

J E S 2 J O B L O G

```
18.27.28 JOB 163 IEF677I WARNING MESSAGE(S) FOR JOB VSTESTR7 ISSUED
18.27.28 JOB 163 $HASP373 VSTESTR7 STARTED - INIT 1 - CLASS A - SYS HMVS
18.27.28 JOB 163 IEF403I VSTESTR7 - STARTED - TIME=18.27.28
18.27.28 JOB 163 CCI001C PL1L /IEMAA /00:00:00.18/ /00004/SYS /VSTESTR7
18.27.28 JOB 163 CCI001C LKED /IEWL /00:00:00.04/ /00000/SYS /VSTESTR7
18.27.28 JOB 163 CCI001C GO /PGM=*.DD/00:00:00.01/ /00000/SYS /VSTESTR7
18.27.28 JOB 163 IEF404I VSTESTR7 - ENDED - TIME=18.27.28
18.27.28 JOB 163 $HASP395 VSTESTR7 ENDED
```

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

07 JUL 20 JOB EXECUTION DATE

22 CARDS READ

1,179 SYSOUT PRINT RECORDS

0 SYSOUT PUNCH RECORDS

0.00 MINUTES EXECUTION TIME

```

1 //VSTESTR7 JOB (SYS), 'VSAMIOP IVP RRDSSSEQ', CLASS=A, MSGCLASS=X, JOB 163
// REGION=4096K
***
*****
*** PL/1 MODULE: RRDSSSEQ VSAM DATASET: VSTESTRR.CLUSTER (RRDS)
***
*** TESTS START AND READ FUNCTIONS AGAINST RRDS 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.SYSIN DD DSN=SYSC.VSAMIOP.SOURCE(RRDSSSEQ), DISP=SHR
10 //PL1L.SYSLIB DD DSN=SYSC.VSAMIOP.MACLIB, 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.SYSUDUMP DD SYSOUT=*
25 //GO.SYSPRINT DD SYSOUT=*
26 //GO.RRDSF01 DD DSN=PUB001.VSTESTRR.CLUSTER, DISP=OLD

```

STMT NO. MESSAGE

19 IEF686I DDNAME REFERRED TO ON DDNAME KEYWORD IN PRIOR STEP WAS NOT RESOLVED

IEF236I ALLOC. FOR VSTESTR7 PL1L PL1F
IEF237I 253 ALLOCATED TO STEPLIB
IEF237I 253 ALLOCATED TO SYS00414
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 380 ALLOCATED TO SYSLIN
IEF237I 251 ALLOCATED TO SYSUT3
IEF237I 370 ALLOCATED TO SYSUT1
IEF237I 253 ALLOCATED TO SYSIN
IEF237I 253 ALLOCATED TO SYSLIB

IEF142I VSTESTR7 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.JOB00163.SO0101 SYSOUT
IEF285I SYS20189.T182728.RA000.VSTESTR7.LOADSET PASSED \*-----221
IEF285I VOL SER NOS= MVS380.
IEF285I SYS20189.T182728.RA000.VSTESTR7.SYSUT3 DELETED \*-----311
IEF285I VOL SER NOS= WORK00.
IEF285I SYS20189.T182728.RA000.VSTESTR7.SYSUT1 DELETED \*-----0
IEF285I VOL SER NOS= MVS370.
IEF285I SYSC.VSAMIOP.SOURCE KEPT \*-----3
IEF285I VOL SER NOS= SYSCPK.
IEF285I SYSC.VSAMIOP.MACLIB KEPT \*-----27
IEF285I VOL SER NOS= SYSCPK.

IEF373I STEP /PL1L / START 20189.1827

IEF374I STEP /PL1L / STOP 20189.1827 CPU 0MIN 00.18SEC SRB 0MIN 00.04SEC VIRT 4096K SYS 212K

\*\*\*\* JOBCARD READ 20189 18:27:28 \*\*\*\*

Table with job statistics including PRC-CCI, VS2, R03.8, HMVS, STEP STATISTICS, CPU usage, and memory usage.

IEF236I ALLOC. FOR VSTESTR7 LKED PL1F
IEF237I 253 ALLOCATED TO SYSLIB
IEF237I 253 ALLOCATED TO
IEF237I 253 ALLOCATED TO SYS00416
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 VSTESTR7 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.T182728.RA000.VSTESTR7.GOSET PASSED \*-----57
IEF285I VOL SER NOS= WORK00.
IEF285I SYS20189.T182728.RA000.VSTESTR7.SYSUT1 DELETED \*-----0
IEF285I VOL SER NOS= MVS370.
IEF285I JES2.JOB00163.SO0102 SYSOUT

```

IEF285I  SYS20189.T182728.RA000.VSTESTR7.LOADSET      DELETED      *-----222
IEF285I  VOL SER NOS= MVS380.
IEF373I  STEP /LKED      / START 20189.1827
IEF374I  STEP /LKED      / STOP  20189.1827 CPU      OMIN 00.04SEC SRB      OMIN 00.01SEC VIRT    260K SYS    208K
*****
*                               PRC-CCI 370/148 VS2 R03.8  HMVS  STEP STATISTICS                               *
*  STEP NAME  LKED      USER CORE      260K  TAPES USED/IO 000/000000000  START  TIME 18:27:28  TCB TIME 00:00:00.04 *
*  PGM NAME   IEWL      SYSTEM CORE    208K  DISKS USED/IO 004/000000385  STOP   TIME 18:27:28  SRB TIME 00:00:00.01 *
*  COND CODE  0000      PRIVATE AREA SZ 4096K  ALLOC TIME 18:27:28  ELAPSED TIME          PGM LOAD 18:27:28 *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
*  004      1968      00:00:00.06          0          0          0          0          0          0          0          0          0 *
*****
* CPU $ ( 0.01) + EXCP $ ( 0.51) + MEMORY $ ( 0.02) = TOTAL $ ( 0.54)                               *
*****
IEF236I  ALLOC. FOR VSTESTR7 GO PL1F
IEF237I  251  ALLOCATED TO PGM=*.DD
IEF237I  253  ALLOCATED TO STEPLIB
IEF237I  253  ALLOCATED TO
IEF237I  253  ALLOCATED TO SYS00418
IEF237I  JES2 ALLOCATED TO SYSPRINT
IEF237I  JES2 ALLOCATED TO PRINTR
IEF237I  JES2 ALLOCATED TO SYSUDUMP
IEF237I  JES2 ALLOCATED TO SYSPRINT
IEF237I  190  ALLOCATED TO RRDSF01
IEF237I  190  ALLOCATED TO SYS00420
IEF142I  VSTESTR7 GO PL1F - STEP WAS EXECUTED - COND CODE 0000
IEF285I  SYS20189.T182728.RA000.VSTESTR7.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.JOB00163.SO0103      SYSOUT
IEF285I  JES2.JOB00163.SO0104      SYSOUT
IEF285I  JES2.JOB00163.SO0105      SYSOUT
IEF285I  JES2.JOB00163.SO0106      SYSOUT
IEF285I  PUB001.VSTESTRR.CLUSTER    KEPT      *-----1
IEF285I  VOL SER NOS= PUB001.
IEF285I  UCPUB001      KEPT      *-----0
IEF285I  VOL SER NOS= PUB001.
IEF373I  STEP /GO      / START 20189.1827
IEF374I  STEP /GO      / STOP  20189.1827 CPU      OMIN 00.01SEC SRB      OMIN 00.00SEC VIRT    96K SYS    220K
*****
*                               PRC-CCI 370/148 VS2 R03.8  HMVS  STEP STATISTICS                               *
*  STEP NAME  GO      USER CORE      96K  TAPES USED/IO 000/000000000  START  TIME 18:27:28  TCB TIME 00:00:00.01 *
*  PGM NAME   PGM=*.DD  SYSTEM CORE    220K  DISKS USED/IO 003/000000001  STOP   TIME 18:27:28  SRB TIME 00:00:00.00 *
*  COND CODE  0000      PRIVATE AREA SZ 4096K  ALLOC TIME 18:27:28  ELAPSED TIME          PGM LOAD 18:27:28 *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
*  004      29      00:00:00.01          0          0          0          0          0          0          0          0 *
*****
* CPU $ ( 0.00) + EXCP $ ( 0.00) + MEMORY $ ( 0.00) = TOTAL $ ( 0.00)                               *
*****
IEF237I  251  ALLOCATED TO SYS00001
IEF285I  SYS20189.T182728.RA000.VSTESTR7.R0000001    KEPT      *-----0
IEF285I  VOL SER NOS= WORK00.
IEF285I  SYS20189.T182728.RA000.VSTESTR7.GOSET      DELETED
IEF285I  VOL SER NOS= WORK00.
IEF375I  JOB /VSTESTR7/ START 20189.1827
IEF376I  JOB /VSTESTR7/ STOP  20189.1827 CPU      OMIN 00.23SEC SRB      OMIN 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  /*****2530000
2  25310000
3  RRDSSEQ - TESTS THE VSAMIO ROUTINE BY USING START AND READ COMMANDS 25320000
4  ON AN RRDS CLUTER TO READ THE FILE IN SKIP-SEQUENTIAL 25330000
5  MODE. 25340000
6  25350000
7  *****/25360000
8  RRDSSEQ: 25370000
9  PROCEDURE OPTIONS(MAIN); 25380000
10 25390000
11  ON ERROR 25400000
12  BEGIN; 25410000
13  ON ERROR SYSTEM; 25420000
14  PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*'); 25430000
15  PUT SKIP DATA; 25440000
16  PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*'); 25450000
17  END; 25460000
18 25470000
19  OPEN 25480000
20  FILE(PRINTR) LINESIZE(133); 25490000
21 25500000
22  PRINT_AREA = 'RRDSSEQ: READ RRDS SEQUENTIALLY (W/START)'; 25510000
23  WRITE FILE(PRINTR) FROM(PRINT_LINE); 25520000
24  PRINT_AREA = '-----'; 25530000
25  WRITE FILE(PRINTR) FROM(PRINT_LINE); 25540000
26  PRINT_AREA = ' '; 25550000
27  WRITE FILE(PRINTR) FROM(PRINT_LINE); 25560000
28 25570000
29  /*****25580000
30  ESTABLISH PARAMETERS OF VSAM DATASET AND CALL ROUTINE TO OPEN IT 25590000
31  *****/25600000
32  VSFB_DDNAME = 'RRDSF01'; 25610000
33  VSFB_ORGANIZATION = VSIO_RRDS; 25620000
34  VSFB_ACCESS = VSIO_SEQUENTIAL; 25630000
35  VSFB_MODE = VSIO_INPUT; 25640000
36  VSFB_RECORD_LENGTH = 80; 25650000
37  VSFB_KEY_POSITION = 0; 25660000
38  VSFB_KEY_LENGTH = 0; 25670000
39  VSIO_COMMAND = VSIO_OPEN; 25680000
40  CALL VSAMIOP (VSIO_PARAMETER_BLOCK, 25690000
41  VSIO_FILE_BLOCK, 25700000
42  RECORD_IMAGE); 25710000
43  IF (VSIO_RETURN_CODEa = VSIO_RC_SUCCESS) THEN 25720000
44  DO; 25730000
```

## MACRO SOURCE2 LISTING

```
45          CALL VSIO_ERROR;                25740000
46          RETURN;                          25750000
47          END;                              25760000
48                                              25770000
49          VSFB_KEY_LENGTH = 21;            25780000
50          VSIO_COMMAND = VSIO_START_EQUAL; 25790000
51          PRINT_AREA = 'STARTING KEY EQUAL TO 21'; 25800000
52          WRITE FILE(PRINTR) FROM(PRINT_LINE); 25810000
53          CALL PROCESS_BLOCK;              25820000
54                                              25830000
55          VSFB_KEY_LENGTH = 51;            25840000
56          VSIO_COMMAND = VSIO_START_EQUAL; 25850000
57          PRINT_AREA = 'STARTING KEY EQUAL TO 51'; 25860000
58          WRITE FILE(PRINTR) FROM(PRINT_LINE); 25870000
59          CALL PROCESS_BLOCK;              25880000
60                                              25890000
61          VSFB_KEY_LENGTH = 81;            25900000
62          VSIO_COMMAND = VSIO_START_NOTLESS; 25910000
63          PRINT_AREA = 'STARTING KEY NOT LESS THAN 81'; 25920000
64          WRITE FILE(PRINTR) FROM(PRINT_LINE); 25930000
65          CALL PROCESS_BLOCK;              25940000
66                                              25950000
67          VSFB_KEY_LENGTH = 111;           25960000
68          VSIO_COMMAND = VSIO_START_NOTLESS; 25970000
69          PRINT_AREA = 'STARTING KEY NOT LESS THAN 111'; 25980000
70          WRITE FILE(PRINTR) FROM(PRINT_LINE); 25990000
71          CALL PROCESS_BLOCK;              26000000
72                                              26010000
73 /*****26020000
74          CALL ROUTINE TO CLOSE VSAM DATASET 26030000
75 *****26040000
76          VSIO_COMMAND = VSIO_CLOSE;       26050000
77          CALL VSAMIOP (VSIO_PARAMETER_BLOCK, 26060000
78                      VSIO_FILE_BLOCK,    26070000
79                      RECORD_IMAGE);      26080000
80          IF (VSIO_RETURN_CODE = VSIO_RC_SUCCESS) THEN 26090000
81              CALL VSIO_ERROR;            26100000
82                                              26110000
83          RETURN;                          26120000
84                                              26130000
85          PROCESS_BLOCK:                   26140000
86          PROCEDURE;                       26150000
87                                              26160000
88          CALL VSAMIOP (VSIO_PARAMETER_BLOCK, 26170000
89                      VSIO_FILE_BLOCK,    26180000
```

## MACRO SOURCE2 LISTING

```
90          RECORD_IMAGE);                26190000
91      IF (VSIO_RETURN_CODEa= VSIO_RC_SUCCESS) THEN 26200000
92          IF (VSIO_VSAM_FEEDBACK = VSIO_FB_RECORD_NOT_FOUND) THEN 26210000
93              DO;                          26220000
94                  PRINT_AREA = '*** NOT FOUND ***'; 26230000
95                  WRITE FILE(PRINTR) FROM(PRINT_LINE); 26240000
96                  RETURN;                  26250000
97              END;                          26260000
98          ELSE                              26270000
99              DO;                          26280000
100                  CALL VSIO_ERROR;         26290000
101                  RETURN;                 26300000
102              END;                          26310000
103                                          26320000
104      VSIO_COMMAND = VSIO_READ;            26330000
105      MORE_RECORDS = YES;                  26340000
106      RECORD_COUNTER = 0;                  26350000
107                                          26360000
108      DO WHILE(MORE_RECORDS & RECORD_COUNTER < 5); 26370000
109          CALL READ_RR;                    26380000
110          IF (MORE_RECORDS) THEN          26390000
111              DO;                          26400000
112                  RRN_EDIT = VSFB_KEY_LENGTH; 26410000
113                  PRINT_AREA = 'RRN: ' || RRN_EDIT || ' DATA: ' || 26420000
114                      RECORD_IMAGE_SCALAR; 26430000
115                  WRITE FILE(PRINTR) FROM(PRINT_LINE); 26440000
116              END;                          26450000
117          END;                              26460000
118                                          26470000
119          RETURN;                          26480000
120                                          26490000
121      END PROCESS_BLOCK;                    26500000
122                                          26510000
123      READ_RR:                             26520000
124          PROCEDURE;                       26530000
125                                          26540000
126      /*****26550000
127          CALL ROUTINE TO READ NEXT RECORD FROM VSAM DATASET 26560000
128          *****/26570000
129          VSIO_COMMAND = VSIO_READ;        26580000
130          CALL VSAMIOP (VSIO_PARAMETER_BLOCK, 26590000
131                      VSIO_FILE_BLOCK,    26600000
132                      RECORD_IMAGE);      26610000
133          IF (VSIO_RETURN_CODEa= VSIO_RC_SUCCESS) THEN 26620000
134              IF (VSIO_RETURN_CODE = VSIO_RC_END_OF_FILE) THEN 26630000
```



## MACRO SOURCE2 LISTING

```
135         MORE_RECORDS = NO;                26640000
136         ELSE                                26650000
137             CALL VSIO_ERROR;                26660000
138         ELSE                                26670000
139             RECORD_COUNTER = RECORD_COUNTER + 1; 26680000
140                                             26690000
141         RETURN;                            26700000
142                                             26710000
143     END READ_RR;                            26720000
144                                             26730000
145 VSIO_ERROR:                                26740000
146     PROCEDURE;                              26750000
147         PRINT_AREA = 'VSAMIO ERROR OCCURRED DURING ' ||
148             VSIO_COMMAND;                    26770000
149         WRITE FILE(PRINTR) FROM(PRINT_LINE); 26780000
150         PRINT_AREA = 'VSIO_RETURN_CODE = ' ||
151             VSIO_RETURN_CODE;                26800000
152         WRITE FILE(PRINTR) FROM(PRINT_LINE); 26810000
153         PRINT_AREA = 'VSIO_VSAM_RETURN_CODE = ' ||
154             VSIO_VSAM_RETURN_CODE;          26830000
155         WRITE FILE(PRINTR) FROM(PRINT_LINE); 26840000
156         PRINT_AREA = 'VSIO_VSAM_FUNCTION_CODE = ' ||
157             VSIO_VSAM_FUNCTION_CODE;        26860000
158         WRITE FILE(PRINTR) FROM(PRINT_LINE); 26870000
159         PRINT_AREA = 'VSIO_VSAM_FEEDBACK_CODE = ' ||
160             VSIO_VSAM_FEEDBACK_CODE;        26890000
161         WRITE FILE(PRINTR) FROM(PRINT_LINE); 26900000
162         PRINT_AREA = ' ';                    26910000
163                                             26920000
164         RETURN;                            26930000
165                                             26940000
166     END VSIO_ERROR;                        26950000
167                                             26960000
168     DECLARE                                26970000
169         PRINTR FILE OUTPUT RECORD SEQUENTIAL EXTERNAL 26980000
170         ENV(F CTLASA);                      26990000
171                                             27000000
172     DECLARE                                27010000
173         MORE_RECORDS                        BIT(1), 27020000
174         NO                                  BIT(1) INIT('0'B), 27030000
175         RECORD_COUNTER                      FIXED BINARY(15,0) INIT(0), 27040000
176         RRN_EDIT                            PICTURE 'ZZ,ZZ9V', 27050000
177         YES                                  BIT(1) INIT('1'B); 27060000
178                                             27070000
179     DECLARE                                27080000
```

## MACRO SOURCE2 LISTING

```

180      1 RECORD_IMAGE,                27090000
181          2 RECORD_KEY              CHAR(10),    27100000
182          2 RECORD_FIELDS          CHAR(70);    27110000
183                                          27120000
184      DECLARE                        27130000
185          RECORD_IMAGE_SCALAR      DEFINED RECORD_IMAGE 27140000
186                                          CHAR(80);    27150000
187                                          27160000
188      DECLARE                        27170000
189          1 PRINT_LINE,             27180000
190              2 CARRIAGE_CONTROL    CHAR(1)  INIT(' '), 27190000
191              2 PRINT_AREA          CHAR(132);  27200000
192                                          27210000
193      %INCLUDE (VSAMIO);             27220000
194      %INCLUDE (VSAMIOFB);           27230000
195                                          27240000
196      END RRDSSSQ;                   27250000

```

INCLUDED TEXT FOLLOWS FROM DD.MEMBER = SYSLIB .VSAMIO

```

197      /*****31100000
198                                          31110000
199          VV  VV  SSSSS      A      M      M      IIII      OOOOO      31120000
200          VV  VV  SS  SS      AAA      MM  MM      II      OO  OO      31130000
201          VV  VV  SS      AA AA      MMM MMM      II      OO  OO      31140000
202          VV  VV  SSSSS      AA  AA      MMMMMMMM      II      OO  OO      31150000
203          VV  VV      SS      AA  AA      MM M MM      II      OO  OO      31160000
204          VV VV  SS  SS      AAAAAA      MM  MM      II      OO  OO      31170000
205          VVV      SS  SS      AA  AA      MM  MM      II      OO  OO      31180000
206          V      SSSSS      AA  AA      MM  MM      IIII      OOOOO      31190000
207                                          31200000
208      *****/31210000
209      THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS 31220000
210      ROUTINE.                                                                31230000
211                                          31240000
212      THE VSIO_PARAMETER_VALUES SUPPLY THE VALUES USED TO MOVE INTO      31250000
213      PARAMETER ENTRIES TO TAILOR THE ROUTINE TO A SPECIFIC DATASET AND    31260000
214      TO PROVIDE COMMANDS TO DRIVE THE ROUTINE.                             31270000
215      *****/31280000
216                                          31290000
217      DECLARE                                                                31300000
218          1 VSIO_PARAMETER_VALUES  STATIC,                                  31310000
219              2 VSIO_OPEN          CHAR(8)  INIT('OPEN  '),              31320000
220              2 VSIO_CLOSE        CHAR(8)  INIT('CLOSE  '),              31330000

```

## MACRO SOURCE2 LISTING

```
221      2 VSIO_READ          CHAR(8)  INIT('READ  '),      31340000
222      2 VSIO_WRITE         CHAR(8)  INIT('WRITE  '),      31350000
223      2 VSIO_REWRITE        CHAR(8)  INIT('REWRITE '),      31360000
224      2 VSIO_DELETE         CHAR(8)  INIT('DELETE '),      31370000
225      2 VSIO_START_EQUAL    CHAR(8)  INIT('STARTEQ '),     31380000
226      2 VSIO_START_NOTLESS  CHAR(8)  INIT('STARTGE '),     31390000
227      2 VSIO_KSDS           CHAR(4)  INIT('KSDS'),        31400000
228      2 VSIO_ESDS           CHAR(4)  INIT('ESDS'),        31410000
229      2 VSIO_RRDS           CHAR(4)  INIT('RRDS'),        31420000
230      2 VSIO_SEQUENTIAL     CHAR(10) INIT('SEQUENTIAL'),  31430000
231      2 VSIO_DIRECT         CHAR(10) INIT('DIRECT  '),     31440000
232      2 VSIO_DYNAMIC        CHAR(10) INIT('DYNAMIC  '),     31450000
233      2 VSIO_INPUT          CHAR(6)  INIT('INPUT '),      31460000
234      2 VSIO_OUTPUT         CHAR(6)  INIT('OUTPUT'),      31470000
235      2 VSIO_INPUT_OUTPUT   CHAR(6)  INIT('UPDATE'),      31480000
236      2 (VSIO_RC_SUCCESS     INIT(0),      31490000
237          VSIO_RC_LOGIC_ERROR INIT(8),      31500000
238          VSIO_RC_END_OF_FILE INIT(9999),    31510000
239          VSIO_RC_UNKNOWN_COMMAND INIT(20),    31520000
240          VSIO_RC_DATASET_ALREADY_OPEN INIT(21),  31530000
241          VSIO_RC_DATASET_NOT_OPEN INIT(22),    31540000
242          VSIO_RC_ORGANIZATION_UNKNOWN INIT(23),   31550000
243          VSIO_RC_ACCESS_UNKNOWN INIT(24),      31560000
244          VSIO_RC_ORG_ACCESS_MISMATCH INIT(25),   31570000
245          VSIO_RC_MODE_UNKNOWN INIT(26),        31580000
246          VSIO_RC_MODE_UNSUPPORTED INIT(27),     31590000
247          VSIO_RC_DDNAME_BLANK INIT(28))        31600000
248          FIXED BINARY(15,0),                  31610000
249      2 (VSIO_FB_DUPLICATE_RECORD INIT(8),      31620000
250          VSIO_FB_KEY_SEQUENCE INIT(12),        31630000
251          VSIO_FB_RECORD_NOT_FOUND INIT(16),     31640000
252          VSIO_FB_NO_MORE_SPACE INIT(28),       31650000
253          VSIO_FB_READ_WITHOUT_START INIT(88))   31660000
254          FIXED BINARY(15,0),                  31670000
255 /*****31680000
256     THE VSIO_PARAMETER_BLOCK IS THE COMMUNICATION INTERFACE TO THE 31690000
257     THE ROUTINE. 31700000
258 *****/31710000
259 31720000
260      1 VSIO_PARAMETER_BLOCK STATIC, 31730000
261      2 VSIO_COMMAND        CHAR(8)  INIT(' '),      31740000
262      2 (VSIO_RETURN_CODE, 31750000
263          VSIO_VSAM_RC, 31760000
264          VSIO_VSAM_FUNCTION, 31770000
265          VSIO_VSAM_FEEDBACK) FIXED BINARY(15,0) INIT(0); 31780000
```

## MACRO SOURCE2 LISTING

```
266                                     31790000
267 /*****31800000
268             END OF VSAMIO COPY BOOK      31810000
269 *****/31820000
```

INCLUDED TEXT FOLLOWS FROM DD.MEMBER = SYSLIB .VSAMIOFB

```
270 /*****00000100
271                                     00000200
272     VV  VV  SSSSS  A    M    M  IIII  OOOOO  FFFFFFFF  BBBBBB  00000300
273     VV  VV  SS   SS   AAA  MM   MM  II   OO   OO  FF      BB   BB  00000400
274     VV  VV  SS           AA AA  MMM MMM  II   OO   OO  FF      BB   BB  00000500
275     VV  VV  SSSSS  AA   AA  MMMMMMMM  II   OO   OO  FFFFFF  BBBBBB  00000600
276     VV  VV           SS  AA   AA  MM M MM  II   OO   OO  FF      BB   BB  00000700
277     VV VV  SS   SS  AAAAAA  MM   MM  II   OO   OO  FF      BB   BB  00000800
278     VVV   SS   SS  AA   AA  MM   MM  II   OO   OO  FF      BB   BB  00000900
279     V     SSSSS  AA   AA  MM   MM  IIII  OOOOO  FF      BBBBBB  00001000
280                                     00001100
281 *****/00001200
282     THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS 00001300
283     ROUTINE, AND ARE USED TO COMMUNICATE CHARACTERISTICS FOR A SINGLE 00001400
284     VSAM DATASET. 00001500
285                                     00001600
286     WITH THE 2 EXCEPTIONS FOR RECORD LENGTH (TO ACCOMODATE VARIABLE 00001700
287     LENGTH RECORDS) AND RELATIVE RECORD (TO ACCOMODATE RELATIVE RECORD 00001800
288     DATASETS), THESE DATA NAMES MUST BE POPULATED PRIOR TO CALLING THE 00001900
289     ROUTINE TO OPEN THE DATASET AND MUST NOT THEN BE CHANGED UNTIL THE 00002000
290     DATASET HAS BEEN CLOSED. 00002100
291 *****/00002200
292                                     00002300
293     DECLARE 00002400
294     1 VSIO_FILE_BLOCK          STATIC, 00002500
295     2 VSFB_DDNAME              CHAR(8)  INIT(' '), 00002600
296     2 VSFB_ORGANIZATION        CHAR(4)  INIT(' '), 00002700
297     2 VSFB_ACCESS              CHAR(10) INIT(' '), 00002800
298     2 VSFB_MODE                CHAR(6)  INIT(' '), 00002900
299     2 (VSFB_RECORD_LENGTH,     00003000
300     VSFB_KEY_POSITION,        00003100
301     VSFB_KEY_LENGTH)         FIXED BINARY(15,0) INIT(0), 00003200
302     2 VSFB_FILE_STATUS         CHAR(1)  INIT('C'), 00003300
303     2 VSFB_RESERVED           CHAR(161); 00003400
304                                     00003500
305 /*****00003600
306             END OF VSAMIOFB COPY BOOK      00003700
```

MACRO SOURCE2 LISTING

307 \*\*\*\*\*/00003800

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

## SOURCE LISTING.

```

/*****
RRDSSSEQ - TESTS THE VSAMIO ROUTINE BY USING START AND READ COMMANDS
          ON AN RRDS CLUTER TO READ THE FILE IN SKIP-SEQUENTIAL
          MODE.
*****/
1  RRDSSSQ:
   PROCEDURE OPTIONS(MAIN);
2      ON ERROR
3      BEGIN;
4      ON ERROR SYSTEM;
5      PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');
6      PUT SKIP DATA;
7      PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');
8      END;
9      OPEN
   FILE(PRINTR) LINESIZE(133);
10     PRINT_AREA = 'RRDSSSEQ: READ RRDS SEQUENTIALLY (W/START)';
11     WRITE FILE(PRINTR) FROM(PRINT_LINE);
12     PRINT_AREA = '-----';
13     WRITE FILE(PRINTR) FROM(PRINT_LINE);
14     PRINT_AREA = ' ';
15     WRITE FILE(PRINTR) FROM(PRINT_LINE);
/*****
ESTABLISH PARAMETERS OF VSAM DATASET AND CALL ROUTINE TO OPEN IT
*****/
16     VSFB_DDNAME = 'RRDSF01';
17     VSFB_ORGANIZATION = VSIO_RRDS;
18     VSFB_ACCESS = VSIO_SEQUENTIAL;
19     VSFB_MODE = VSIO_INPUT;
20     VSFB_RECORD_LENGTH = 80;
21     VSFB_KEY_POSITION = 0;
22     VSFB_KEY_LENGTH = 0;
23     VSIO_COMMAND = VSIO_OPEN;
24     CALL VSAMIOP (VSIO_PARAMETER_BLOCK,
                  VSIO_FILE_BLOCK,
```

```

                                RECORD_IMAGE);
25     IF (VSIO_RETURN_CODEa= VSIO_RC_SUCCESS) THEN
26         DO;
27             CALL VSIO_ERROR;
28             RETURN;
29         END;

30     VSFB_KEY_LENGTH = 21;
31     VSIO_COMMAND = VSIO_START_EQUAL;
32     PRINT_AREA = 'STARTING KEY EQUAL TO 21';
33     WRITE FILE(PRINTR) FROM(PRINT_LINE);
34     CALL PROCESS_BLOCK;

35     VSFB_KEY_LENGTH = 51;
36     VSIO_COMMAND = VSIO_START_EQUAL;
37     PRINT_AREA = 'STARTING KEY EQUAL TO 51';
38     WRITE FILE(PRINTR) FROM(PRINT_LINE);
39     CALL PROCESS_BLOCK;

40     VSFB_KEY_LENGTH = 81;
41     VSIO_COMMAND = VSIO_START_NOTLESS;
42     PRINT_AREA = 'STARTING KEY NOT LESS THAN 81';
43     WRITE FILE(PRINTR) FROM(PRINT_LINE);
44     CALL PROCESS_BLOCK;

45     VSFB_KEY_LENGTH = 111;
46     VSIO_COMMAND = VSIO_START_NOTLESS;
47     PRINT_AREA = 'STARTING KEY NOT LESS THAN 111';
48     WRITE FILE(PRINTR) FROM(PRINT_LINE);
49     CALL PROCESS_BLOCK;

    /*****
    CALL ROUTINE TO CLOSE VSAM DATASET
    *****/

50     VSIO_COMMAND = VSIO_CLOSE;
51     CALL VSAMIOP (VSIO_PARAMETER_BLOCK,
                    VSIO_FILE_BLOCK,
                    RECORD_IMAGE);
52     IF (VSIO_RETURN_CODEa= VSIO_RC_SUCCESS) THEN
53         CALL VSIO_ERROR;

54     RETURN;

55     PROCESS_BLOCK:
        PROCEDURE;

```

```

42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
73
73
75
76
77
78
79
80
81
82
83
84
85
86
87

```

```
56      CALL VSAMIOP (VSIO_PARAMETER_BLOCK,      88
                VSIO_FILE_BLOCK,              89
                RECORD_IMAGE);                 90
57      IF (VSIO_RETURN_CODEa= VSIO_RC_SUCCESS) THEN 91
58          IF (VSIO_VSAM_FEEDBACK = VSIO_FB_RECORD_NOT_FOUND) THEN 92
59              DO;                             93
60                  PRINT_AREA = '*** NOT FOUND ***'; 94
61                  WRITE FILE(PRINTR) FROM(PRINT_LINE); 95
62                  RETURN;                       96
63              END;                             97
64          ELSE                                 98
64              DO;                             99
65                  CALL VSIO_ERROR;             100
66                  RETURN;                     101
67              END;                             102
                                                103
68      VSIO_COMMAND = VSIO_READ;               104
69      MORE_RECORDS = YES;                     105
70      RECORD_COUNTER = 0;                     106
                                                107
71      DO WHILE(MORE_RECORDS & RECORD_COUNTER < 5); 108
72          CALL READ_RR;                       109
73          IF (MORE_RECORDS) THEN             110
74              DO;                             111
75                  RRN_EDIT = VSFB_KEY_LENGTH; 112
76                  PRINT_AREA = 'RRN: ' || RRN_EDIT || ' DATA: ' || 113
                          RECORD_IMAGE_SCALAR; 114
77                  WRITE FILE(PRINTR) FROM(PRINT_LINE); 115
78              END;                             116
79          END;                                 117
                                                118
80      RETURN;                                 119
                                                120
81      END PROCESS_BLOCK;                      121
                                                122
82      READ_RR:                                123
          PROCEDURE;                           124
                                                125
          /*****                               126
          CALL ROUTINE TO READ NEXT RECORD FROM VSAM DATASET 126
          *****/                                126
                                                128
83      VSIO_COMMAND = VSIO_READ;             129
84      CALL VSAMIOP (VSIO_PARAMETER_BLOCK,    130
                VSIO_FILE_BLOCK,            131
                RECORD_IMAGE);              132
85      IF (VSIO_RETURN_CODEa= VSIO_RC_SUCCESS) THEN 133
```



```
86         IF (VSIO_RETURN_CODE = VSIO_RC_END_OF_FILE) THEN      134
87             MORE_RECORDS = NO;                                135
88         ELSE                                                    136
88             CALL VSIO_ERROR;                                    137
89         ELSE                                                    138
89             RECORD_COUNTER = RECORD_COUNTER + 1;              139
89                                                         140
90         RETURN;                                                141
89                                                         142
91     END READ_RR;                                              143
89                                                         144
92     VSIO_ERROR:                                              145
92     PROCEDURE;                                              146
93         PRINT_AREA = 'VSAMIO ERROR OCCURRED DURING ' ||      147
93             VSIO_COMMAND;                                      148
94         WRITE FILE(PRINTR) FROM(PRINT_LINE);                  149
95         PRINT_AREA = 'VSIO_RETURN_CODE = ' ||                 150
95             VSIO_RETURN_CODE;                                  151
96         WRITE FILE(PRINTR) FROM(PRINT_LINE);                  152
97         PRINT_AREA = 'VSIO_VSAM_RETURN_CODE = ' ||           153
97             VSIO_VSAM_RETURN_CODE;                            154
98         WRITE FILE(PRINTR) FROM(PRINT_LINE);                  155
99         PRINT_AREA = 'VSIO_VSAM_FUNCTION_CODE = ' ||         156
99             VSIO_VSAM_FUNCTION_CODE;                          157
100        WRITE FILE(PRINTR) FROM(PRINT_LINE);                  158
101        PRINT_AREA = 'VSIO_VSAM_FEEDBACK_CODE = ' ||          159
101            VSIO_VSAM_FEEDBACK_CODE;                          160
102        WRITE FILE(PRINTR) FROM(PRINT_LINE);                  161
103        PRINT_AREA = ' ';                                       162
103                                                         163
104        RETURN;                                                164
103                                                         165
105    END VSIO_ERROR;                                           166
103                                                         167
106    DECLARE                                                    168
106        PRINTR FILE OUTPUT RECORD SEQUENTIAL EXTERNAL         169
106            ENV(F CTLASA);                                     170
106                                                         171
107    DECLARE                                                    172
107        MORE_RECORDS          BIT(1),                            173
107        NO                     BIT(1) INIT('0'B),              174
107        RECORD_COUNTER        FIXED BINARY(15,0) INIT(0),      175
107        RRN_EDIT              PICTURE 'ZZ,ZZ9V',                176
107        YES                   BIT(1) INIT('1'B);              177
107                                                         178
108    DECLARE                                                    179
108        1 RECORD_IMAGE,                                         180
```

```

2 RECORD_KEY          CHAR(10),          181
2 RECORD_FIELDS       CHAR(70);          182
109 DECLARE           183
    RECORD_IMAGE_SCALAR   DEFINED RECORD_IMAGE 184
                                CHAR(80);      185
110 DECLARE           186
    1 PRINT_LINE,        187
    2 CARRIAGE_CONTROL   CHAR(1)  INIT(' '),  188
    2 PRINT_AREA         CHAR(132);         189
/*190
V V  V V  S S S S S  A  M  M  I I I I  O O O O O  190
V V  V V  S S  S S  A A A  M M  M M  I I  O O  O O  191
V V  V V  S S      A A A A  M M M M M  I I  O O  O O  192
V V  V V  S S S S S  A A  A A  M M M M M M  I I  O O  O O  193
V V  V V      S S  A A  A A  M M M M M  I I  O O  O O  194
V V  V V  S S  S S  A A A A A A  M M  M M  I I  O O  O O  195
V V V  S S  S S  A A  A A  M M  M M  I I  O O  O O  196
V      S S S S S  A A  A A  M M  M M  I I I I  O O O O O  197
*****197
THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS 197
ROUTINE.197
THE VSIO_PARAMETER_VALUES SUPPLY THE VALUES USED TO MOVE INTO 197
PARAMETER ENTRIES TO TAILOR THE ROUTINE TO A SPECIFIC DATASET AND 197
TO PROVIDE COMMANDS TO DRIVE THE ROUTINE.197
*****197
111 DECLARE           215
    1 VSIO_PARAMETER_VALUES  STATIC,        216
    2 VSIO_OPEN              CHAR(8)  INIT('OPEN  '),  217
    2 VSIO_CLOSE             CHAR(8)  INIT('CLOSE '),  218
    2 VSIO_READ              CHAR(8)  INIT('READ  '),  219
    2 VSIO_WRITE             CHAR(8)  INIT('WRITE '),  220
    2 VSIO_REWRITE          CHAR(8)  INIT('REWRITE '),  221
    2 VSIO_DELETE           CHAR(8)  INIT('DELETE '),  222
    2 VSIO_START_EQUAL      CHAR(8)  INIT('STARTEQ '),  223
    2 VSIO_START_NOTLESS   CHAR(8)  INIT('STARTGE '),  224
    2 VSIO_KSDS             CHAR(4)  INIT('KSDS '),  225
    2 VSIO_ESDS            CHAR(4)  INIT('ESDS '),  226
    2 VSIO_RRDS            CHAR(4)  INIT('RRDS '),  227
    2 VSIO_SEQUENTIAL       CHAR(10) INIT('SEQUENTIAL'),  228
    2 VSIO_DIRECT          CHAR(10) INIT('DIRECT '),  229

```

```

2 VSIO_DYNAMIC          CHAR(10)  INIT('DYNAMIC  '),      232
2 VSIO_INPUT            CHAR(6)    INIT('INPUT  '),      233
2 VSIO_OUTPUT           CHAR(6)    INIT('OUTPUT'),      234
2 VSIO_INPUT_OUTPUT     CHAR(6)    INIT('UPDATE'),      235
2 (VSIO_RC_SUCCESS      INIT(0),      236
   VSIO_RC_LOGIC_ERROR  INIT(8),      237
   VSIO_RC_END_OF_FILE  INIT(9999),    238
   VSIO_RC_UNKNOWN_COMMAND INIT(20),    239
   VSIO_RC_DATASET_ALREADY_OPEN INIT(21),    240
   VSIO_RC_DATASET_NOT_OPEN  INIT(22),    241
   VSIO_RC_ORGANIZATION_UNKNOWN INIT(23),    242
   VSIO_RC_ACCESS_UNKNOWN   INIT(24),    243
   VSIO_RC_ORG_ACCESS_MISMATCH INIT(25),    244
   VSIO_RC_MODE_UNKNOWN     INIT(26),    245
   VSIO_RC_MODE_UNSUPPORTED  INIT(27),    246
   VSIO_RC_DDNAME_BLANK     INIT(28))    247
                               FIXED BINARY(15,0),    248
2 (VSIO_FB_DUPLICATE_RECORD INIT(8),      249
   VSIO_FB_KEY_SEQUENCE     INIT(12),    250
   VSIO_FB_RECORD_NOT_FOUND  INIT(16),    251
   VSIO_FB_NO_MORE_SPACE    INIT(28),    252
   VSIO_FB_READ_WITHOUT_START INIT(88))    253
                               FIXED BINARY(15,0),    254

```

```

/*****
THE VSIO_PARAMETER_BLOCK IS THE COMMUNICATION INTERFACE TO THE
THE ROUTINE.
*****/

```

```

1 VSIO_PARAMETER_BLOCK  STATIC,      260
  2 VSIO_COMMAND        CHAR(8)     INIT(' '),      261
  2 (VSIO_RETURN_CODE,  262
     VSIO_VSAM_RC,      263
     VSIO_VSAM_FUNCTION, 264
     VSIO_VSAM_FEEDBACK) FIXED BINARY(15,0) INIT(0); 265

```

```

/*****
END OF VSAMIO COPY BOOK
*****/

```

```

VV  VV  SSSSS  A    M    M  IIII  OOOO  FFFFFFFF  BBBB  270
VV  VV  SS   SS   AAA  MM   MM  II   OO   OO  FF      BB   BB  270
VV  VV  SS   AA  AA  MMM  MMM  II   OO   OO  FF      BB   BB  270
VV  VV  SSSSS  AA  AA  MMMMMM  II   OO   OO  FFFFFF  BBBB  270
VV  VV  SS   SS  AA  AA  MM  M  MM  II   OO   OO  FF      BB   BB  270
VV  VV  SS   SS  AAAAAA  MM  MM  II   OO   OO  FF      BB   BB  270
VVV  SS   SS  AA  AA  MM  MM  II   OO   OO  FF      BB   BB  270

```

V SSSS AA AA MM MM IIII OOOO FF BBBB 270

\*\*\*\*\* 270

THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS ROUTINE, AND ARE USED TO COMMUNICATE CHARACTERISTICS FOR A SINGLE VSAM DATASET. 270

WITH THE 2 EXCEPTIONS FOR RECORD LENGTH (TO ACCOMODATE VARIABLE LENGTH RECORDS) AND RELATIVE RECORD (TO ACCOMODATE RELATIVE RECORD DATASETS), THESE DATA NAMES MUST BE POPULATED PRIOR TO CALLING THE ROUTINE TO OPEN THE DATASET AND MUST NOT THEN BE CHANGED UNTIL THE DATASET HAS BEEN CLOSED. 270

\*\*\*\*\*/ 270

112 DECLARE 293

1 VSIO\_FILE\_BLOCK STATIC, 294

2 VSFB\_DDNAME CHAR(8) INIT(' '), 295

2 VSFB\_ORGANIZATION CHAR(4) INIT(' '), 296

2 VSFB\_ACCESS CHAR(10) INIT(' '), 297

2 VSFB\_MODE CHAR(6) INIT(' '), 298

2 (VSFB\_RECORD\_LENGTH, 299

VSFB\_KEY\_POSITION, 300

VSFB\_KEY\_LENGTH) FIXED BINARY(15,0) INIT(0), 301

2 VSFB\_FILE\_STATUS CHAR(1) INIT('C'), 302

2 VSFB\_RESERVED CHAR(161); 303

/\* 305

END OF VSAMIOFB COPY BOOK 305

\*\*\*\*\*/ 305

113 END RRSSSQ; 196

## ATTRIBUTE AND CROSS-REFERENCE TABLE

| DCL NO. | IDENTIFIER           | ATTRIBUTES AND REFERENCES   |
|---------|----------------------|---|
| 110     | CARRIAGE_CONTROL     | IN PRINT_LINE,AUTOMATIC,UNALIGNED,INITIAL,STRING(1),CHARACTER   |
| 107     | MORE_RECORDS         | AUTOMATIC,UNALIGNED,STRING(1),BIT<br>69,71,73,87  |
| 107     | NO                   | AUTOMATIC,UNALIGNED,INITIAL,STRING(1),BIT<br>87   |
| 110     | PRINT_AREA           | IN PRINT_LINE,AUTOMATIC,UNALIGNED,STRING(132),CHARACTER<br>10,12,14,32,37,42,47,60,76,93,95,97,99,101,103     |
| 110     | PRINT_LINE           | AUTOMATIC,STRUCTURE<br>11,13,15,33,38,43,48,61,77,94,96,98,100,102  |
| 106     | PRINTR               | FILE,EXTERNAL,OUTPUT,RECORD,SEQUENTIAL,ENVIRONMENT(F CTLASA)<br>9,11,13,15,33,38,43,48,61,77,94,96,98,100,102 |
| 55      | PROCESS_BLOCK        | ENTRY,DECIMAL,FLOAT(SINGLE)<br>34,39,44,49  |
| 82      | READ_RR              | ENTRY,DECIMAL,FLOAT(SINGLE)<br>72   |
| 107     | ***** RECORD_COUNTER | AUTOMATIC,ALIGNED,INITIAL,BINARY,FIXED(15,0)<br>70,71,89,89   |
| 108     | RECORD_FIELDS        | IN RECORD_IMAGE,AUTOMATIC,UNALIGNED,STRING(70),CHARACTER  |
| 108     | RECORD_IMAGE         | AUTOMATIC,STRUCTURE<br>24,51,56,84  |
| 109     | RECORD_IMAGE_SCALAR  | AUTOMATIC,DEFINED,UNALIGNED,STRING(80),CHARACTER<br>76  |
| 108     | RECORD_KEY           | IN RECORD_IMAGE,AUTOMATIC,UNALIGNED,STRING(10),CHARACTER  |
| 1       | RRDSSSQ              | ENTRY,DECIMAL,FLOAT(SINGLE)   |
| 107     | RRN_EDIT             | AUTOMATIC,UNALIGNED,DECIMAL,PICTURE(ZZ,ZZ9V)<br>75,76   |

| DCL NO. | IDENTIFIER               | ATTRIBUTES AND REFERENCES  |
|---------|--------------------------|--|
|         | SYSPRINT                 | FILE, EXTERNAL<br>5, 6, 7  |
|         | VSAMIOP                  | EXTERNAL, ENTRY, DECIMAL, FLOAT(SINGLE)<br>24, 51, 56, 84  |
| 112     | VSFB_ACCESS              | IN VSIO_FILE_BLOCK, STATIC, UNALIGNED, INITIAL, STRING(10), CHARACTER<br>18  |
| 112     | VSFB_DDNAME              | IN VSIO_FILE_BLOCK, STATIC, UNALIGNED, INITIAL, STRING(8), CHARACTER<br>16   |
| 112     | VSFB_FILE_STATUS         | IN VSIO_FILE_BLOCK, STATIC, UNALIGNED, INITIAL, STRING(1), CHARACTER   |
| 112     | ***** VSFB_KEY_LENGTH    | IN VSIO_FILE_BLOCK, STATIC, ALIGNED, INITIAL, BINARY, FIXED(15, 0)<br>22, 30, 35, 40, 45, 75                       |
| 112     | ***** VSFB_KEY_POSITION  | IN VSIO_FILE_BLOCK, STATIC, ALIGNED, INITIAL, BINARY, FIXED(15, 0)<br>21   |
| 112     | VSFB_MODE                | IN VSIO_FILE_BLOCK, STATIC, UNALIGNED, INITIAL, STRING(6), CHARACTER<br>19   |
| 112     | VSFB_ORGANIZATION        | IN VSIO_FILE_BLOCK, STATIC, UNALIGNED, INITIAL, STRING(4), CHARACTER<br>17   |
| 112     | ***** VSFB_RECORD_LENGTH | IN VSIO_FILE_BLOCK, STATIC, ALIGNED, INITIAL, BINARY, FIXED(15, 0)<br>20   |
| 112     | VSFB_RESERVED            | IN VSIO_FILE_BLOCK, STATIC, UNALIGNED, STRING(161), CHARACTER  |
| 111     | VSIO_CLOSE               | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(8),<br>CHARACTER<br>50                                |
| 111     | VSIO_COMMAND             | IN VSIO_PARAMETER_BLOCK, STATIC, UNALIGNED, INITIAL, STRING(8),<br>CHARACTER<br>23, 31, 36, 41, 46, 50, 68, 83, 93 |
| 111     | VSIO_DELETE              | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(8),<br>CHARACTER                                      |
| 111     | VSIO_DIRECT              | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(10),<br>CHARACTER                                     |

| DCL NO. | IDENTIFIER                       | ATTRIBUTES AND REFERENCES  |
|---------|----------------------------------|--|
| 111     | VSIO_DYNAMIC                     | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(10), CHARACTER        |
| 92      | VSIO_ERROR                       | ENTRY, DECIMAL, FLOAT(SINGLE)<br>27, 53, 65, 88                                    |
| 111     | VSIO_ESDS                        | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(4), CHARACTER         |
| 111     | ***** VSIO_FB_DUPLICATE_RECORD   | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)       |
| 111     | ***** VSIO_FB_KEY_SEQUENCE       | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)       |
| 111     | ***** VSIO_FB_NO_MORE_SPACE      | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)       |
| 111     | ***** VSIO_FB_READ_WITHOUT_START | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)       |
| 111     | ***** VSIO_FB_RECORD_NOT_FOUND   | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)<br>58 |
| 112     | VSIO_FILE_BLOCK                  | STATIC, STRUCTURE<br>24, 51, 56, 84  |
| 111     | VSIO_INPUT                       | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(6), CHARACTER<br>19   |
| 111     | VSIO_INPUT_OUTPUT                | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(6), CHARACTER         |
| 111     | VSIO_KSDS                        | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(4), CHARACTER         |
| 111     | VSIO_OPEN                        | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(8), CHARACTER<br>23   |
| 111     | VSIO_OUTPUT                      | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(6), CHARACTER         |

| DCL NO. | IDENTIFIER                         | ATTRIBUTES AND REFERENCES  |
|---------|------------------------------------|--|
| 111     | VSIO_PARAMETER_BLOCK               | STATIC, STRUCTURE<br>24, 51, 56, 84  |
| 111     | VSIO_PARAMETER_VALUES              | STATIC, STRUCTURE  |
| 111     | ***** VSIO_RC_ACCESS_UNKNOWN       | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | ***** VSIO_RC_DATASET_ALREADY_OPEN | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | ***** VSIO_RC_DATASET_NOT_OPEN     | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | ***** VSIO_RC_DDNAME_BLANK         | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | ***** VSIO_RC_END_OF_FILE          | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)<br>86             |
| 111     | ***** VSIO_RC_LOGIC_ERROR          | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | ***** VSIO_RC_MODE_UNKNOWN         | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | ***** VSIO_RC_MODE_UNSUPPORTED     | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | ***** VSIO_RC_ORG_ACCESS_MISMATCH  | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | ***** VSIO_RC_ORGANIZATION_UNKNOWN | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | ***** VSIO_RC_SUCCESS              | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)<br>25, 52, 57, 85 |
| 111     | ***** VSIO_RC_UNKNOWN_COMMAND      | IN VSIO_PARAMETER_VALUES, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15, 0)                   |
| 111     | VSIO_READ                          | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(8),<br>CHARACTER                  |



| DCL NO. | IDENTIFIER               | ATTRIBUTES AND REFERENCES   |
|---------|--------------------------|---|
|         |                          | 68,83   |
| 111     | ***** VSIO_RETURN_CODE   | IN VSIO_PARAMETER_BLOCK, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15,0)<br>25,52,57,85,86,95 |
| 111     | VSIO_REWRITE             | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(8),<br>CHARACTER                   |
| 111     | VSIO_RRDS                | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(4),<br>CHARACTER<br>17             |
| 111     | VSIO_SEQUENTIAL          | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(10),<br>CHARACTER<br>18            |
| 111     | VSIO_START_EQUAL         | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(8),<br>CHARACTER<br>31,36          |
| 111     | VSIO_START_NOTLESS       | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(8),<br>CHARACTER<br>41,46          |
| 111     | ***** VSIO_VSAM_FEEDBACK | IN VSIO_PARAMETER_BLOCK, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15,0)<br>58                |
|         | VSIO_VSAM_FEEDBACK_CODE  | AUTOMATIC, ALIGNED, DECIMAL, FLOAT(SINGLE)<br>101   |
| 111     | ***** VSIO_VSAM_FUNCTION | IN VSIO_PARAMETER_BLOCK, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15,0)                      |
|         | VSIO_VSAM_FUNCTION_CODE  | AUTOMATIC, ALIGNED, DECIMAL, FLOAT(SINGLE)<br>99  |
| 111     | ***** VSIO_VSAM_RC       | IN VSIO_PARAMETER_BLOCK, STATIC, ALIGNED, INITIAL, BINARY, FIXED<br>(15,0)                      |
|         | VSIO_VSAM_RETURN_CODE    | AUTOMATIC, ALIGNED, DECIMAL, FLOAT(SINGLE)<br>97  |
| 111     | VSIO_WRITE               | IN VSIO_PARAMETER_VALUES, STATIC, UNALIGNED, INITIAL, STRING(8),                                |

| DCL NO. | IDENTIFIER | ATTRIBUTES AND REFERENCES  |
|---------|------------|--|
| 107     | YES        | CHARACTER<br>AUTOMATIC, UNALIGNED, INITIAL, STRING(1), BIT<br>69 |

AGGREGATE LENGTH TABLE

| STATEMENT NO. | IDENTIFIER            | LENGTH IN BYTES |
|---------------|-----------------------|-----------------|
| 110           | PRINT_LINE            | 133             |
| 108           | RECORD_IMAGE          | 80              |
| 112           | VSIO_FILE_BLOCK       | 196             |
| 111           | VSIO_PARAMETER_BLOCK  | 16              |
| 111           | VSIO_PARAMETER_VALUES | 158             |

STORAGE REQUIREMENTS.  
-----

THE STORAGE AREA FOR THE PROCEDURE LABELLED RRDSSSQ IS 504 BYTES LONG.

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

THE STORAGE AREA (IN STATIC) FOR THE PROCEDURE LABELLED PROCESS\_BLOCK IS 276 BYTES LONG.

THE STORAGE AREA (IN STATIC) FOR THE PROCEDURE LABELLED READ\_RR IS 176 BYTES LONG.

THE STORAGE AREA (IN STATIC) FOR THE PROCEDURE LABELLED VSIO\_ERROR IS 256 BYTES LONG.

THE PROGRAM CSECT IS NAMED RRDSSSQ AND IS 2086 BYTES LONG.

THE STATIC CSECT IS NAMED RRDSSSQ AND IS 6000 BYTES LONG.

\*STATISTICS\*      MACRO RECORDS =      307, SOURCE RECORDS =      309, PROG TEXT STMENTS =      113, OBJECT BYTES =      2086

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 PROCEDURE PROCESS\_BLOCK

|              |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |
|--------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| OFFSET (HEX) | 0000 | 0048 | 0068 | 0074 | 0080 | 0080 | 0090 | 00A8 | 00AE | 00B2 | 00B2 | 00BC | 00C2 | 00C6 | 00CC | 00D6 | 00DC | 010C | 0116 | 0122 | 0122 |
| STATEMENT NO | 55   | 56   | 57   | 58   | 59   | 60   | 61   | 62   | 63   | 64   | 65   | 66   | 67   | 68   | 69   | 70   | 71   | 72   | 73   | 74   | 75   |

|              |      |      |      |      |      |      |
|--------------|------|------|------|------|------|------|
| OFFSET (HEX) | 0142 | 0174 | 018C | 018C | 0190 | 0196 |
| STATEMENT NO | 76   | 77   | 78   | 79   | 80   | 81   |

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN PROCEDURE READ\_RR

|              |      |      |      |      |      |      |      |      |      |      |
|--------------|------|------|------|------|------|------|------|------|------|------|
| OFFSET (HEX) | 0000 | 0034 | 003A | 005A | 0066 | 0072 | 0080 | 008E | 009E | 00A4 |
| STATEMENT NO | 82   | 83   | 84   | 85   | 86   | 87   | 88   | 89   | 90   | 91   |

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 | 92   | 93   | 94   | 95   | 96   | 97   | 98   | 99   | 100  | 101  | 102  | 103  | 104  | 105  |

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN PROCEDURE RRDSSSQ

|              |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |
|--------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| OFFSET (HEX) | 0000 | 00CC | 00DA | 00E4 | 00F0 | 0108 | 0114 | 012C | 0138 | 0150 | 0156 | 015C | 015C | 0162 | 0168 | 016E | 0174 | 017A | 0196 | 01A2 | 01A2 |
| STATEMENT NO | 1    | 2    | 9    | 10   | 11   | 12   | 13   | 14   | 15   | 16   | 17   | 18   | 19   | 20   | 21   | 22   | 23   | 24   | 25   | 26   | 27   |

|              |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |
|--------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| OFFSET (HEX) | 01AC | 01B2 | 01B2 | 01B8 | 01BE | 01CA | 01E2 | 01EC | 01F2 | 01F8 | 0204 | 021C | 0226 | 022C | 0232 | 023E | 0256 | 0260 | 0266 | 026C | 0278 |
| STATEMENT NO | 28   | 29   | 30   | 31   | 32   | 33   | 34   | 35   | 36   | 37   | 38   | 39   | 40   | 41   | 42   | 43   | 44   | 45   | 46   | 47   | 48   |

|              |      |      |      |      |      |      |      |
|--------------|------|------|------|------|------|------|------|
| OFFSET (HEX) | 0290 | 029A | 02A0 | 02BC | 02C8 | 02D2 | 02D8 |
| STATEMENT NO | 49   | 50   | 51   | 52   | 53   | 54   | 113  |

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 97, 99, 101.

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.

RRDSSSEQ: READ RRDS SEQUENTIALLY (W/START)

STARTING KEY EQUAL TO 21

|      |    |       |            |                      |                        |                 |    |
|------|----|-------|------------|----------------------|------------------------|-----------------|----|
| RRN: | 21 | DATA: | 1033846021 | CHERYL I TUCKER      | 660 SHORE ROAD         | LOUISVILLE      | KY |
| RRN: | 22 | DATA: | 9998019022 | HANNAH F QUIMBY      | 6151 MAIN COURT        | PHOENIX         | AZ |
| RRN: | 23 | DATA: | 1168050023 | TAMMY J FRANKLIN     | 5226 ROSA LINDA ROAD   | LOUISVILLE      | KY |
| RRN: | 24 | DATA: | 9971074024 | TAMMY M HARMON       | 243 KINGS RIDGE STREET | GREENVILLE      | SC |
| RRN: | 26 | DATA: | 1511914026 | SAMANTHA J TEMPLETON | 1534 SUN MEADOW AVENUE | FORT LAUDERDALE | FL |

STARTING KEY EQUAL TO 51

\*\*\* NOT FOUND \*\*\*

STARTING KEY NOT LESS THAN 81

|      |    |       |            |                  |                          |              |    |
|------|----|-------|------------|------------------|--------------------------|--------------|----|
| RRN: | 81 | DATA: | 3851331081 | CLIFF J DRAKE    | 169 SHORE AVENUE         | CHICAGO      | IL |
| RRN: | 82 | DATA: | 9912384082 | CRAIG O LABROIE  | 8021 MILL MOUNTAIN PLACE | MURFREESBORO | TN |
| RRN: | 83 | DATA: | 4077702083 | ROLAND P RODGERS | 827 MEADOW STREET        | TULSA        | OK |
| RRN: | 84 | DATA: | 9993285084 | JACK J SCHWAB    | 250 BUCKLEY PLACE        | CHICAGO      | IL |
| RRN: | 86 | DATA: | 4197550086 | TROY S POWERS    | 1707 BRIDGE STREET       | ENGLEWOOD    | CO |

STARTING KEY NOT LESS THAN 111

|      |     |       |            |                   |                           |                |    |
|------|-----|-------|------------|-------------------|---------------------------|----------------|----|
| RRN: | 121 | DATA: | 5582803121 | ANGELA J ERWIN    | 2709 BIENVILLE STREET     | NEWPORT BEACH  | CA |
| RRN: | 122 | DATA: | 9918369122 | PETER F ALEXANDER | 5500 KNICKERBOCKER AVENUE | BINGHAMTON     | NY |
| RRN: | 123 | DATA: | 5731895123 | BETTY H BOWERS    | 3787 WINDWOOD PLACE       | SALT LAKE CITY | UT |
| RRN: | 124 | DATA: | 9964671124 | PETER L SCHAEFER  | 358 ATOLL STREET          | PASADENA       | CA |
| RRN: | 126 | DATA: | 5877149126 | ROBERT P GLOVER   | 6394 PARSON ROAD          | NORWALK        | CT |