURN_CODE = ' ||                 106
                                   VSIO_VSAM_RETURN_CODE;                         107
61                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                       108
62                    PRINT_AREA = 'VSIO_VSAM_FUNCTION_CODE = ' ||               109
                                   VSIO_VSAM_FUNCTION_CODE;                       110
63                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                       111
64                    PRINT_AREA = 'VSIO_VSAM_FEEDBACK_CODE = ' ||               112
                                   VSIO_VSAM_FEEDBACK_CODE;                       113
65                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                       114
66                    PRINT_AREA = ' ';                                          115
                                                                                 116
67                    RETURN;                                                    117
                                                                                 118
68                 END VSIO_ERROR;                                               119
                                                                                 120
69                 DECLARE                                                       121
                      IMAGES FILE INPUT RECORD SEQUENTIAL EXTERNAL               122
                         ENV(F),                                                 123
                      PRINTR FILE OUTPUT RECORD SEQUENTIAL EXTERNAL              124
                         ENV(F CTLASA);                                          125
                                                                                 126
70                 DECLARE                                                       127
                      COUNTER_EDIT                  PICTURE 'ZZ,ZZZ,ZZ9V',     128
                      MORE_RECORDS                  BIT(1),                    129
                      NO                            BIT(1)    INIT('0'B),      130
                      RECORD_COUNTER                FIXED BINARY(15,0),        131
                      YES                           BIT(1)    INIT('1'B);      132
                                                                                 133
```

```
71                 DECLARE                                                      134
                      1 RECORD_IMAGE,                                           135
                         2 RECORD_FIELDS              CHAR(80);                 136
                                                                                137
72                 DECLARE                                                      138
                      1 PRINT_LINE,                                             139
                         2 CARRIAGE_CONTROL           CHAR(1)   INIT(' '),      140
                         2 PRINT_AREA                 CHAR(120);                141
                                                                                142
                  /*************************************************************** 147
                                                                                147
                        VV   VV    SSSSS       A      M     M    IIII      OOOOO  147
                        VV   VV   SS   SS     AAA     MM   MM     II      OO   OO 147
                        VV   VV   SS         AA AA    MMM MMM     II      OO   OO 147
                        VV   VV    SSSSS    AA   AA   MMMMMMM     II      OO   OO 147
                        VV   VV        SS   AA   AA   MM M MM     II      OO   OO 147
                         VV VV    SS   SS   AAAAAAA   MM   MM     II      OO   OO 147
                          VVV     SS   SS   AA   AA   MM   MM     II      OO   OO 147
                           V       SSSSS    AA   AA   MM   MM    IIII      OOOOO  147
                                                                                147
                   ***************************************************************147
                    THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS 147
                    ROUTINE.                                                    147
                                                                                147
                    THE VSIO_PARAMETER_VALUES SUPPLY THE VALUES USED TO MOVE INTO 147
                    PARAMETER ENTRIES TO TAILOR THE ROUTINE TO A SPECIFIC DATASET AND 147
                    TO PROVIDE COMMANDS TO DRIVE THE ROUTINE.                   147
                   ***************************************************************/147
                                                                                165
73                 DECLARE                                                      167
                      1 VSIO_PARAMETER_VALUES   STATIC,                         168
                         2 VSIO_OPEN            CHAR(8)   INIT('OPEN    '),     169
                         2 VSIO_CLOSE           CHAR(8)   INIT('CLOSE   '),     170
                         2 VSIO_READ            CHAR(8)   INIT('READ    '),     171
                         2 VSIO_WRITE           CHAR(8)   INIT('WRITE   '),     172
                         2 VSIO_REWRITE         CHAR(8)   INIT('REWRITE '),     173
                         2 VSIO_DELETE          CHAR(8)   INIT('DELETE  '),     174
                         2 VSIO_START_EQUAL     CHAR(8)   INIT('STARTEQ '),     175
                         2 VSIO_START_NOTLESS   CHAR(8)   INIT('STARTGE '),     176
                         2 VSIO_KSDS            CHAR(4)   INIT('KSDS'),         177
                         2 VSIO_ESDS            CHAR(4)   INIT('ESDS'),         178
                         2 VSIO_RRDS            CHAR(4)   INIT('RRDS'),         179
                         2 VSIO_SEQUENTIAL      CHAR(10)  INIT('SEQUENTIAL'),   180
                         2 VSIO_DIRECT          CHAR(10)  INIT('DIRECT    '),   181
                         2 VSIO_DYNAMIC         CHAR(10)  INIT('DYNAMIC   '),   182
                         2 VSIO_INPUT           CHAR(6)   INIT('INPUT '),       183
                         2 VSIO_OUTPUT          CHAR(6)   INIT('OUTPUT'),       184
```

```
                        2 VSIO_INPUT_OUTPUT    CHAR(6)   INIT('UPDATE'),          185
                        2 (VSIO_RC_SUCCESS                INIT(0),                 186
                           VSIO_RC_LOGIC_ERROR            INIT(8),                 187
                           VSIO_RC_END_OF_FILE            INIT(9999),              188
                           VSIO_RC_UNKNOWN_COMMAND        INIT(20),                189
                           VSIO_RC_DATASET_ALREADY_OPEN   INIT(21),                190
                           VSIO_RC_DATASET_NOT_OPEN       INIT(22),                191
                           VSIO_RC_ORGANIZATION_UNKNOWN   INIT(23),                192
                           VSIO_RC_ACCESS_UNKNOWN         INIT(24),                193
                           VSIO_RC_ORG_ACCESS_MISMATCH    INIT(25),                194
                           VSIO_RC_MODE_UNKNOWN           INIT(26),                195
                           VSIO_RC_MODE_UNSUPPORTED       INIT(27),                196
                           VSIO_RC_DDNAME_BLANK           INIT(28))                197
                                                  FIXED BINARY(15,0),              198
                        2 (VSIO_FB_DUPLICATE_RECORD       INIT(8),                 199
                           VSIO_FB_KEY_SEQUENCE           INIT(12),                200
                           VSIO_FB_RECORD_NOT_FOUND       INIT(16),                201
                           VSIO_FB_NO_MORE_SPACE          INIT(28),                202
                           VSIO_FB_READ_WITHOUT_START     INIT(88))                203
                                                  FIXED BINARY(15,0),              204
                     /*****************************************************************  205
                       THE VSIO_PARAMETER_BLOCK IS THE COMMUNICATION INTERFACE TO THE     205
                       THE ROUTINE.                                                       205
                      *****************************************************************/  205
                                                                                         208
                     1 VSIO_PARAMETER_BLOCK    STATIC,                                 210
                        2 VSIO_COMMAND         CHAR(8)   INIT(' '),                    211
                        2 (VSIO_RETURN_CODE,                                           212
                           VSIO_VSAM_RC,                                               213
                           VSIO_VSAM_FUNCTION,                                         214
                           VSIO_VSAM_FEEDBACK) FIXED BINARY(15,0) INIT(0);             215
                                                                                       216
                     /*****************************************************************  217
                                          END OF VSAMIO COPY BOOK                      217
                      *****************************************************************/  217
                     /*****************************************************************  220
                                                                                         220
                        VV    VV    SSSSS       A      M      M  IIII    OOOOO    FFFFFFF  BBBBBB    220
                        VV    VV   SS    SS    AAA     MM    MM   II    OO   OO  FF       BB    BB   220
                        VV    VV   SS         AA AA    MMM MMM    II    OO   OO  FF       BB    BB   220
                        VV    VV    SSSSS    AA   AA   MMMMMMM    II    OO   OO  FFFFF    BBBBBB    220
                        VV    VV        SS   AA   AA   MM M MM    II    OO   OO  FF       BB    BB   220
                         VV VV    SS    SS   AAAAAAA  MM   MM    II    OO   OO  FF       BB    BB   220
                          VVV     SS    SS   AA   AA  MM   MM    II    OO   OO  FF       BB    BB   220
                           V       SSSSS    AA   AA  MM   MM   IIII    OOOOO   FF       BBBBBB    220
                                                                                         220
                      *****************************************************************  220
```

```
                  THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS   220
                  ROUTINE, AND ARE USED TO COMMUNICATE CHARACTERISTICS FOR A SINGLE     220
                  VSAM DATASET.                                                         220
                                                                                        220
                  WITH THE 2 EXCEPTIONS FOR RECORD LENGTH (TO ACCOMODATE VARIABLE       220
                  LENGTH RECORDS) AND RELATIVE RECORD (TO ACCOMODATE RELATIVE RECORD    220
                  DATASETS), THESE DATA NAMES MUST BE POPULATED PRIOR TO CALLING THE    220
                  ROUTINE TO OPEN THE DATASET AND MUST NOT THEN BE CHANGED UNTIL THE    220
                  DATASET HAS BEEN CLOSED.                                              220
                ******************************************************************/    220
                                                                                        241
74                 DECLARE                                                              243
                      1 VSIO_FILE_BLOCK           STATIC,                               244
                         2 VSFB_DDNAME            CHAR(8)   INIT(' '),                  245
                         2 VSFB_ORGANIZATION      CHAR(4)   INIT(' '),                  246
                         2 VSFB_ACCESS            CHAR(10)  INIT(' '),                  247
                         2 VSFB_MODE              CHAR(6)   INIT(' '),                  248
                         2 (VSFB_RECORD_LENGTH,                                         249
                            VSFB_KEY_POSITION,                                          250
                            VSFB_KEY_LENGTH)      FIXED BINARY(15,0) INIT(0),           251
                         2 VSFB_FILE_STATUS       CHAR(1)   INIT('C'),                  252
                         2 VSFB_RESERVED          CHAR(161);                            253
                                                                                        254
                ******************************************************************     255
                                  END OF VSAMIOFB COPY BOOK                             255
                ******************************************************************/    255
                                                                                        145
75                 END ESDSLOD;                                                         146
```

ATTRIBUTE AND CROSS-REFERENCE TABLE

DCL NO.		IDENTIFIER	ATTRIBUTES AND REFERENCES
72		CARRIAGE_CONTROL	IN PRINT_LINE,AUTOMATIC,UNALIGNED,INITIAL,STRING(1),CHARACTER
70		COUNTER_EDIT	AUTOMATIC,UNALIGNED,DECIMAL,PICTURE(ZZ,ZZZ,ZZ9V) 43,44
1		ESDSLOD	ENTRY,DECIMAL,FLOAT(SINGLE)
69		IMAGES	FILE,EXTERNAL,INPUT,RECORD,SEQUENTIAL,ENVIRONMENT(F) 9,10,34,38
70		MORE_RECORDS	AUTOMATIC,UNALIGNED,STRING(1),BIT 11,18,33,35
70		NO	AUTOMATIC,UNALIGNED,INITIAL,STRING(1),BIT 11
72		PRINT_AREA	IN PRINT_LINE,AUTOMATIC,UNALIGNED,STRING(120),CHARACTER 12,14,16,44,56,58,60,62,64,66
72		PRINT_LINE	AUTOMATIC,STRUCTURE 13,15,17,45,57,59,61,63,65
69		PRINTR	FILE,EXTERNAL,OUTPUT,RECORD,SEQUENTIAL,ENVIRONMENT(F CTLASA) 9,13,15,17,45,57,59,61,63,65
70	*********	RECORD_COUNTER	AUTOMATIC,ALIGNED,BINARY,FIXED(15,0) 43,52,52
71		RECORD_FIELDS	IN RECORD_IMAGE,AUTOMATIC,UNALIGNED,STRING(80),CHARACTER
71		RECORD_IMAGE	AUTOMATIC,STRUCTURE 27,34,40,49
		SYSPRINT	FILE,EXTERNAL 5,6,7
		VSAMIOP	EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE) 27,40,49
74		VSFB_ACCESS	IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(10),CHARACTER 21

DCL NO.	IDENTIFIER	ATTRIBUTES AND REFERENCES
74	VSFB_DDNAME	IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(8),CHARACTER 19
74	VSFB_FILE_STATUS	IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(1),CHARACTER
74 *********	VSFB_KEY_LENGTH	IN VSIO_FILE_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED(15,0) 25
74 *********	VSFB_KEY_POSITION	IN VSIO_FILE_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED(15,0) 24
74	VSFB_MODE	IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(6),CHARACTER 22
74	VSFB_ORGANIZATION	IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(4),CHARACTER 20
74 *********	VSFB_RECORD_LENGTH	IN VSIO_FILE_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED(15,0) 23
74	VSFB_RESERVED	IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,STRING(161),CHARACTER
73	VSIO_CLOSE	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER 39
73	VSIO_COMMAND	IN VSIO_PARAMETER_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER 26,39,48,56
73	VSIO_DELETE	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
73	VSIO_DIRECT	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(10), CHARACTER
73	VSIO_DYNAMIC	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(10), CHARACTER
55	VSIO_ERROR	ENTRY,DECIMAL,FLOAT(SINGLE) 30,42,51
73	VSIO_ESDS	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(4), CHARACTER

DCL NO.		IDENTIFIER	ATTRIBUTES AND REFERENCES
			20
73	*********	VSIO_FB_DUPLICATE_RECORD	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_FB_KEY_SEQUENCE	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_FB_NO_MORE_SPACE	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_FB_READ_WITHOUT_START	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_FB_RECORD_NOT_FOUND	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
74		VSIO_FILE_BLOCK	STATIC,STRUCTURE 27,40,49
73		VSIO_INPUT	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(6), CHARACTER
73		VSIO_INPUT_OUTPUT	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(6), CHARACTER
73		VSIO_KSDS	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(4), CHARACTER
73		VSIO_OPEN	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
			26
73		VSIO_OUTPUT	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(6), CHARACTER
			22
73		VSIO_PARAMETER_BLOCK	STATIC,STRUCTURE 27,40,49
73		VSIO_PARAMETER_VALUES	STATIC,STRUCTURE
73	*********	VSIO_RC_ACCESS_UNKNOWN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)

DCL NO.		IDENTIFIER	ATTRIBUTES AND REFERENCES
73	*********	VSIO_RC_DATASET_ALREADY_OPEN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_RC_DATASET_NOT_OPEN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_RC_DDNAME_BLANK	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_RC_END_OF_FILE	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_RC_LOGIC_ERROR	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_RC_MODE_UNKNOWN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_RC_MODE_UNSUPPORTED	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_RC_ORG_ACCESS_MISMATCH	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_RC_ORGANIZATION_UNKNOWN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73	*********	VSIO_RC_SUCCESS	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0) 28,41,50
73	*********	VSIO_RC_UNKNOWN_COMMAND	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
73		VSIO_READ	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
73	*********	VSIO_RETURN_CODE	IN VSIO_PARAMETER_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0) 28,41,50,58
73		VSIO_REWRITE	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
73		VSIO_RRDS	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(4),

DCL NO.		IDENTIFIER	ATTRIBUTES AND REFERENCES
			CHARACTER
73		VSIO_SEQUENTIAL	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(10), CHARACTER 21
73		VSIO_START_EQUAL	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
73		VSIO_START_NOTLESS	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
73	*********	VSIO_VSAM_FEEDBACK	IN VSIO_PARAMETER_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
		VSIO_VSAM_FEEDBACK_CODE	AUTOMATIC,ALIGNED,DECIMAL,FLOAT(SINGLE) 64
73	*********	VSIO_VSAM_FUNCTION	IN VSIO_PARAMETER_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
		VSIO_VSAM_FUNCTION_CODE	AUTOMATIC,ALIGNED,DECIMAL,FLOAT(SINGLE) 62
73	*********	VSIO_VSAM_RC	IN VSIO_PARAMETER_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
		VSIO_VSAM_RETURN_CODE	AUTOMATIC,ALIGNED,DECIMAL,FLOAT(SINGLE) 60
73		VSIO_WRITE	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER 48
47		WRITE_ES	ENTRY,DECIMAL,FLOAT(SINGLE) 36
70		YES	AUTOMATIC,UNALIGNED,INITIAL,STRING(1),BIT 18

AGGREGATE LENGTH TABLE

STATEMENT NO.	IDENTIFIER	LENGTH IN BYTES
72	PRINT_LINE	121
71	RECORD_IMAGE	80
74	VSIO_FILE_BLOCK	196
73	VSIO_PARAMETER_BLOCK	16
73	VSIO_PARAMETER_VALUES	158

STORAGE REQUIREMENTS.

THE STORAGE AREA FOR THE PROCEDURE LABELLED ESDSLOD IS 520 BYTES LONG.

THE STORAGE AREA FOR THE ON UNIT AT STATEMENT NO. 3 IS 184 BYTES LONG.

THE STORAGE AREA FOR THE ON UNIT AT STATEMENT NO. 10 IS 176 BYTES LONG.

THE STORAGE AREA (IN STATIC) FOR THE PROCEDURE LABELLED WRITE_ES IS 176 BYTES LONG.

THE STORAGE AREA (IN STATIC) FOR THE PROCEDURE LABELLED VSIO_ERROR IS 256 BYTES LONG.

THE PROGRAM CSECT IS NAMED ESDSLOD AND IS 1682 BYTES LONG.

THE STATIC CSECT IS NAMED ESDSLODA AND IS 5488 BYTES LONG.

STATISTICS MACRO RECORDS = 257,SOURCE RECORDS = 259,PROG TEXT STMNTS = 75,OBJECT BYTES = 1682

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN ON UNIT

OFFSET (HEX) 0000 0050 005C 007A 0094 00B2
STATEMENT NO 3 4 5 6 7 8

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN ON UNIT

OFFSET (HEX) 0000 0048 0052
STATEMENT NO 11 11

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN PROCEDURE WRITE_ES

OFFSET (HEX) 0000 0038 003E 005E 006A 0078 0088 008E
STATEMENT NO 47 48 49 50 51 52 53 54

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN PROCEDURE VSIO_ERROR

OFFSET (HEX) 0000 0038 0052 006A 00B6 00CE 0104 011C 014E 0166 0198 01B0 01BC 01C2
STATEMENT NO 55 56 57 58 59 60 61 62 63 64 65 66 67 68

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN PROCEDURE ESDSLOD

OFFSET (HEX) 0000 00D4 00E2 00EC 00FA 0106 011E 012A 0142 014E 0166 016C 0172 0178 017E 0184 018A 0190 0196 019C 01B8
STATEMENT NO 1 2 9 10 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

OFFSET (HEX) 01C4 01C4 01CE 01D4 01D4 01DC 01F4 01FC 0206 020A 0214 021A 0236 0242 024C 0266 0282 029A 02A0
STATEMENT NO 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 75

COMPILER DIAGNOSTICS.

WARNINGS.

IEM0227I NO FILE/STRING OPTION SPECIFIED IN ONE OR MORE GET/PUT STATEMENTS. SYSIN/SYSPRINT HAS BEEN
ASSUMED IN EACH CASE.

IEM0764I ONE OR MORE FIXED BINARY ITEMS OF PRECISION 15 OR LESS HAVE BEEN GIVEN HALFWORD STORAGE. THEY
ARE FLAGGED '*********' IN THE XREF/ATR LIST.

IEM1790I DATA CONVERSIONS WILL BE DONE BY SUBROUTINE CALL IN THE FOLLOWING STATEMENTS 60, 62, 64.

END OF DIAGNOSTICS.

AUXILIARY STORAGE WILL NOT BE USED FOR DICTIONARY WHEN SIZE = 138K

COMPILE TIME .00 MINS
ELAPSED TIME .00 MINS

```
 F64-LEVEL LINKAGE EDITOR OPTIONS SPECIFIED   NONE
          DEFAULT OPTION(S) USED -  SIZE=(231424,55296)
****GO        DOES NOT EXIST BUT HAS BEEN ADDED TO DATA SET
AUTHORIZATION CODE IS         0.
```

```
ESDSLOAD: WRITE ESDS SEQUENTIALLY
---------------------------------
       100 RECORDS WERE LOADED SUCCESSFULLY
```