Problem Determination under VSE/ESA

1. Analysis of Recent Problems
   - Tracing Hardware Errors..... Page 2
   - VSAM File Access Conflicts ..... Page 11
   - Loop during CICS Initialization..... Page 15
   - Correlate Dump with COBOL Source..... Page 19
   - Stand-alone Dump to Disk ..... Page 24
   - Sharing Physical Tape Units ..... Page 31
   - Installing Service from Disk..... Page 32
2. PD/PSI Console Commands:..... Page 33

   - ALTER / DSPLY
   - CANCEL cuu,FORCE
   - DTRIATTN
   - DOIO
   - GETVIS
   - IESINSRT
   - LOCATE
   - MAP
   - NOAVR
   - QUERY DSPACE
   - REIPL
   - SHOW
   - SIR
   - STACK
   - STATUS
   - System Level
   - TAPE
   - TIME
   - VOLUME (tape)

VM/ESA and VSE/ESA
Technical Conference
Orlando, Florida
May 31st - June 3rd, 2000

Charles E. Olsen

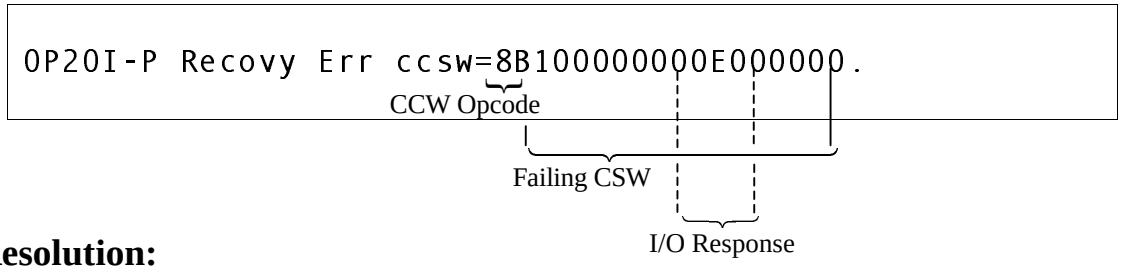
RETURN TO INDEX

Tracing Hardware Errors

Symptom:

Customer having a problem writing 1600 BPI IRS and BANK tapes. No software changes have been made recently. He has upgraded from a 9021 to a 7060H30. When an assign is typed in for a tape drive and followed with a mode of C0,he receives following message. Will not write in C0 mode.

```
BG 0001 1Q47I BG JACDITCT 33432 FROM VSEPROD(K00B) U='12500Óa'
BG // JOB JACDITCT JOB TO LOAD CARD TO TAPE
BG DITTO OUTPUT=SYS030,VOLSER=TAPTST,OUTFILE='TAPETEST'
BG DITTO ASCII=NO
BG 0014 0P20I P RECOVY ERR SYS030=180
        CCSW=8B100000000E000000  CCB=566EF4
        SNS= 0844042C 04422D00 00080000 0009E406 29917000 001A0029
BG JOB TERMINATED DUE TO RETURN CODE
BG EOJ JACDITCT MAX.RETURN CODE=0016
   DATE 02/23/2000, CLOCK 11/54/44, DURATION 00/00/05
BG 0001 1Q34I BG WAITING FOR WORK
```



Resolution:

"*We should close this problem. The hardware repairman used the SDAID dump to locate a failed piece of the controller for the 3420 tape drives. After replacing the part the problem no longer existed.*"

<u>Another example, this time from VSE-L newsgroup:</u>

```
FA 0014 0P64I R  MAINT REQD SYS093=604
        CCSW=6330D7588802000000 CCB=D75840 SK=000080000000
        SNS= 00003000 2837CF0A ED008002 09AA4004 23000BE3 0001E35B
             05100200 AE282900
```

Resolution:

"*I just got off the phone with IBM. They asked me if we were using a spare drawer, to which I was able to give my favorite answer, "Um...what's that?" But, it's OK because the controller had already phoned home and told IBM that one of the drawers had a bad battery. IBM will replace it on Monday and while the CE is here, I'll find out more about this spare-drawer business.*"

Tracing Hardware Errors

Analysis Procedure:

Most messages from 0P00 – 0P90I are triggered by a hardware response (or lack of response) to a software initiated I/O request. Unless the cause is inherently obvious, it is helpful to obtain an SDAID I/O trace, which will show the sequence of commands being issued, and the hardware response. Follow these steps:

1. Initiate SDAID tracing, either with the following batch job, or via Attention Routine (AR) commands:

```
// EXEC SDAID
   OUTDEV T=<cuu1>
   TRACE IO   AREA=ALL UNIT=<cuu2> OUTPUT=(TOD,
CCWD=64)
   TRACE SIO  AREA=ALL UNIT=<cuu2> OUTPUT=(TOD,
CCWD=64)
/*
```

   Where **<cuu1>** is a scratch tape for SDAID output, and **<cuu2>** is the tape he wants to write to in 1600 BPI mode.

2. Mount a scratch tape for SDAID output on **<cuu1>**.
3. Setup the failing condition.
4. Start SDAID tracing by issuing the "`STARTSD`" (or "`STRTSD`") AR command.
5. After the failure, terminate SDAID tracing by issuing the "`ENDSD`" AR command. This should rewind and unload the SDAID trace output tape.
6. Print the SDAID tape using "`DOSVSDMP`" (option 4).
7. Locate the SDAID trace entry immediately preceding the failing error message, and show it to your hardware support person.

For VTAM SNA , the address of the controller should be traced. Generally, however, you should use the VTAM Generalized Trace Facility (GTF). See "*VTAM Operations*" (SC31-6495-01) under "2.1.87 *MODIFY TRACE Command*" or "*VTAM Resource Definition Reference*" (SC31-6498-01) under "4.4.111 *TRACE for Buffer, I/O, NCP Line, SIT, or SMS Traces*"

Potential Problems using SDAID:

1. SDAID cannot be run with Turbo Dispatcher when more than one processor is active. Use the **SYSDEF TD,STOP=ALL**. Starting with VSE/ESA 2.5, SDAID can be invoked in a multi-processor environment (up to 10 CPUs). All in-bound events are processed in parallel mode, that is none of the CPUs are stopped. The exception is the small period of time when the record is written to the SDAID buffer.

2. SDAID uses the SDAREA as work space. (Defined via **SYS SDSIZE**=nn**K** IPL command). Some vendor products use the SDAREA, thus preventing SDAID from being initialized

3. If output tape fills up, or has intervention required (or an I/O error), SDAID places the system in a wait state with the following PSW: 000A0000 00EEEEEE. In this case:
   a) Replace the tape (if required)
   b) Make the tape unit ready
   c) Press the external interrupt key. Now, of course, there is no external interrupt key any more. This will be an option through the hardware console.

Tracing Hardware Errors

CP TRACE:

If you are running VSE/ESA under VM/ESA, are unable to run SDAID, and the amount of trace data is small, you may use the CP TRACE command.

1. The trace output goes straight to the VSE/ESA virtual machine console, so be sure to spool your console before you begin.
2. Does not have as much flexibility as SDAID, including the ability to see the data transmitted by / retrieved by CCW commands.
3. One advantage: It translates the CCW opcodes into clear text.
4. It is easy to lock your VSE machine out, so specify "NORUN" or be prepared to monitor closely.
5. Enter "* CP TRACE END ALL" to terminate trace. If the trace has locked up your console, and you cannot get into VSE, try "**#CP** TRACE END ALL".

Executes this command for each trace "hit"
In this case, it displays the CSW in low core.

```
* CP TRACE IO 141 INST INT CCW NORUN CMD D T40
AR 0015 HCPTRI1027I An active trace set has turned RUN off.
AR 0015 1I40I  READY
```

(screen flips to CP mode)

```
 → 0001CE18' TSCH  B2352000 >> 0001CC50    CC 0    SCH 0002    DEV 0141
            CCWA 0007FA30   DEV STS 0C  SCH STS 00  CNT 0000
            KEY 0   FPI 40  CC 0   CTLS 4007
V00000040  000824F0 0C000000 0007FA18 00000000 06 *...0.............* L0000004

B
 → 0001BD22' SSCH  B2332000    0001BD98    CC 0    SCH 0002    DEV 0141
            CPA 0006BD68   PARM 00003378    KEY 0  FPI 00  LPM F0
CCW    0006BD68  6306A8E0 64000010   0000  63...... 60000010
 IDAL            006BE350
 EXTENT                  C0C01000 00000040 0A090000 0A0C000E
CCW    0006BD70  4706A8E4 64000010   0008  47...... 60000010
 IDAL            006BE360
 LOCATE RECORD           01000001 0A0A000A 022E000A 054D0000
CCW    0006BD78  0506A8E8 04001000   0010  05...... 04001000
 IDAL            006BE4A8                  3F2284A8
 IDAL            006BE800                  3F228800
 IDAL            006EF000                  2173F000
V00000040  0006A8E8 0C000000 0006BD68 00000000 06 *..yY.............* L0000004

B
 → 0001BB82' TPI   B2360000 >> 000000B8    CC 1
            SCH 0002    DEV 0141       PARM 00003378
V00000040  0006A8E8 0C000000 0006BD68 00000000 06 *..yY.............* L0000004
```

Tracing Hardware Errors

TRSOURCE / TRSAVE:

Sometimes, if you are running VSE/ESA as a guest under VM/ESA, it is meaningful to use `TRSOURCE` and `TRSAVE` to also trace the VM I/O to the real device. See "*CP Command and Utility Reference*" (SC24-5519-04) The TRSOURCE trace can be run concurrently with an SDAID trace.

1. Determine the real device address for the virtual device defined to the VSE machine:

```
* cp q v 140
AR 0015 DASD 0140 3390 CC3U3B R/W    1112 CYL ON DASD  7A0E SUBCHANNEL =
0001
AR 0015 1I40I  READY
```

2. Define the trace sets and traces to be run using the `TRSOURCE` CP command. This is a privilege class "C" command, so cannot be issued from the VSE/ESA virtual machine or from a normal "G" class user. Define where the trace output should be saved using the `TRSAVE` CP command.

```
trsource id iotrace type io dev 7a0e
trsave id iotrace dasd to olsen name vsetest2 size 8 keep 1
trsource enable iotrace
      .
      .     (Failing event)
      .
trsource disable iotrace
```

3. Set up event to be traced.
4. Start TRSOURCE trace via the `TRSOURCE ENABLE` CP command.
5. After the failing event, end the trace using the `TRSOURCE DISABLE` CP command
6. View the trace data in the CP System Trace File (TRFiles) using the `TRACERED` command. See "*VM/ESA Dump Viewing Facility*" (SC24-5530-04).

I/O Tracing for CMS/VSAM

Some have wondered why the CP TRACE I/O doesn't show CMS/VSAM I/O activity. VSAM (or, in this case, CMSBAM for tape) issues its I/Os via SVC0. DMSXCP then executes a Diagnose x'A8'. So a real I/O is never executed within the virtual machine.
To setup an I/O trace within a virtual machine, you should issue:

TRACE DIAG A8 PSWA xxxx.yyy CMD D G

where the PSWA range covers the CMSDOS (or CMSVSAM) segment.

Also, see II05609 for TRSOURCE / TRSAVE instructions to trace VM/ESA CP I/O commands.

Tracing Hardware Errors

CCB: (Command Control Block) used by application to request I/O processing from VSE

```
00009800 0C000124 00631280 00631298
```

+x'00' **Residual Count** (two bytes)

+x'02' **Communications Byte 1**

x'80' Traffic Bit (Set at Channel End or Device End) (Set by VSE)
x'40' End of File (Set by VSE)
x'20' Unrecoverable I/O Error (Set by VSE)
x'10' Accept Unrecoverable Errors (Set by application)
x'08' Return Data Checks (Set by application)
x'04' Post at Device End (Set by application)
x'02' Return Data Check / Read Check (Set by application)
x'01' User Error Routine (Set by application)

+x'03' **Communications Byte 2**

x'80' Data Check in count area (Set by VSE)
x'40' Track Overrun (Set by VSE)
x'20' End of Cylinder (Set by VSE)
x'10' Data Check (Set by VSE)
x'08' No Record Found (Set by VSE)
x'04' Retry No-record-found (Set by application)
x'02' Verify Error (Set by VSE)
x'01' Command Chain (retry) (Set by VSE)

+x'04' **CSW Status Bytes** (Two bytes, see CSW) (Set by VSE)

+x'06' **Symbolic Unit** (SYSUNIT) (Set by application)

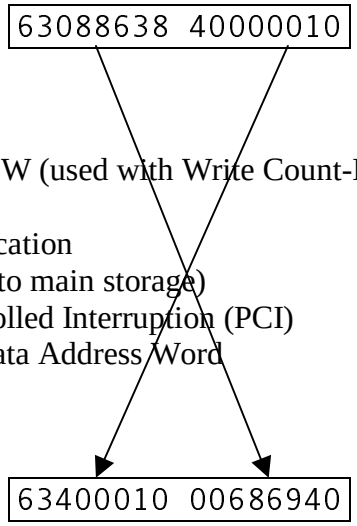
If x'01xx': Programmer unit (e.g. x'0124' = SYS036)
If x'00xx': 0000 = SYSRDR
0001 = SYSIPT
0002 = SYSPCH
0003 = SYSLST
0004 = SYSLOG
0005 = SYSLNK
0006 = SYSRES
(and so forth)

+x'08' **Pointer to first CCW** (Set by application)

+x'0C' **Pointer to last CCW executed** (Set by VSE)

Tracing Hardware Errors

CCW: (Channel Command Word, specific I/O commands)



Format0 (original) CCW:

+x'00' CCW Command Code
+x'01' Data Address (three bytes)
+x'04' Flags
x'80' Use address portion of next CCW (used with Write Count-Key-Data)
x'40' Chain to next sequential CCW
x'20' Suppress Incorrect Length Indication
x'10' Skip (suppress transfer of data to main storage)
x'08' Cause Channel Program Controlled Interruption (PCI)
x'04' Address refers to an Indirect Data Address Word
+x'05' Unused
+x'06' Byte Count (two bytes)

Format1 CCW:

+x'00' CCW Command Code
+x'01' Flags (see above)
+x'02' Byte Count (two bytes)
+x'04' Data Address (four bytes)

CCW Op-codes commonly-used for DASD:

Count-Key-Data (CKD):

```
    07   Seek
    1B   Seek Head
    23   Set Sector
    31   Search ID Equal
    08   TIC
    06   Read Data
    05   Write Data
.   86   Multi-track read
```

Extended-Count-Key-Data (ECKD):

```
    63   Define Extent
    47   Locate Record
    12   Read Count
    06   Read Data
    05   Write Data
.   86   Multi-track read
```

CCW Op-codes commonly used for printers:

```
    09   Space one line after printing
    0B   Space one line immediately
    63   Load FCB
    89   Skip to channel one after printing
    8B   Skip to channel one immediately
```

CCW Op-codes commonly used for tape:

```
    01   Write                  2F   Backspace file
    02   Read forward           3F   Forward space file
    07   Rewind                 9F   Lead display
    0F   Rewind and Unload      DB   Modeset
```

Sense CCW: 04

Tracing Hardware Errors

CSW: (Channel Status Word) passed to hardware by VSE via SSCH command.
Returned by hardware when I/O complete.

```
0000E258 0C000000
```

+x'00' Multi-use byte
- x'---- . . . .' Protection Key
- x'08' Unused
- x'04' Logout pending
- x' . . . . . --' Condition code from Start I/O command

+x'01' CCW Address (three byes, real address of last CCW executed)

+x'04' Unit Status

x'0C00' indicates normal completion (Channel End + Device End)
x'0D00' indicates an exceptional condition (CE + DE + Unit Exception)
x'0E00' indicates an error condition (CE + DE + Unit Check)

- x'80' Attention interrupt
- x'40' Status Modifier
- x'20' Control Unit End
- x'10' Busy
- x'08' Channel End
- x'04' Device End
- x'02' Unit Check
- x'01' Unit Exception

+x'05' Channel Status
- x'80' Program Controlled Interrupt
- x'40' Incorrect Length
- x'20' Program Check
- x'10' Protection Check
- x'08' Channel Data Check
- x'04' Channel Control Check
- x'02' Interface Control Check
- x'01' Chaining Check

+x'06' Residual Bye count (Difference between bytes requested and number read)

Tracing Hardware Errors

SDAID trace output:

Start Sub-channel
Software request out to hardware

Hardware response

Channel End (CE)

Device End (DE)

```
START OF SDAID TRACE

SSCH-1 0  BG   UNIT=0181
TOD = 2000.054 14.24.42.922
CCB= 0005E7F8  00001000 00002009 0005E8D0 00000000
CCW= 00067460  C30638E5 60000001
CCW= 00067468  0805E8D0
CCW= 0005E8D0  0705E8D0

I/O    0  BG   UNIT=0181           CSW =1005E8D8 08000001
TOD = 2000.054 14.24.42.922
CCB= 0005E7F8  00001000 00002009 0005E8D0 00000000
CCW= 0005E8D0  0705E8D0 20000001 **** NO DATA TRANSFERED ****
----IRB-----
0001D320 10404045 0005E8D8 08000001 00400000 00000000 00000000 00000000     *.  ...YQ..... ...................*
0001D340 00000000 00000000 00000000 00000000 00000000 00000000 00000000     *................................*

I/O    00 AR   UNIT=0181           CSW =00000000 04000000
TOD = 2000.054 14.24.44.062
CCB CAN NOT BE FOUND
----IRB-----
0001D320 10400003 00000000 04000000 00400000 00000000 00000000 00000000     *. .......... ...................*
0001D340 00000000 00000000 00000000 00000000 00000000 00000000 00000000     *................................*

SSCH-1 0  BG   UNIT=0181
TOD = 2000.054 14.24.44.402
CCB= 000275D0  00001400 0000081D 000274F8 20027580
CCW= 000274F8  03027F38 60000001
CCW= 00027500  04027F38 20000020
---CCW DATA---
00027F38                                                 00000000 00000000     *                        ........*
00027F40 00000000 00000000 00000000 00000000 00000000 00000000                 *........................        *
      .
      .
      .
```

Tracing Hardware Errors

```
SSCH-1 0  BG   UNIT=0181
TOD = 2000.054 14.24.47.516
CCB= 0007F8D0  00001600 0000211E 0007BF28 00000000
CCW= 00067460  C30638E5 60000001
CCW= 00067468  03061A67 60000001
CCW= 00067470  8B000000 60000001
CCW= 00067478  030638E5 00000001

I/O    0  BG   UNIT=0181           CSW =1006747B 0E000000
TOD = 2000.054 14.24.47.528
CCB= 0007F8D0  00001600 0000211E 0007BF28 00000000
CCW= 0007BF28  010718F0 24000050 IDAL= 000718F0  017794B5
---CCW DATA---
017794B5                                          E3C8C                 *                    THIS IS A D*
017794C0 E4D4D4E8 40C6C9D3 C5404040 40404040 40404040 4040404           *UMMY FILE                      *
017794E0 40404040 40404040 40404040 40404040 40404040 40                *                               *
---CCW CHAIN NOT FOUND. LAST EXECUTED CCW FOLLOWS:
CCW= 00067470  8B000000 60000001
----IRB-----
0001D320 10404017 00067478 0E000000 00400000 0045D658 00000000 00000000 00000000     *.  ......... ....O.............*
0001D340 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000     *................................*

SSCH-1 0  BG   UNIT=0181
TOD = 2000.054 14.24.47.528
CCB= 0001F19A  00081400 0000081D 4001F1B0 8001F1B8
CCW= 0001F1B0  0401F1F0 20000020
---CCW DATA---
0001F1F0                                  00000000 00000000 00000000 00000000     *                 .................*
0001F200 00000000 00000000 00000000 00000000                                      *.................               *

I/O    0  BG   UNIT=0181           CSW =0001F1B8 0C000008
TOD = 2000.054 14.24.47.528
CCB= 0001F19A  00081400 0000081D 4001F1B0 8001F1B8
CCW= 0001F1B0  0401F1F0 20000020
---CCW DATA---
0001F1F0                                  0844042C 04422D00 00080000 0009E406     *                 ..............U.*
0001F200 29917000 001A0029                                                        *.j......                       *
----IRB-----
0001D320 00404007 0001F1B8 0C000008 00400000 00000000 00000000 00000000 00000000     *.  ...1..... ..................*
0001D340 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000     *................................*
END OF SDAID OUTPUT
```

CE + DE + Unit Check

Just like a PSW, the CSW always points PAST the last CCW executed. So, the last CCW executed was an x'8B'.

Software generally responds to a Unit Check by issuing a Sense command (x'04').

The Sense data should match the msg0Pxx. This is what you show the hardware CE.

VSAM File Access Conflicts

Symptom:

"*Unexplained errors running VSAM operations. File being accessed is SHR(2), and not shared between multiple VSE systems.*"

```
F4 004 4228I FILE KSDS       OPEN  ERROR X'A8'(168) CAT=UCAT
(0CSHR--5) FILE ALREADY OPEN IN ANOTHER PARTITION
```

```
F4 004 4228I FILE KSDS       OPEN  ERROR X'FF'(255) CAT=UCAT
(0CSHR--8) LOCK RETURN CODE X'0C'
```

```
DELETE (CICS.CSD) CLUSTER PURGE
  IDC3028I DATA SET IN USE
  IDC3009I ** VSAM CATALOG RETURN CODE IS 184 - REASON CODE IS IGG0CLCX-4
  IDC0551I **ENTRY CICS.CSD NOT DELETED
  IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 8
```

Resolution:

File is open in another partition, but you don't know which one. Use the (new) LOCK TRACE command to ascertain which partition has the file open (See "*Hints and Tips for VSE*, p. 31) The "owner element" immediately prior to the msg4228I on the console will show the task id of the owner of the lock. If you do not receive a console msg (e.g. IDCAMS error), you can still locate the appropriate trace entry by:

1. Locktab entries and Owner elements are matched pairs.
2. If the error produces msg4228I rcx'A8', then it is easy. The Locktab and Owner entry is immediately prior to the msg4228I in the console log.
3. Having located the failing locktab entry, move to the Owner element immediately following. +5 (counting from zero) contains the task id of the owner of the file in question

```
lock trace=f4
AR 0015 1I40I  READY
    • • • (skip several pages of trace data)
F4 0025 LOCKTAB ENTRY
V02CCCAE0  02CCCB10 00000000 E5C3E3E2 F2F2F000 * öô     VCTS220 *  R0096DAE0
V02CCCAF0  00560001 04C00000 02CCCAC0 02CCCB20 * î   {   ö-{ öô *  R0096DAF0
F4 0025 OWNER ELEMENT
V02CCCB10  00000000 00210001 00000000 00000000 *                *  R0096DB10
F4 0025 LOCKTAB ENTRY
V02CCCAE0  02CCCB10 00000000 E5C3E3E2 F2F2F000 * öô     VCTS220 *  R0096DAE0
V02CCCAF0  00560001 04C00000 02CCCAC0 02CCCB20 * î   {   ö-{ öô *  R0096DAF0
F4 0025 OWNER ELEMENT
V02CCCB10  00000000 00210001 00000000 00000000 *                *  R0096DB10
F4 0004 4228I FILE KSDS       OPEN  ERROR X'A8'(168) CAT=IJSYSCT
   (OPNAB-15) DATASET ALREADY BEING LOADED BY ANOTHER ACB
```

Task id: 21 = BG, 22 = F1, 23 = F2 … 2A = F9, 2B = FA, 2C = FB, 30 – 99 are assigned to dynamic partitions or sub-tasks. (Use STATUS command).

VSAM File Access Conflicts

4. In cases where no console error message is issued (e.g. IDCAMS DELETE “data set in use”), the failing trace entry will be more difficult to find. For an IDCAMS DELETE, it will be towards the end of all the trace entries for this job.
5. Scan the trace entries ignoring all locktab entries which do not contain "V" plus the volume name of the user catalog where the file in question resides.
6. Ignore all locktab entries in the following format: **"V"<volser>nnnnnnmmmm**, where nnnnnn = '000000' or '000001'.
7. We are looking for a locktab entry in the following format: **“V”<volser>cccccc0001” :**. “cccccc” will be the control interval number of the file in question within the user catalog. In our case, the user catalog resides on volume CTS220, and the file is ci# x'000056': e.g. **x’E5C3E3E2 F2F2F000 00560001’ (VCTS220..î..).** “0001” may also be “0000”.
8. In some cases, there will be two locktab entries following each other, that match this format. This would be for a KSDS file, and the first one describes the data component, and the second, the index.
9. Terminate the lock trace using “**LOCK TRACE=END**”

Once you know which partition has the file open, use the LOCATE command to find the AMBL:_

```
locate bg,'test.ksds
AR 0015   MATCH FOUND AT 004044B0
V004044B0 E3C5E2E3 4BD2E2C4 E24BC3D3 E4E2E3C5 16 *TEST.KSDS.CLUSTE* R00D7F4B0
V004044C0 D9404040 40404040 40404040 40404040 16 *R               * R00D7F4C0
AR+0015
15 e
```

Backup x’98’. The VSAM AMBL always starts with ‘110000C4’. AMBL+x’14 points at the ACB.

```
show bg,404418.40
AR 0015   DATA  FOUND AT 00404400
V00404410 ........ ........ 110000C4 004045E0 16 *.......\...D. .\* R00D7F41
V00404420 00404780 00402D18 00404418 00400494 16 *. ... ... ... .m* R00D7F42
V00404430 02A00100 00003A00 9A000422 00020003 16 *.............. * R00D7F43
V00404440 00000000 00008000 00008000 FFFFFFFF 16 *.............. * R00D7F44
V00404450 FFFFFFFF 00800000 ........ ........ 16 *.............. * R00D7F45
      •
      •
      •
V004044B0 E3C5E2E3 4BD2E2C4 E24BC3D3 E4E2E3C5 16 *TEST.KSDS.CLUSTE* R00D7F4B0
V004044C0 D9404040 40404040 40404040 40404040 16 *R               * R00D7F4C0

show bg,400494.40
V00400490 ........ A040004C 00404418 001B3FF8 16 *.... .<. ....8* R00D8F490
V004004A0 00000000 00009A00 28110400 00000000 16 *.............. * R00D8F4A0
V004004B0 D2E2C4E2 40404040 00000000 00000000 16 *KSDS    ........* R00D8F4B0
V004004C0 0040A000 00000000 00000000 00080000 16 *. ............ * R00D8F4C0
V004004D0 00000000                            16 *....            * R00D8F4D0
```

VSAM File Access Conflicts

Following errors will result in OPEN return code x'A8':

1. Attempt to OPEN a SHR(2) file for output while the same file is already open for output in another partition or same partition. Extended message will tell customer if file is open in this partition or another partition. Remember that OPENs over a path (AIX or base) count, unless the customer is using DataSet Namesharing, CICS File Control Definition (**DFHFCT TYPE=FILE**) option **DSNSHR=ALL | UPDATE**.
2. KSDS files can be opened in one of three modes: Keyed, Addressed, or CI-mode. Once a SHR(4) file is opened in one of these modes, all other OPENs (even from a different partition) must be in the same mode. This often happens when sharing files between CICS partitions, particularly with the IESCNTL file. IESCNTL must be specified as SHR(4) if it is to be shared between CICS partitions.
3. If the file is shared between multiple systems, ensure that the CPUID has not changed. The external locks include the CPUID in the key, and if the CPUID is changed, old locks under the old CPUID are not cleared. If VSE systems are run second level under VM, changing the CPUID is often done using the "CP SET CPUID" which will, in turn, drive a different ASI proc. Also, powering off a secondary system without first closing all VSAM files, may also leave ghost lock entries being left in the lock file, which are not freed until the next IPL. Use the AR UNLOCK command to release lock entries for the bogus CPUIDs.
4. If file is shared between multiple releases of VSE, check to ensure that DOSRES or SYSWK1 are not marked as ",SHR" on either system. Even though each system has their own version of DOSRES or SYSWK1, or only link to the appropriate volume, all locks are done using only the volume of the catalog and the CI number within the catalog. Thus, VSAM may pick up a lock from one release / catalog, d find that it matches the CI from the other release / catalog, and reject the OPEN.
5. When a file is being initially loaded (prior to first CLOSE), the SHAREOPTIONS for the file are temporarily reset to '1'. This means, even if the SHAREOPTIONS are '2' or '4', another partition will get rc_x'A8' attempting to open the file, even for input.
6. Close of a file under CICS may not always be successful. This is the case whether the close is issued via CEMT command, or from one of a number of available vendor products which issue a request from batch to CICS to close an on-line file so it can be opened for output in batch. CICS will not close the file if there is an outstanding request against the file. This includes an update request that has not been either returned or released to File Control.
7. Consider the following case:
   - File defined with SHR(2)
   - The file is defined multiple times to CICS (or base and path are both defined for the same file)
   - The file definitions are made with "Dataset Name Sharing"
   - At least one file definition is made with SERVREQ of OUTPUT

   CICS will adjust all file SERVREQs to output. Then if file originally defined as "output" is closed, the remaining FCT entries still maintain the output lock for the file, even if they were originally defined as "input". This will prevent opening the file in another partition.

Loop during CICS Initialization

Symptom:

CICS crashed. When we re-ipled then attempted to bring it back up, it hangs during initialization.

Resolution:

1. Check status of CICS partition using **STATUS** command. If CICS/VSE, check status of partition main task. If CICS TS, check status of Quasi-Reentrant (second DFHEVID1) sub-task. If task shows "Ready to Run", it may be in a loop.

2. Display task savearea using **SHOW** command. Display address in current PSW. Identify failing module.
   - Check for a base register
   - Scan backwards looking for an eyecatcher.

3. If active module (phase) eyecatcher begins with "IKQ", this is VSAM, and we are probably suffering a corrupted file index. A hard outage often prevents VSAM from purging active buffers with the result that partial changes are left on disk.

4. Determine what file is currently being accessed. Check contents of Reg10 – Reg13. At least one of them will be pointing at one of the following VSAM control blocks:

   - **AMBL** begins with "110000C4". ACB pointer is at displacement x'14', and the 44-character fileid is visible at x'78'.
   - **ACB** begins with "A040004C". AMBL pointer is at displacement x'08', and the 8 character filename used to open the file is at x'1C'.
   - **PLH** (Place Holder) has "55FF" at displacement x'114', followed by a pointer to the ACB at x'134'. Two pointers to the current RPL at are x'C8' and x'CC'.
   - **RPL** begins with "00100034". ACB pointer is at displacement x'18'.

5. If you are successful in identifying a failing file, take it off-line, and attempt to bring up CICS without this file. Print the entire file, or copy it to tape using **IDCAMS REPRO**. Often this will also show a loop. At this point you will have to re-build the file:

   - **REPRO** the data component of the file to tape
   - Sort the output tape (eliminate duplicates)
   - Delete and re-define the file
   - Re-loading the file from the sorted tape.

Loop during CICS Initialization

1. Check status of CICS partition

CICS/VSE:

```
status f2
AR 0015 S3E-F2 DFHLOADR 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004088C PCB=00041280 TCB=0025907C TIB=00259000 SAV=008F90E0
AR 0015 S3F-F2 DFHSKP   82 WAITING ON TIMER INTERRUPT
AR 0015    SCB=0004088C PCB=00041280 TCB=0025937C TIB=00259300 SAV=004D1D34
AR 0015 S40-F2 DTSCOPCM 82 WAITING ON TIMER INTERRUPT
AR 0015    SCB=0004088C PCB=00041280 TCB=0025967C TIB=00259600 SAV=00903028
AR 0015 S41-F2 DTSNTFY  82 WAITING ON TIMER INTERRUPT
AR 0015    SCB=0004088C PCB=00041280 TCB=0025997C TIB=00259900 SAV=0091DA90
AR 0015 S42-F2 DTSCHIGH 82 WAITING ON TIMER INTERRUPT
AR 0015    SCB=0004088C PCB=00041280 TCB=00259C7C TIB=00259C00 SAV=0094F8A0
AR 0015 M23 F2 CICSICCF 83 READY TO RUN
AR 0015    SCB=0004088C PCB=00041280 TCB=0004AA80 TIB=00043680 SAV=00400000
AR 0015 1I40I  READY
```

CICS TS:

```
status f2
AR 0015 S40-F2 EVA10MST 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002C461C TIB=002C45A0 SAV=00405260
AR 0015 S41-F2 DFHEVID2 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002C48EC TIB=002C4870 SAV=00292400
AR 0015 S42-F2 DFHEVID1 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002C4BBC TIB=002C4B40 SAV=00292480
AR 0015 S43-F2 DFHEVID1 83 READY TO RUN
AR 0015    SCB=0004980C PCB=0004A200 TCB=002E607C TIB=002E6000 SAV=00292500
AR 0015 S44-F2 DFHSKTSK 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002E634C TIB=002E62D0 SAV=00292580
AR 0015 S45-F2 DTSCOPCM 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002E661C TIB=002E65A0 SAV=0046E028
AR 0015 S46-F2 DTSNTFY  82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002E68EC TIB=002E6870 SAV=0098ABD0
AR 0015 S47-F2 DTSCHIGH 82 WAITING ON TIMER INTERRUPT
AR 0015    SCB=0004980C PCB=0004A200 TCB=002E6BBC TIB=002E6B40 SAV=0046F8A0
AR 0015 M23 F2 CICSICCF 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=00054E40 TIB=0004C600 SAV=00400000
AR 0015 1I40I  READY
```

2. Display task savearea

```
show f2,400000.60
AR 0015   DATA  FOUND AT 00400000
V00400000 C4C6C8E2 C9D74040 07BD2000 801ADE18 B6 *DFHSIP  ........* R01310000
V00400010 009128E0[9] 0090C990[A] 00912910[B] 001AD578[C] B6 *.j.\..I..j....N.* R01310010
V00400020 012E4798[D] 801AEC26[E] 001AF7E9[F] 801AF30E[0] B6 *...q......7Z..3.* R01310020
V00400030 009128C0[1] 00000000[2] 012E970C[3] 00912840[4] B6 *.j.{......p..j. * R01310030
V00400040 004CE8BC[5] 801ADE1C[6] 0090CA98[7] 012F1398[8] B6 *.<Y........q...q* R01310040
V00400050 0000B3D9 A4C8980D 40404040 40404040 B6 *...RuHq.        * R01310050
```

PSW Address

Loop during CICS Initialization

2. … Display address in current PSW

PSW points here

```
show 1ADE10.40
V001ADE10 12114770 C82A07F6 5810D050 58101070 06 *....H..6..}&....* R00624E10
V001ADE20 911010A6 07F6185B 58B0D050 4510C8E0 06 *j..w.W.$..}&..H\* R00624E20
V001ADE30 5B5BC2C3 E5E2F0F3 C9D2D8E5 C4E4D4D7 06 *$$BCVS03IKQVDUMP* R00624E30
V001ADE40 FFFFFF67 00000000 001AD57C FFFFFFFF 06 *.........N@....* R00624E40
AR 0015 1I40I  READY
```

Reg12 is used as base reg:
4770**C**82A

2. … Identify failing module.

```
show f2,001AD578.60
V001AD570 ........ ........ 47F0C018 C9D2D8C9 06 *.........0{.IKQI* R00624570
V001AD580 D6C44040 F3F5C340 C4E8F4F4 F7F7F800 06 *OD  35C DY44778.* R00624580
V001AD590 4560C8A0 5860D108 58106030 12114780 06 *.-H..-J...-.....* R00624590
V001AD5A0 C0D49110 10024710 C0CA1B55 BF571009 06 *{Mj.....{.......* R006245A0
V001AD5B0 951B5000 4770C044 92075000 58501020 06 *n.&...{.k.&..&..* R006245B0
V001AD5C0 4A55003A 4B50C900                   06 *¢....&I.        * R006245C0
```

4. Determine what file is currently being accessed

Register 11 often points to the **AMBL**:

Eyecatcher

Pointer to ACB
AMBL+x'14'

```
show f2,0090C818.100
V0090C810 ........ ........ 110000C4 0090C890 B6 *...........D..I.* R01413810
V0090C820 0090CB30 012E4798 0090C818 008F1BF4 B6 *.......q..H....4* R01413820
V0090C830 02A80380 00005600 8A0005D4 00040009 B6 *.y.........M....* R01413830
V0090C840 00000000 00008000 00008000 FFFFFFFF B6 *................* R01413840
V0090C850 FFFFFFFF 00800000 00000000 012F1798 B6 *.............q..* R01413850
V0090C860 00000000 0090E800 00000000 00000000 B6 *.....          * R01413860
V0090C870 00000000 00000000 00000000 00000000 B6 *.....          * R01413870
V0090C880 00008000 00000000 00003F00 00000000 B6 *.....          * R01413880
V0090C890 01000000 00000000 00000000 00000000 B6 *................* R01413890
V0090C8A0 001BFBC0 00000000 00000000 00000000 B6 *................* R014138A0
V0090C8B0 C3C9C3E2 4BD9E2C4 40404040 40404040 B6 *CICS.RSD        * R014138B0
V0090C8C0 40404040 40404040 40404040 40404040 B6 *                * R014138C0
V0090C8D0 40404040 40404040 40404040 00000000 B6 *            ....* R014138D0
```

Fileid
AMBL+x'98'

Register 12 often points to the **RPL**:

Eyecatcher
CICS TS: "0011003C"

```
show f2,004CE8BC.40
V004CE8B0 ........ ........ ........ 00100034 B6 *................* R014368B0
V004CE8C0 00000125 004CE881 004E63F4 000000A6 B6 *.....<Ya.+.4...w* R014368C0
V004CE8D0 00000004 008F1BF4 011C0006 0800FFFF B6 *.......4........* R014368D0
V004CE8E0 00000000 00000000 00000000          B6 *............    * R014368E0
```

Pointer to ACB
RPL+x'18'

Loop during CICS Initialization

Register 13 often points to the **PLH** (Placeholder):

```
show f2,012E4798.140
V012E4790 ........ ........ 00000000 005113C8 B6 *...............H* R01416790
V012E47A0 00000000 00000066 012E4904 012E96F0 B6 *..............o0* R014167A0
V012E47B0 00901300 00000000 008F1BF4 00901E80 B6 *...........4....* R014167B0
V012E47C0 00000000 001BA990 801BABCC 0090C990 B6 *......z.......I.* R014167C0
V012E47D0 0090C818 004CE8BC 012E4798 801BACFC B6 *..H..<Y....q....* R014167D0
V012E47E0 001B6580 0090C990 0090C818 004CE8BC B6 *......I...H..<Y.* R014167E0
V012E47F0 012E4798 801B6F2E 001AE9C8 000001F4 B6 *...q..?...ZH...4* R014167F0
V012E4800 012E96F0 801AC2E4 801AD298 801ACF00 B6 *..o0..BU..Kq....* R01416800
V012E4810 00000000                            B6 *...........f....* R01416810
V012E4820 00000001                            B6 *.........<Ya..u.* R01416820
V012E4830 00000001                            B6 *..........i.....* R01416830
V012E4840 001AA358                            B6 *..t...e.........* R01416840
V012E4850 0000000          801A89BE                   ...i.....* R01416850
V012E4860 004CE8BC 004CE8BC 23FF80FF                  .........* R01416860
V012E4870 FFFFFFFF 801AEB82 00000000                  b........* R01416870
V012E4880 00000000 00000000 00000000                  .........* R01416880
V012E4890 00000000 00000000 00000001 00460015 B6 *................* R01416890
V012E48A0 012F1398 012E96F0 012F138C 55FF0023 B6 *...q..o0........* R014168A0
V012E48B0 000CA900 012E4A28 00000000 00000000 B6 *...                * R014168B0
V012E48C0 00000000 00000000 00000000 008F1BF4 B6 *..                .* R014168C0
V012E48D0 00000000 00000000 01FF0023 00901300 B6 *..                 * R014168D0
V012E48E0 00000000 001AE9C8 00000000 00000000 B6 *......ZH........* R014168E0
V012E48F0 012F1398 012F1298 004CE8BC 00000000 B6 *...q...q.<Y.....* R014168F0
```

Pointers to Current RPL
PLH+x'C8' and x'CC'

Eyecatcher
PLH+x'114'

Pointer to ACB
PLH+x'134'

ACB:

Eyecatcher

Pointer to AMBL
ACB+x'04'

Filename
ACB+x'1C'

```
show f2,008F1BF4.60
V008F1BF0 ........ A0400040 0090C818 001A8450 B6 *.....    ..H...d&* R013F6BF0
V008F1C00 00000000 00008A00 28110300 00000000 B6 *................* R013F6C00
V008F1C10 C4C6C8D9 E2C44040 00000000 00000000 B6 *DFHRSD  ........* R013F6C10
V008F1C20 012D3A00 004A5FB4 00000000 00080000 B6 *.....¢¬.........* R013F6C20
V008F1C30 00000000 00000000 01000000 00000000 B6 *................* R013F6C30
V008F1C40 00100034                            B6 *....            * R013F6C40
```

Correlate VSE/ESA Dump with COBOL Source

Symptom: CICS Partition hang. No response to MSG F2.

```
status f2
AR 0015 S3E-F2 EVA10MST 83 READY TO RUN
AR 0015    SCB=0004908C PCB=00049A80 TCB=003B90
AR 0015 S3F-F2 DFHEVID2 83 READY TO RUN
AR 0015    SCB=0004908C PCB=00049A80 TCB=003B91
AR 0015 S40-F2 DFHEVID1 83 READY TO RUN
AR 0015    SCB=0004908C PCB=00049A80 TCB=003B961C TIB=003B95A0 SAV=00391380
AR 0015 S41-F2 DFHEVID1 83 READY TO RUN
AR 0015    SCB=0004908C PCB=00049A80 TCB=003B98EC TIB=003B9870 SAV=00391400
AR 0015 S42-F2 DFHSKTSK 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004908C PCB=00049A80 TCB=003B9BBC TIB=003B9B40 SAV=00391480
AR 0015 M23 F2 DBDCCICS 83 READY TO RUN
AR 0015    SCB=0004908C PCB=00049A80 TCB=000546C0 TIB=0004BE80 SAV=00600000
```

The partition appears to be looping. Repeatedly issuing the STATUS command, and checking the second DFHEVID1 save area shows PSWs in a limited range of storage.

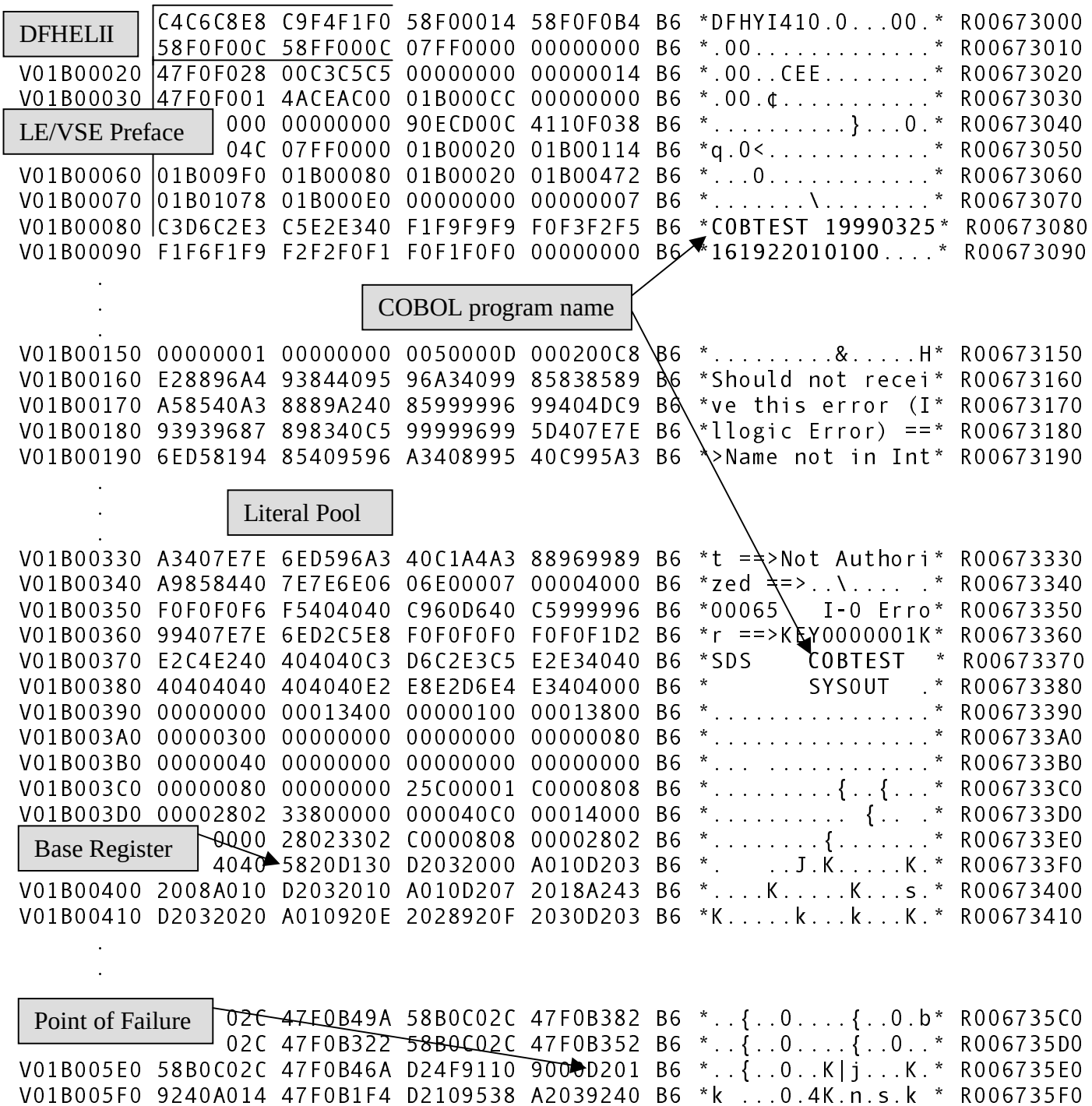
```
show 391400.60
AR 0015   DATA  FOUND AT 00391400
V00391400 C4C6C8C5 E5C9C4F1 07BD0000 81B005EE 06 *DFHEVID1....a...* R00256400
V00391410 0165B710[9] 01B00144[A] 01B003F0[B] 01B00114[C] 06 *............0....* R00256410
V00391420 0165B4C8[D] 81B00544[E] 00000000[F] 0165B60C[0] 06 *...Ha............* R00256420
V00391430 0165B678[1] 00000000[2] 0165B820[3] 01B00058[4] 06 *................* R00256430
V00391440 0165B190[5] 00710B9C[6] 00000000[7] 007C00D0[8] 06 *.............@.}* R00256440
V00391450 0000B1FF 004190CF 40404040 40404040 06 *........          * R00256450
AR 0015 1I40I  READY

V01B003C0 00000080 00000000 25C00001 C0000808 B6 *................*
V01B003D0 00002802 33800000 000040C0 00014000 B6 *.........  ...  .*
V01B003E0 07080000 38023302 C0000808 00002802 B6 *................*
V01B003F0 33404040 5820D130 D2032000 A010D203 B6.*.   ..J.K.....K.*
V01B00400 2008A010 D2032010 A010D207 2018A243 B6 *....K.....K.....*
      .
      .
      .
V01B005C0 58B0C02C 47F0B49A 58B0C02C 47F0B382 B6 *.....0.......0..*
V01B005D0 58B0C02C 47F0B322 58B0C02C 47F0B352 B6 *.....0.......0..*
V01B005E0 58B0C02C 47F0B46A D24F9110 9000D201 B6 *.....0..K.....K.*
V01B005F0 9240A014 47F0B1F4 D2109538 A2039240 B6 *.. ...0.4K......*
```

Register 11 is base reg

We don't appear to have an eyecatcher around the base register, so we fall back to scrolling backwards.

Correlate VSE/ESA Dump with COBOL Source

As we scroll back, we notice a distinctive pattern. Preceding the beginning of code (pointed to by the base register), we encounter an area with mixed hex and declarative literals / messages. Toward the end (high end) of this literal pool, we see the name of the COBOL program.
As we scroll back further, we again see the name of the COBOL program with a compile date/time. x'80' bytes prior to this we see 'DFHYI410, which is the eyecatcher for DFHELII.




```
          C4C6C8E8 C9F4F1F0 58F00014 58F0F0B4 B6 *DFHYI410.0...00.* R00673000
          58F0F00C 58FF000C 07FF0000 00000000 B6 *.00.............* R00673010
V01B00020 47F0F028 00C3C5C5 00000000 00000014 B6 *.00..CEE........* R00673020
V01B00030 47F0F001 4ACEAC00 01B000CC 00000000 B6 *.00.¢...........* R00673030
          000 00000000 90ECD00C 4110F038 B6 *..........}...0.* R00673040
          04C 07FF0000 01B00020 01B00114 B6 *q.0<............* R00673050
V01B00060 01B009F0 01B00080 01B00020 01B00472 B6 *...0...........* R00673060
V01B00070 01B01078 01B000E0 00000000 00000007 B6 *.......\........* R00673070
V01B00080 C3D6C2E3 C5E2E340 F1F9F9F9 F0F3F2F5 B6 *COBTEST 19990325* R00673080
V01B00090 F1F6F1F9 F2F2F0F1 F0F1F0F0 00000000 B6 *161922010100....* R00673090
      .
      .
      .
V01B00150 00000001 00000000 0050000D 000200C8 B6 *.........&.....H* R00673150
V01B00160 E28896A4 93844095 96A34099 85838589 B6 *Should not recei* R00673160
V01B00170 A58540A3 8889A240 85999996 99404DC9 B6 *ve this error (I* R00673170
V01B00180 93939687 898340C5 99999699 5D407E7E B6 *llogic Error) ==* R00673180
V01B00190 6ED58194 85409596 A3408995 40C995A3 B6 *>Name not in Int* R00673190
      .
      .
      .
V01B00330 A3407E7E 6ED596A3 40C1A4A3 88969989 B6 *t ==>Not Authori* R00673330
V01B00340 A9858440 7E7E6E06 06E00007 00004000 B6 *zed ==>..\.... .* R00673340
V01B00350 F0F0F0F6 F5404040 C960D640 C5999996 B6 *00065   I-O Erro* R00673350
V01B00360 99407E7E 6ED2C5E8 F0F0F0F0 F0F0F1D2 B6 *r ==>KEY0000001K* R00673360
V01B00370 E2C4E240 404040C3 D6C2E3C5 E2E34040 B6 *SDS    COBTEST  * R00673370
V01B00380 40404040 404040E2 E8E2D6E4 E3404000 B6 *       SYSOUT  .* R00673380
V01B00390 00000000 00013400 00000100 00013800 B6 *................* R00673390
V01B003A0 00000300 00000000 00000000 00000080 B6 *................* R006733A0
V01B003B0 00000040 00000000 00000000 00000000 B6 *... ............* R006733B0
V01B003C0 00000080 00000000 25C00001 C0000808 B6 *.........{..{...* R006733C0
V01B003D0 00002802 33800000 000040C0 00014000 B6 *.......... {.. .* R006733D0
          0000 28023302 C0000808 00002802 B6 *........{.......* R006733E0
          4040 5820D130 D2032000 A010D203 B6 *.   ..J.K.....K.* R006733F0
V01B00400 2008A010 D2032010 A010D207 2018A243 B6 *....K.....K...s.* R00673400
V01B00410 D2032020 A010920E 2028920F 2030D203 B6 *K.....k...k...K.* R00673410
      .
      .
          02C 47F0B49A 58B0C02C 47F0B382 B6 *..{..0....{..0.b* R006735C0
          02C 47F0B322 58B0C02C 47F0B352 B6 *..{..0....{..0..* R006735D0
V01B005E0 58B0C02C 47F0B46A D24F9110 9000D201 B6 *..{..0..K|j...K.* R006735E0
V01B005F0 9240A014 47F0B1F4 D2109538 A2039240 B6 *k ...0.4K.n.s.k * R006735F0
```

Correlate VSE/ESA Dump with COBOL Source

```
// EXEC LNKEDT,SIZE=256K
JOB  TEST2CMP  03/25/1999 5686-066-06-35C-0   LINKA
ACTION TAKEN  MAP
-------------------------------------------------------
  PHASE CICSTST2,*
-------------------------------------------------------
  INCLUDE DFHELII
     ** MODULE DFHELII  1999-02-04 11.27    INCLUDED FROM  PRD1.BASE
     ** MODULE CEEBETBL 1999-02-04 11.32    AUTOLNKD FROM  PRD2.SCEEBASE
     ** MODULE CEEBINT  1999-02-04 11.32    AUTOLNKD FROM  PRD2.SCEEBASE
      .
      .
      .
ENTRY
03/25/1999 PHASE  XFR-AD  LOCORE  HICORE    CSECT/    LOADED   TAKEN   RMODE/
                                            ENTRY     AT       FROM
          ------------------------------------------------------------------
          CICSTST2 600098  600078  601BC9                                 31
          ------------------------------------------------------------------
                                            DFHELII   600078   DFHELII   31
                                           *DFHEPIN   600078
                                           *DFHEXEC   600080
      .
.     .
      .
                                            COBTEST   600098   SYSLNK   ANY
                                            CEEBETBL  600C88   CEEBETBL ANY
                                            CEEBINT   600CA8   CEEBINT  ANY
      .
      .
      .
UNRESOLVED EXTERNAL REFERENCES                      WXTRN     CEEUOPT
                                                    WXTRN     CEEBXITA
                                                    WXTRN     CEESG000
                                                    WXTRN     CEESG001
                                                    WXTRN     CEESG002
      .
      .
      .
UNRESOLVED ADCON  AT OFFSET 00600C98
UNRESOLVED ADCON  AT OFFSET 00600C8C
UNRESOLVED ADCON  AT OFFSET 00600CC0
UNRESOLVED ADCON  AT OFFSET 00600CC4
      .
      .
      .
024 UNRESOLVED ADDRESS CONSTANTS
PHASE(S) CATALOGED INTO  SUBLIBRARY   PRIVB1 .OLSEN     VOLID=  PACC89
1S55I  LAST RETURN CODE WAS 0002
EOJ TEST2CMP  MAX.RETURN CODE=0002                          DATE 03/25/1999,
```

PSW points to: 1B005EE,
which is x'5EE' into the phase.
Since a phase is comprised of modules, use the LNKEDT listing to identify the failing module.

Phase + x'5EE' is at
COBTEST + x'5CE'

RC=02 is normal for a
COBOL/VSE link-edit.

Correlate VSE/ESA Dump with COBOL Source

Compile listing with options LIST, SOURCE.

```
// JOB TEST2CMP COMPILE PROGRAM                                  DATE 03/25/1999,
// LIBDEF *,SEARCH=(PRIVB1.OLSEN,PRD2.SCEECICS,PRD2.SCEEBASE,PRD2.PROD)
// LIBDEF PHASE,CATALOG=PRIVB1.OLSEN
// OPTION ERRS,SXREF,SYM,CATAL,NODECK
   PHASE CICSTST2,*
   INCLUDE DFHELII
// EXEC IGYCRCTL,SIZE=IGYCRCTL
PP 5686-068 IBM COBOL FOR VSE/ESA 1.1.0          09/13/1994                DATE
JCL OPTION PARAMETERS:
NODECK,LINK,LIST,NOLISTX,SYM,NOTERM,SXREF
PROCESS(CBL) STATEMENTS:
 CBL LIB,APOST,NOADV,NODYNAM,RENT,BUF(4096),SOURCE,LIST
       CBL RENT,NODYNAM,LIB
LINEID  PL SL  ----+-*A-1-B--+----2----+----3----+----4----+----5----+----6--
  000001               IDENTIFICATION DIVISION.
  000002               PROGRAM-ID.                COBTEST.
  000003               AUTHOR.                    CHARLES E. OLSEN
  000007               ENVIRONMENT DIVISION.
  000008               INPUT-OUTPUT SECTION.
  000009               FILE-CONTROL.
  000015               DATA DIVISION.
  000021               WORKING-STORAGE SECTION.
     .
     .
     .
  000031               01  Starting-Message.
  000032                   05                          PIC X(31)
  000033                       VALUE 'This program contains a loop.'.
  000034                   05                          PIC X(30)
  000035                       VALUE 'Press enter to begin loop. '.
  000036               77  MSGAREA                     PIC X(80).
  000037               77  Console-Inarea              PIC X(200).
  000038               77  VSAM-Key                    PIC X(10) VALUE ...
  000097               LINKAGE SECTION.
  000098               01  DFHEIBLK.
     .
     .
     .
  000130               01  DFHCOMMAREA PICTURE X(1).
  000131               PROCEDURE DIVISION USING DFHEIBLK DFHCOMMAREA.
  000132                   MOVE 2 TO Console-Record-Length.
     .
     .
     .
  000153               LOOP.
  000154                   MOVE VSAM-Base-Record TO MSGAREA.
  000155                   MOVE 80 TO Console-Record-Length.
  000156                   GO TO LOOP.
     .
     .
     .
  000258               EOJ.
  000259                   GOBACK.
     .
     .
     .
```

Identify the COBOL statement which is at displacement x'5CE'

Program loop at statement 153-156

Correlate VSE/ESA Dump with COBOL Source

```
PP 5686-068 IBM COBOL FOR VSE/ESA 1.1.0        09/13/1994      COBTEST   DATE
    000000            COBTEST  DS   0H
                               USING *,15
    000000  47F0 F028          B    40(,15)       BYPASS CONSTANTS. BRANCH TO
    000004  00                 DC   AL1(0)        ZERO NAME LENGTH FOR DUMPS
    000005  C3C5C5             DC   CL3'CEE'      CEE EYE CATCHER
    000008  00000000           DC   F'0'          STACK FRAME SIZE
    00000C  00000014           DC   A(@PPA1-COBTEST) OFFSET TO PPA1 FROM ENTR
    000010  47F0 F001          B
      .                                  LE/VSE preface = x'F4' bytes
      .                                  Setup and call CEESTART
      .
    000060  C3D6C2E3C5E2E340   DC   C'COBTEST '   PROGRAM NAME
    000068  F1F9F9F9           DC   CL4'1999'   @TIMEVRS: YEAR OF COMPILATION
    00006C  F0F3F2F5           DC   CL4'0325'             MONTH/DAY OF COMPIL
    000070  F1F6F1F9           DC   CL4'1619'             HOURS/MINUTES OF CO
    000074  F2F2               DC   CL2'22'               SECONDS FOR COMPILA
    000076  F0F1F0F1F0F0       DC   CL6'010100'   VERSION/RELEASE/MOD LEVEL 0
      .
      .
      .
PROGRAM GLOBAL TABLE BEGINS AT LOCATION 0000F4 FOR 000030 BYTES

LITERAL POOL MAP FOR SYSTEM LITERALS IN THE PGT:   Literal Pool
0000F4 (LIT+0)  00000000 40404040 40404040 40404040 40404040 40404040 ...
000114 (LIT+32) 40404000 00000C
000130 (LIT+0)  00000001 00000000 0050000D 000200C8 E28896A4 93844095 ...
                85838589  ³.........&.....HShould not recei³
000150 (LIT+32) A58540A3 8889A240 85999996 99404DC9 93939687 898340C5 ...
                5D407E7E  ³ve this error (Illogic Error) ==³
000170 (LIT+64) 6ED58194 85409596 A3408995 40C995A3 8599A2A8 A2A38594 ...
                9385407E  ³>Name not in Intersystem Table =³
      .
      .                  Start of Code (where base reg was pointing)
      .
    0003D4  5820 D130           L    2,304(0,13)            BL=1
    0003D8  D203 2000 A010      MVC  0(4,2),16(10)          SORT-CORE-SIZE
    0003DE  D203 2008 A010      MVC  8(4,2),16(10)          SORT-FILE-SIZE
    0003E4  D203 2010 A010      MVC  16(4,2),16(10)         SORT-MODE-SIZE
      .
      .
      .
    0005C0  58B0 C02C           L                          L=1
    0005C4  47F0 B46A           BC                         -RECORD-FOUND
    0005C8              GN=23   EQU
000153  *LOOP                                 Loop: Statements 153 – 156
000154  MOVE                                  Program Displacement +x'5CE'
    0005C8  D24F 9110 9000      MV
000155  MOVE
    0005CE  D201 9240 A014      MV
000156  GO
    0005D4  47F0 B1F4           BC
```

To summarize:

1. Loop occurred at address x'01B005EE'.
2. By locating the eyecatcher, we determined this was +x'5EE' into the phase.
3. From the link-edit listing, this is COBTEST + x'5CE'
4. COBOL listing shows this to be statement 155.

Stand-alone Dump to Disk

Symptom:

A Stand-alone dump is taken when VSE/ESA has suffered a catastrophic error, and IBM support may require a snapshot of the entire VSE/ESA system. Most Stand-alone dumps are taken to tape. This requires, however, that the operator have a tape available when the error occurs, and knows how to use it. In addition, system outages always occur at the worst time, usually when the system is fully loaded and under stress. Operations may be reluctant to take the time for a Stand-alone dump prior to bringing the system back up.

Resolution:

A Stand-alone dump may be taken to either tape or dasd. The advantages of using dasd as the medium for Stand-alone dump are:

1. **Quicker**. In a "system down" situation, a stand-alone dump to disk takes approximately 1/3 as much time as a stand-alone dump to tape.
2. **More convenient**. Operator does not have to locate the stand-alone dump tape in a crisis situation.
3. **Remote access**. Processing the dump after the system has been re-ipl'ed does not require additional tape mounts.

Steps to follow:
1. Define Stand-alone dump on dasd
2. Add labels to standard labels, and assigns to CICS start-up job
3. Take a trial stand-alone dump
4. Check, using DITTO, to see how much space you really used
5. Scan dump file using DOSVSDMP to ensure that all critical partitions are being dumped
6. Test procedure to onload dump into VSEDUMP (InfoAnal) library
7. Test procedure to offload dump to tape for transmission to VSE Level2.
8. Ensure Operations understands the procedures when and how to take a stand-alone dump.
9. During the first few months with a new operating system release, and after all major system changes, it is advisable to run with **DEBUG ON**. This captures critical system diagnostic data in a non-obtrusive manner which will substantially facilitate problem determination if a stand-alone dump is necessary.

Stand-alone dumps on dasd can also be processed using the Interactive Interface (4.3:"Storage Dump Management"). However, this requires that the stand-alone dump extent on disk must be created using the Interactive Interface (4.6.2. "Create Stand-alone Dump Program on Disk")

1. Define Stand-alone dump on dasd: (See "VSE/ESA Diagnosis Tools", Chapter 3)

CICS TS	(30 Meg)
POWER	(1.6 Meg)
VTAM	(7 Meg)
Supervisor and SVA	(12 Meg)
Total:	(50.6 Meg)

Three partition s on our test system were brought up with "// OPTION SADUMP=5", indicating they should be included in the stand-alone dump file. Partition sizes indicate a stand-alone dump size of over 50meg. However, dump records are not written for unused pages.

Stand-alone Dump to Disk

Create Stand-alone dump files on disk:

```
R RDR,PAUSEBG
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R88I  OK
BG 0001 1Q47I   BG PAUSEBG 00148 FROM LOCAL , TIME=10:53:36
BG 0000 // JOB PAUSEBG
        DATE 03/18/2000, CLOCK 10/53/36
BG-0000 // PAUSE

0 // EXEC DOSVSDMP
BG 0000 4G01D SELECT ONE OF THE FOLLOWING FUNCTIONS:

  1   CREATE STAND-ALONE DUMP PROGRAM        ←
  2   SCAN DUMP TAPE/DISK
  3   PRINT DUMP TAPE/DISK
  4   PRINT SDAID TAPE
  5   PRINT IPL DIAGNOSTICS
  R   END DOSVSDMP PROCESSING

BG-0000
```

Or, DOSVSDMP can be invoked via JCL:

```
// JOB DOSVSDMP
// DLBL IJSYSDI,'VSEDUMP.PROGRAM.ON.DISK',99/365,SD
// EXTENT SYS013,SYSWK2,1,0,1,7
// DLBL IJSYSDU,'STANDALONE.DUMP.ON.DISK',99/365,SD
// EXTENT SYS013,SYSWK2,1,0,8,7000
// ASSGN SYS013,DISK,VOL=SYSWK2,SHR
// EXEC DOSVSDMP,PARM='CREATE DUMP DEVICE=SYS013'
/*
/&
```

```
BG 0000 // JOB DOSVSDMP
        DATE 03/13/2000, CLOCK 22/57/46
BG 0000 4G09I DUMP PROGRAM HAS BEEN CREATED
BG 0000 4G27I DUMP FILE CAPACITY IS   72000 K BYTES
BG 0000 EOJ DOSVSDMP  MAX.RETURN CODE=0008
        DATE 03/13/2000, CLOCK 23/02/31, DURATION   00/04/45
```

2. Add labels to standard labels, and assigns to CICS start-up job

```
CATALOG DTRINFOA.PROC     EOD=YY DATA=YES REPLACE=YES
// ASSGN SYS016,DISK,VOL=SYSWK1,SHR       INFO ANAL MANAGEMENT FILE
// ASSGN SYS017,DISK,VOL=SYSWK1,SHR       INFO ANAL ROUTINES FILE
// ASSGN SYS013,DISK,VOL=SYSWK2,SHR       STAND-ALONE DUMP          ←
YY
```

Stand-alone Dump to Disk

3. Take a trial stand-alone dump

```
* CP I 142
16:10:52  * MSG FROM VSETEST3: 4G34I STAND-ALONE DUMP IN PROGRESS ON DISK 142
16:11:29  * MSG FROM VSETEST3: 4G10I STAND-ALONE DUMP COMPLETE
HCPGIR450W CP entered; disabled wait PSW 000A0000 00CE0000
```

Stand-alone dump to disk took 35 seconds, versus 90 seconds to tape. This was a minimal test system. Obviously, a fully-functional production system will take much longer.

**** Be sure to close your on-line files before you try this step. Doing a system reset with on-line files still open can cause corruption in VSAM and DL/I files. ****

4. Check, using DITTO, to see how much space you really used

```
// JOB DITTO - PRINT DISK EXTENTS
*
*  ADDRESS IS CCHHH IN DECIMAL. CYLINDER AND HEAD BOTH START WITH ZERO.
*
// UPSI 1
// ASSGN SYS010,DISK,VOL=SYSWK2,SHR
// EXEC DITTO,SIZE=512K
$$DITTO SET PAGESKIP=NO,DATAHDR=NO,DUMP=ACROSS
$$DITTO DP FORMAT=HEX,INPUT=SYS010,BEGIN=03209,END=03514
/*
/&
```

5. Scan dump file using DOSVSDMP to ensure that all critical partitions are being dumped

All long-running jobs (CICS, POWER, VTAM, SQL, Vendor products) should be included in a stand-alone dump. Add "// **OPTION SADUMP=**nn" to the start-up jobs of any missing applications.

```
// EXEC DOSVSDMP,PARM='SCAN DEVICE=SYS013'

PRINTOUT OF VSE DUMP DATA SET
DIRECTORY OF VSE DUMP DATA SET
DUMP FILE  DUMP TYPE  NAME      DATE      DATA DUMPED
---------  ---------  --------  --------  --------------------------
   001     SADUMP                         SUPERVISOR+SVA
   002     SADUMP               00/03/14  PMRAS-R
   003     SADUMP               00/03/14  PMRAS-00
   004     SADUMP     VTAMSTRT  00/03/14  F3-PARTITION
   005     SADUMP     CICSICCF  00/03/14  F2-PARTITION
   006     SADUMP     POWSTART  00/03/14  F1-PARTITION
END OF DUMP
EOJ DOSVSDMP                          DATE 03/15/2000, CLOCK 11/16/59, DURAT
```

Stand-alone Dump to Disk

6. Test procedure to onload dump into VSEDUMP (InfoAnal) library

Each dump must be loaded into the VSEDUMP library in a separate INFOANAL step.

```
// JOB INFOANAL ONLOAD S/A DUMP FROM DISK
// EXEC PROC=DTRINFOA
// EXEC INFOANA,SIZE=300K
  SELECT DUMP MANAGEMENT          *** DELETE PREVIOUS DUMPS ***
    DUMP NAME SYSDUMP.BG.SAD00001
    DELETE
    DUMP NAME SYSDUMP.F2.SAD00001
    DELETE
    RETURN
  DUMP NAME SYSDUMP.BG.SAD00001   *** Load Supervisor and SVA ***
  SELECT DUMP ONLOAD
     VOLID DISK SYS013
     FILE 1                         ←
     RETURN
  DUMP NAME SYSDUMP.F2.SAD00001   *** Load CICS partition ***
  SELECT DUMP ONLOAD
     VOLID DISK SYS013
     FILE 5 LAST                    ←
     RETURN
  SELECT END
/*
/&
```

7. Test procedure to offload dump to tape for transmission to VSE Level2.

```
// JOB DITTO - COPY STAND-ALONE DUMP TO TAPE
// UPSI 1
// ASSGN SYS013,DISK,VOL=SYSWK2,SHR
// EXEC DITTO,SIZE=512K
$$DITTO REW OUTPUT=181
$$DITTO ST INPUT=SYS013,FILEIN=IJSYSDU,OUTPUT=181,RECFMOUT=F
/*
/&
```

8. Ensure Operations understands the procedures when and how to take a stand-alone dump.

Actual procedures will depend on the processor type and hardware console configuration.

1. Display low core and write down or screen print the first x'100' bytes. Stand-alone dump will destroy several low core locations which may be critical for later problem determination.
2. Perform "**STORE STATUS**".
3. Set the IPL device to the stand-alone dump medium. Mount tape, if necessary
4. Set "**CLEAR**" to "**NO**", and IPL the stand-alone tape / dasd

Stand-alone Dump to Disk

Additional InfoAnal jobs:

Format and print diagnostic information:

```
// JOB INFOANAL ANALYZE S/A DUMP
// EXEC PROC=DTRINFOA
// EXEC INFOANA,SIZE=300K
   SELECT DUMP MANAGEMENT
     DUMP NAME SYSDUMP.BG.SAD00001  *** Run against Supv / SVA dump ***
          RETURN
   SELECT DUMP VIEWING
          CALL IJBXDBUG   *** Analyze Stand-alone dump ***
                          *** Ignore msgBLN9003I 4G80I ***
          CALL IJBXSDA    *** Format In-core SDAID Buffer ***
          CALL IJBXCSMG   *** Print Console Buffer ***
          RETURN
   SELECT DUMP VIEWING
     PRINT FORMAT         *** Print LBD entries ***
   SELECT END
/*
/&
```

IJBXDBUG creates an LBD entry called DBUGHDR. This is printed out by the PRINT FORMAT, along with other dump information and symptom records created during dump creation.

Offload dump to tape for transmission to IBM Support:

```
// JOB OFFLOAD DUMPS FROM VSEDUMP LIBRARY
// ASSGN SYS009,181
// MTC REW,181
// EXEC PROC=DTRINFOA
// EXEC INFOANA,SIZE=300K
  DUMP NAME SYSDUMP.BG.SAD00001  *** Copy Supv / SVA dump to tape ***
  SELECT DUMP OFFLOAD
    VOLID SADUMP SYS009
    ERASE NO
    RETURN
  DUMP NAME SYSDUMP.F2.SAD00001  *** Stack additional dumps to tape ***
  SELECT DUMP OFFLOAD
    VOLID SADUMP SYS009
    ERASE NO
    RETURN
  SELECT END
/*
/&
```

Format of Stand-alone Dump on Disk

DOSVSDMP pre-formats the dasd extent with 4112-byte records beginning with x'00000008' followed by three words of zero:

```
DITTO/ESA for VSE
$$DITTO SET PAGESKIP=NO,DATAHDR=NO,DUMP=ACROSS
$$DITTO DP FORMAT=HEX,INPUT=SYS010,BEGIN=00100,END=00101
* * * * Device 0142, 3390, VOLSER=SYSWK2, Cylinder 0, Head 8, operative primary track                                  * *
CYL-HD-REC 00000-08-001 DATA LEN   4112
 000000    00000008  00000000  00000000  00000000    5C5C5C5C  40E4D5E4  E2C5C440  5C5C5C5C    *                ****  UNUSED *****
 000020    5B5BC2D6  D7C5D540  5B5BC2C3  D3D6E2C5    5BC9D1C2  C1E2C7D5  F1404040  F0F0F0F0    *$$BOPEN $$BCLOSE$IJBASGN1   0000*
 000040    C5D5C440  00000008  00400650  F0F0F0F0    0040019C  0000001C  00000010  00000004    *END        &0000                 *
 000060    00000008  00000018  00400E9A  00400E9C    00400E9D  0001D940  F140F240  F340F440    *                       R 1 2 3 4 *
```

After S/A dump, dasd file looks like this:

- Records beginning with x'00000001' are Symptom / LBD records
- Records beginning with x'00000002' are a 4K storage block. Word 2 is the virtual address; word 3 is the real address. Hex data begins +x'10'.
- Dump for next partitions is preceded by one or more x'00000001' record(s).

```
$$DITTO SET PAGESKIP=NO,DATAHDR=NO,DUMP=ACROSS
$$DITTO DP FORMAT=HEX,INPUT=SYS010,BEGIN=00008,END=00008
* * * * Device 0142, 3390, VOLSER=SYSWK2, Cylinder 0, Head 8, operative primary track                                  * *
CYL-HD-REC 00000-08-001 DATA LEN   4112
 000000    00000001  E2D9F9F6  F7F2F0F4  F9F4F1F0    B3BDFC75  9C1A6644  00000000  00000000    *    SR9672049410               *
 000020    00000000  00000000  00000000  0000F5F6    F8F6F0F6  F6F0F6F4  F5C34080  E2C1C4E4    *              56860660645C  SADU*
 000040    D4D74040  00000000  00000000  00000074    00000074  00A60074  0010011A  00000000    *MP                    w          *
 000060    00000000  00000000  00000000  00000000    00000000  00000000  D4C1C3C8  C9D5C56D    *                         MACHINE_*
 000080    E2E3C1E3  E4E26DD5  D6E36DE2  C1E5C5C4    40C4C1E3  C56DD5D6  E36DC1E5  C1C9D3C1    *STATUS_NOT_SAVED DATE_NOT_AVAILA*
 0000A0    C2D3C540  D4C1C3C8  C9D5C57E  C5E2C140    D4D6C4C5  7ED7C1C7  C9D5C740  C1C3E3C9    *BLE MACHINE=ESA MODE=PAGING ACTI*
 0000C0    E5C56DE2  D7C1C3C5  6DC9C47E  E24040C4    E4D4D7C5  C46DC4C1  E3C16DC6  D9D6D46D    *VE_SPACE_ID=S  DUMPED_DATA_FROM_*
 0000E0    E2D7C1C3  C56DC9C4  7EE24040  D7D4D96D    C1C4C4D9  C5E2E26D  E2D7C1C3  C56DC9C4    *SPACE_ID=S  PMR_ADDRESS_SPACE_ID*
 000100    7EF0F040  C4E4D4D7  C5C46DC4  C1E3C17E    E2E4D7C5  D9E5C9E2  D6D94EE2  E5C10000    *=00 DUMPED_DATA=SUPERVISOR+SVA  *
 000120    00000000  00000000  00000000  00000000    00000000  00000000  00000000  00000000    *                                *
 000140 to  00100F  same as previous line

CYL-HD-REC 00000-08-002 DATA LEN   4112
 000000    00000002  00000000  00000000  061AF1F0    00080000  00000910  06000300  70FF0090    *              10                *

CYL-HD-REC 00000-08-003 DATA LEN   4112
 000000    00000002  00001000  00001000  061A0000    00000000  00000000  00000000  00000000    *                                *

CYL-HD-REC 00000-08-004 DATA LEN   4112
 000000    00000002  00002000  00002000  061A0000    FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF    *                                *

CYL-HD-REC 00000-08-005 DATA LEN   4112
 000000    00000002  00003000  00003000  061A0000    FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF    *                                *
  .
  .
  .
```

Format of Stand-alone Dump on Disk

```
CYL-HD-REC 00032-12-010 DATA LEN   4112
 000000    00000002  021FF000  017DE000  061A0000    021FEFEC  000046A0  FFFF0100  021FF01C    *      0  '                    0 *
CYL-HD-REC 00032-12-011 DATA LEN   4112
 000000    00000001  E2C3E3F6  00000000  00000000    00000000  D3C2C440  002C0000  C6F24040    *    SCT6            LBD     F2  *
CYL-HD-REC 00032-12-012 DATA LEN   4112
 000000    00000004  00000000  00000000  00000000    021FEFEC  000046A0  FFFF0100  021FF01C    *                              0 *
 CYL-HD-REC 00032-13-001 DATA LEN   4112
 000000    00000001  E2D9F9F6  F7F2F0F4  F9F4F1F0    B3BDFC75  9C1A6644  000000F0  F061F0F3    *    SR9672049410           00/03*
 000020    61F1F400  00000000  00000000  0000F5F6    F8F6F0F6  F6F0F6F4  F5C34080  E2C1C4E4    */14           56860660645C  SADU*
 000040    D4D74040  00000000  00000000  00000074    00000074  00B10074  00100125  00000000    *MP                           *
 000060    00000000  00000000  00000000  00000000    00000000  00000000  D4C1C3C8  C9D5C56D    *                        MACHINE_*
 000080    E2E3C1E3  E4E26DD5  D6E36DE2  C1E5C5C4    40C4C1E3  C57EF0F0  61F0F361  F1F440D4    *STATUS_NOT_SAVED DATE=00/03/14 M*
 0000A0    C1C3C8C9  D5C57EC5  E2C140D4  D6C4C57E    D7C1C7C9  D5C740C1  C3E3C9E5  C56DE2D7    *ACHINE=ESA MODE=PAGING ACTIVE_SP*
 0000C0    C1C3C56D  C9C47EE2  4040C4E4  D4D7C5C4    6DC4C1E3  C16DC6D9  D6D46DE2  D7C1C3C5    *ACE_ID=S  DUMPED_DATA_FROM_SPACE*
 0000E0    6DC9C47E  F14040D7  D4D96DC1  C4C4D9C5    E2E26DE2  D7C1C3C5  6DC9C47E  F0F040C4    *_ID=1  PMR_ADDRESS_SPACE_ID=00 D*
 000100    E4D4D7C5  C46DC4C1  E3C17EC6  F160D7C1    D9E3C9E3  C9D6D540  D1D6C26D  D5C1D4C5    *UMPED_DATA=F1-PARTITION JOB_NAME*
 000120    7ED7D6E6  E2E3C1D9  E3000000  00000000    00000000  00000000  00000000  00000000    *=POWSTART                     *
 000140    00000000  00000000  00000000  00000000    00000000  00000000  00000000  00000000    *                              *
 000160 to  00100F  same as previous line
CYL-HD-REC 00032-13-002 DATA LEN   4112
 000000    00000002  00400000  00B9E000  C61A0000    C9D7E6D7  D6E6C5D9  070D0000  0029EB16    *            F   IPWPOWER          *
CYL-HD-REC 00032-13-003 DATA LEN   4112
 000000    00000002  00401000  00BB0000  C61A0000    E2E3C1D9  E340D6C6  40C6C9E7  C1C2D3C5    *            F   START OF FIXABLE*
  .
  .
  .
CYL-HD-REC 00034-03-001 DATA LEN   4112
 000000    00000002  0058F000  00BA0000  061A0000    00000000  FFFFFFFF  FFFFFFFF  00000000    *      0                       *
CYL-HD-REC 00034-03-002 DATA LEN   4112
 000000    00000001  E2C3E3F6  00000000  00000000    00000000  D3C2C440  002C0000  C6F14040    *    SCT6            LBD     F1  *
CYL-HD-REC 00034-03-003 DATA LEN   4112
 000000    00000004  00000000  00000000  00000000    00000000  FFFFFFFF  FFFFFFFF  00000000    *                              *
CYL-HD-REC 00034-03-004 DATA LEN   4112
 000000    00000008  00000000  00000000  00000000    00000000  FFFFFFFF  FFFFFFFF  00000000    *                              *
CYL-HD-REC 00034-03-005 DATA LEN   4112
 000000    00000008  00000000  00000000  00000000    00000000  FFFFFFFF  FFFFFFFF  00000000    *                              *
CYL-HD-REC 00034-03-006 DATA LEN   4112
 000000    00000008  00000000  00000000  00000000    5C5C5C5C  40E4D5E4  E2C5C440  5C5C5C5C    *                 **** UNUSED *****
```

This record format is the same as for a stand-alone dump on tape, and an AR DUMP to tape, as well as for the internal dump format within the VSEDUMP library.
The only difference is that for a stand-alone dump on tape, each partition is in a separate tape file.

Sharing Physical Tape Units

Symptom:

Customer is sharing tape drives between two or more VSE/ESA systems. Occasionally he (she?) receives error message "*0P03I Device=<cuu> in use by other system*".

Resolution:

Generally this problem is caused by a permanent assignment of a logical unit to the tape on the owning system. This prevents VSE/ESA from automatically releasing ownership of the tape when processing is finished.

VSE/ESA supports a tape drive assigned to multiple physical VSE/ESA systems. It is not possible to assign a physical device to multiple virtual machines under VM/ESA. VSE/ESA controls device access using the ASSIGN / UNASSIGN CCW commands (x'B7' / x'C7'), which are supported on most "modern" tape drives (3480 / 3490). These commands are similar to the dasd RESERVE / RELEASE commands.

1. It is generally not necessary to use the **DVCDN** / **DVDUP** or **OFFLINE** / **ONLINE** commands to control tape unit assignment.
2. Whenever VSE/ESA encounters a JCL **ASSGN** statement for a tape unit, it issues an ASSIGN CCW command to that drive.
3. Generally, when the assignment is released (at end-of-job, or via the **ASSGN <cuu>,UA** command) VSE/ESA issues an UNASSIGN CCW command to the device.
4. **OFFLINE <cuu>:** If a tape unit remains assigned to a VSE/ESA system even after all assignments have apparently been released, the **OFFLINE <cuu>** command will unconditionally release all assignments (including permanent assignments) and then issue an UNASSIGN CCW command to the tape device.
5. The **ONLINE <cuu>,HOLD** command will instruct VSE/ESA not to issue the UNASSIGN CCW command, and the tape unit will remain assigned to this VSE/ESA system until released by the **OFFLINE** command.
6. The **VOLUME** or **VOLUME TAPE** command will show the status (including volid) of all tape units accessible to this VSE/ESA system, but will not issue an ASSIGN CCW command. Thus, the **VOLUME** command may be used on multiple VSE/ESA systems. If it is issued at precisely the same time, one system may receive a "device busy" error, but re-issuing the command should work ok.
7. **NOASSGN**: This undocumented AR command was provided in the early 3480 days because of vendor product dependencies. This command prevents all ASSIGN / UNASSIGN CCW commands from being issued to tape units. Thus, any VSE/ESA system can write to any tape unit at any time, with the resultant danger of file contention / corruption. This command should only be used in the most dire of circumstances.
8. **NOAVR** command: When a cartridge is inserted into a drive and the tape is made ready, an interrupt is generated to all VSE/ESA systems physically connected to that drive. The VSE/ESA system will then initiate "Automatic Volume Recognition" processing and read the tape label. This will later be displayed by the **STATUS** command. Sometimes this "AVR" processing interferes with vendor products who are also doing device sensing and volume label management. The AR **NOAVR** command may be used to tell VSE/ESA to skip automatic volume sensing. The output of the **STATUS** command will be unpredictable. Tape ASSIGN or UNASSIGN CCW commands are not issued during AVR processing.

Installing Service from disk

Symptom:

Customer has received PTFs electronically, but cannot apply them serially due to internal co-requisites.

Resolution:

1. Copy PTFs to dasd using following jcl:

```
// JOB DITTO - COPY CARD (PTFS) TO DISK
// UPSI 1
// ASSGN SYS013,DISK,VOL=SYSWK2,SHR
// DLBL IJSYSPF,'PTF.FILES.ON.DISK',0,SD
// EXTENT SYS013,SYSWK2,1,0,15,225
// EXEC DITTO,SIZE=512K
$$DITTO SET EOD=$$$$$$$$
$$DITTO CS OUTPUT=SYS013,FILEOUT=IJSYSPF,RECFMOUT=FB,BLKSIZE=3440
// JOB UD51226
// OPTION CATAL
*  COMPONENT: 5686-066-05(06645C)
*  APARS FIXED: DY45336
*  SPECIAL CONDITIONS:
*    COPYRIGHT:            (C) COPYRIGHT IBM CORP.1998
*               LICENSED MATERIAL - PROGRAM PROPERTY OF IBM
*  COMMENTS:
      •
      • Remainder of PTF stack
      •
$$$$$$$$
/*
/&
```

2. Apply PTFs from disk using following jcl:

```
// JOB MSHP
// ASSGN SYS013,DISK,VOL=SYSWK2,SHR
// DLBL IJSYSPF,'PTF.FILES.ON.DISK',0,SD
// EXTENT SYS013,SYSWK2,1,0,15,225
// EXEC MSHP
  INSTALL SERVICE FROMDISK
/*
/&
```

PD/PSI Console Commands

ALTER { | space_id | partition} ,address

See "*VSE/ESA System Control Statements*" (SC33-6613)

Allows alteration of 16 bytes of storage currently resident (paged-in). Storage can be in supervisor, SVA, or problem program area (including CICS / VTAM / POWER).

```
dsply 1 , 300
AR 015 90F21028 18215811 00185801 0014F9F9            *.2.......99*
AR 015 1I40I  READY
alter 1 , 300
AR 015 1I42D  ADDRESS WITHIN SUPERVISOR OR SVA
AR+015
15 ignore
AR 015 OLD DATA:  90F21028 18215811 00185801 0014F9F9 *.2.......99*
AR 015 ENTER HEX DATA (1-16 BYTES)
AR+015
15 fff2
AR 015 1I40I
dsply 1 , 300
AR 015 FFF21028 18215811 00185801 0014F9F9           *.2........99*
AR 015 1I40I  READY
```

DSPLY (Use SHOW)

CANCEL cuu (,FORCE)

If device I/O has not completed, and is preventing a partition from going to eoj, the FORCE option will artificially flag the I/O as completed, and reset all operating system indicators.
This option must be used only with extreme caution!

// EXEC DTRIATTN , PARM = 'cmd'

Executes an attention routine command from JCL.

```
BG-0000 // PAUSE
0 // exec dtriattn,parm='map'
MAP
AR 0015  SPACE AREA    V-SIZE   GETVIS   V-ADDR   UNUSED NAME
AR 0015    S   SUP       716K                0          $$A$SUPX
      •
      •
      •
```

PD/PSI Console Commands

DOIO

DOIO cuu, {READ | WRITE} {CYL=cchhr | CNT=cchhr}

CYL => Read record and type to console
CNT => Read record and type Count-Key-Data to console

DOIO cuu, VOL1 type volume one label for specified device on console

DOIO cuu, CMD=cc ff llll => Perform command and type response on console
cc = CCW opcode
ff = Chain byte
llll = length

```
doio 120,cmd=04200020              <== Sense command
AR 0015   DATA WHICH WAS READ / PREPARED
 00000000 00000000 13000000 000000AB 49000005    *................*
 00000010 22006E31 70700000 00004CE2 00000000    *..>.......<S....*
AR 0015 1I40I  READY

doio 120,read,cyl=0023000d01  <== Read a record from dasd
AR 0015   DATA WHICH WAS READ / PREPARED
 00000000 1B224320 60004220 4018D258 6002A5AD    *....-...  .K.-.v.*
 00000010 92FF6000 9120B114 4710FBD8 43210000    *k.-.j......Q....*
 00000020 95561000 4740FBC0 41200056 4420FC16    *n....  .{........*
 00000030 BD0CE02C 4740FBE8 BE0C6059 41200059    *..\..  .Y..-.....*

doio 120,vol1                                   <== Read VOL1 label
AR 0015   DATA WHICH WAS READ / PREPARED
 00000000 E5D6D3F1 C3E3E2F1 F2F04000 4A000E01    *VOL1CTS120 .¢...*
 00000010 00000005 0000004B 0000004B 00002BF2    *............2*
 00000020 00000006 00000040 00000040 94032211    *.......  ...  m...*
 00000030 12540000 00000000 00000001 00000016    *................*
 00000040 00000016 00000000 00000000 00000000    *................*
AR 0015 1I40I  READY
```

The **DOIO** command was recently enhanced (DY45166) to improve the formatting on printed / translated output.

PD/PSI Console Commands

GETVIS

See "*VSE/ESA System Control Statements*" (SC33-6613)

GETVIS {**partition**} **{all}**
{**SVA**} **{detail}**

Allows dynamic monitoring of Getvis usage. "ALL" and "DETAIL" show GETVIS sub-pools at both the partition and SVA level. DETAIL also shows addresses of pages allocated to sub-pools.

```
getvis
AR 0015 GETVIS USAGE   SVA-24    SVA-ANY                     SVA-24    SVA-ANY
AR 0015  AREA SIZE:    1,372K     2,580K
AR 0015  USED AREA:      512K     1,220K MAX. EVER USED:      528K     1,240K
AR 0015  FREE AREA:      860K     1,360K LARGEST FREE:        860K       860K
AR 0015 1I40I  READY

getvis sva
AR 0015 GETVIS USAGE   SVA-24    SVA-ANY                     SVA-24    SVA-ANY
AR 0015  AREA SIZE:    1,372K     2,580K
AR 0015  USED AREA:      512K     1,220K MAX. EVER USED:      528K     1,240K
AR 0015  FREE AREA:      860K     1,360K LARGEST FREE:        860K       860K
AR 0015 1I40I  READY

getvis f2
AR 0015 GETVIS USAGE    F2-24     F2-ANY                     F2-24     F2-ANY
AR 0015  AREA SIZE:    4,096K     4,096K
AR 0015  USED AREA:    3,304K     3,304K MAX. EVER USED:    3,356K     3,356K
AR 0015  FREE AREA:      792K       792K LARGEST FREE:        792K       792K
AR 0015 1I40I  READY

getvis sva,all
AR 0015 GETVIS USAGE   SVA-24    SVA-ANY                     SVA-24    SVA-ANY
AR 0015  AREA SIZE:    1,432K     8,740K
AR 0015  USED AREA:      676K     1,772K MAX. EVER USED:      684K     1,780K
AR 0015  FREE AREA:      756K     6,968K LARGEST FREE:        748K     6,212K
AR 0015 SUMMARY REPORT
AR 0015 SUBPOOL       REQUEST  <--SVA-24-AREA---    --SVA-ANY-AREA-->
AR 0015 Default                                204K                 64K
AR 0015 ISTSVF                                 80K                272K
      •
      •
      •
AR 0015 IEXT0011                               0K                   4K
AR 0015 IJBCSM                                 0K                 136K
AR 0015 SUBPOOL TOTALS                       660K               1,004K
```

PD/PSI Console Commands

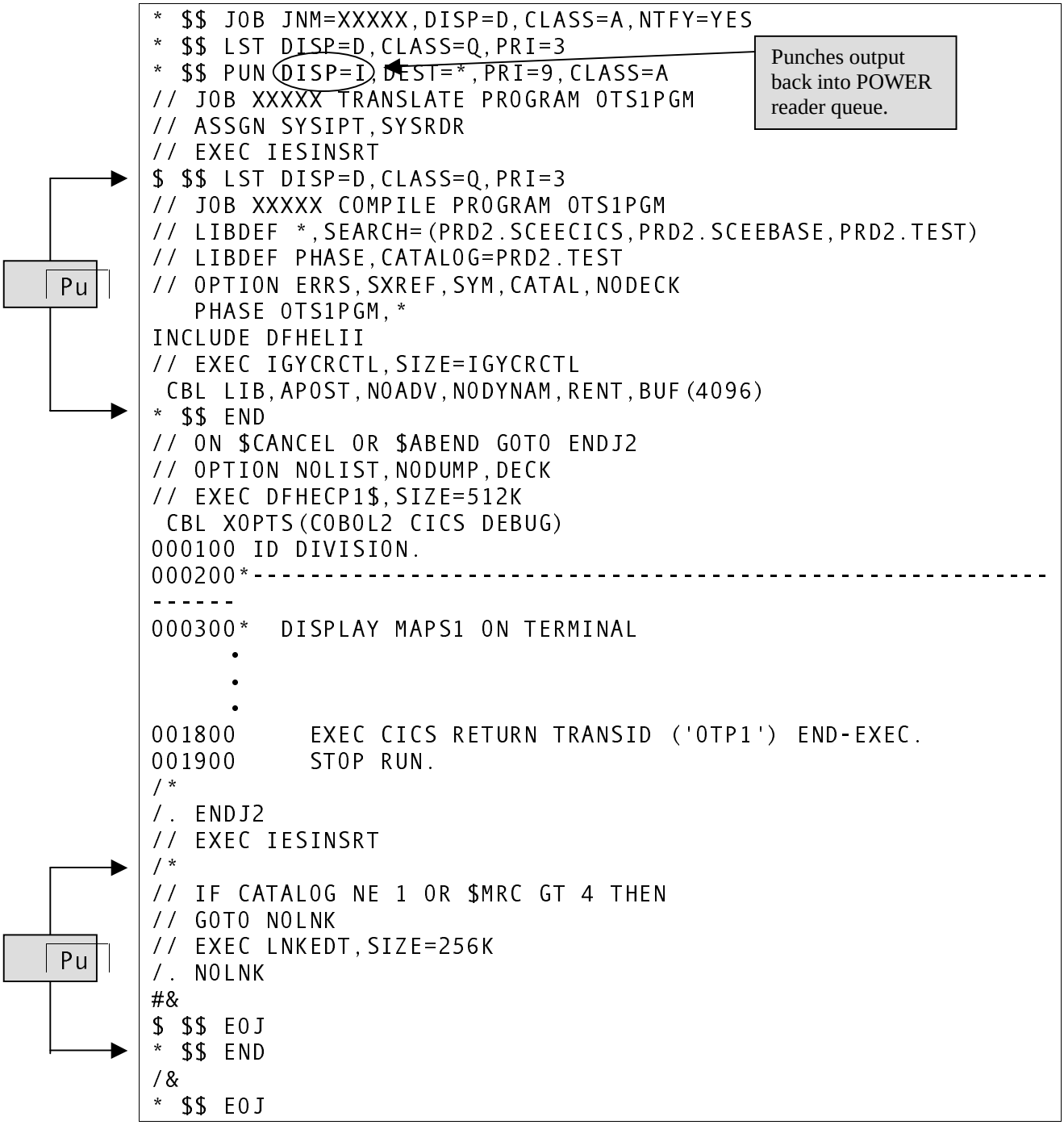
```
getvis sva,detail
AR 0015 GETVIS USAGE   SVA-24    SVA-ANY                SVA-24   SVA-ANY
AR 0015  AREA SIZE:    1,432K     8,740K
AR 0015  USED AREA:      676K     1,772K MAX. EVER USED:  692K    1,808K
AR 0015  FREE AREA:      756K     6,968K LARGEST FREE:    748K    6,212K
AR 0015 SUMMARY REPORT
AR 0015 SUBPOOL       REQUEST  <--SVA-24-AREA---    --SVA-ANY-AREA-->
AR 0015 Default                              204K                  64K
AR 0015                          0023F000-0023FFFF    02CBF000-02CBFFFF
AR 0015                          00244000-0024EFFF    02CB2000-02CBDFFF
AR 0015                          00250000-00250FFF    02C13000-02C15FFF
AR 0015                          00252000-00253FFF
AR 0015                          00255000-00256FFF
AR 0015                          00258000-00272FFF
AR 0015                          00275000-00275FFF
AR 0015                          00278000-00278FFF
AR 0015                          0027B000-0027EFFF
AR 0015                          00281000-00281FFF
AR 0015
AR 0015 ISTSVF                               80K                 272K
AR 0015                          0029B000-0029BFFF    02CA6000-02CA7FFF
AR 0015                          0029B000-0029BFFF    02CA6000-02CA7FFF
AR 0015                          002A2000-002A2FFF    02C91000-02C91FFF
AR 0015                          002AB000-002ABFFF    02C85000-02C86FFF
AR 0015                          002B0000-002B2FFF    02C7C000-02C7EFFF
AR 0015                          002B6000-002B6FFF    02C78000-02C78FFF
AR 0015                          002B8000-002B8FFF    02C69000-02C6DFFF
AR 0015                          002D0000-002D4FFF    02C1E000-02C53FFF
AR 0015                          002D6000-002DCFFF
   •
   •
   •
AR 0015 IEXT0011                               0K                   4K
AR 0015                                                02CDD000-02CDDFFF
AR 0015 IJBCSM                                 0K                 136K
AR 0015                                                02CDE000-02CFFFFF
AR 0015 SUBPOOL TOTALS                       660K               1,004K
AR 0015 1I40I  READY
```

PD/PSI Console Commands

IESINSRT

Punches JCL to allow multi-step jobs without using dasd as intermediate SYSPCH / SYSIPT.storage.

1. Punch all JCL until next * $$ END
2. Convert #& to /&
3. Convert $ $$ to * $$ (POWER JECL)




```
* $$ JOB JNM=XXXXX,DISP=D,CLASS=A,NTFY=YES
* $$ LST DISP=D,CLASS=Q,PRI=3
* $$ PUN DISP=I,DEST=*,PRI=9,CLASS=A
// JOB XXXXX TRANSLATE PROGRAM OTS1PGM
// ASSGN SYSIPT,SYSRDR
// EXEC IESINSRT
$ $$ LST DISP=D,CLASS=Q,PRI=3
// JOB XXXXX COMPILE PROGRAM OTS1PGM
// LIBDEF *,SEARCH=(PRD2.SCEECICS,PRD2.SCEEBASE,PRD2.TEST)
// LIBDEF PHASE,CATALOG=PRD2.TEST
// OPTION ERRS,SXREF,SYM,CATAL,NODECK
   PHASE OTS1PGM,*
INCLUDE DFHELII
// EXEC IGYCRCTL,SIZE=IGYCRCTL
 CBL LIB,APOST,NOADV,NODYNAM,RENT,BUF(4096)
* $$ END
// ON $CANCEL OR $ABEND GOTO ENDJ2
// OPTION NOLIST,NODUMP,DECK
// EXEC DFHECP1$,SIZE=512K
 CBL XOPTS(COBOL2 CICS DEBUG)
000100 ID DIVISION.
000200*-------------------------------------------------------
------
000300*  DISPLAY MAPS1 ON TERMINAL
      .
      .
      .
001800     EXEC CICS RETURN TRANSID ('OTP1') END-EXEC.
001900     STOP RUN.
/*
/. ENDJ2
// EXEC IESINSRT
/*
// IF CATALOG NE 1 OR $MRC GT 4 THEN
// GOTO NOLNK
// EXEC LNKEDT,SIZE=256K
/. NOLNK
#&
$ $$ EOJ
* $$ END
/&
* $$ EOJ
```

PD/PSI Console Commands

LOCATE

See "*VSE/ESA Supervisor Diagnosis Reference*" (LY33-9164)

LOCATE { } 'string ({FROM=<start>}{,TO=<end>}
{S} {,RUN}
{R}
{space_id}
{partition}

' is the single, special character that must be used to indicate that the following string is a character string. If the ' is missing, the string is assumed to be hexadecimal digits.

string is the series which the user wants to be located; limited to either 16 characters or, to 32 hexadecimal digits representing 16 bytes of storage. Any character or hexadecimal digit to be excluded from the scan, must be replaced by a . (dot).

```
map
AR 0015  SPACE AREA       V-SIZE   GETVIS   V-ADDR   UNUSED NAME
AR 0015    S   SUP          640K                0          $$A$SUPX
AR 0015    S   SVA-24      2304K    1600K    A0000    1600K
AR 0015    0   BG V        1280K     256K   600000   10752K

locate 'IKQBFA (from=A0000,to=600000
AR 0015   MATCH FOUND AT 001B2A2C
V001B2A20 ........ ........ ........ C9D2D8C2 06 *............IKQB*
V001B2A30 C6C14040 F1F5C340 C4E8F4F3 F5F3F140 06 *FA  15C DY43531 *
V001B2A40 900ED00C 189F45E0 99765830 D1589110 06 *î }  _É\rn¿ J¿j *
V001B2A50 A06C4780 90325830 D15C5030 D1081861 06 *+%_°î ¿ J*& J  /*
AR+0015

15
AR+0015   MATCH FOUND AT 002CEA84
V002CEA80 ........ C9D2D8C2 C6C1404D 40404040 06 *....IKQBFA (    *
V002CEA90 40404040 40404040 40404040 40404040 06 *                *
V002CEAA0 40404040 40404040 40404040 40404040 06 *                *
V002CEAB0 40404040 40404040 40404040 40404040 06 *                *

15 e
AR 0015 1I40I  READY
```

Be sure to terminate the **LOCATE** command by entering "**15**" plus any non-blank character, for instance "**15 E**". The attention routine remains locked (blocked) until the command is terminated.

PD/PSI Console Commands

MAP

See "*VSE/ESA System Control Statements*" (SC33-6613)

```
          {VIRTUAL}
   MAP    {  }
          {REAL}
          {BG}
          {Fn}
          {dyn_partition}
          {CLASS=       {ALL}}
                        {class}
          {SVA}
```

```
map
AR 0015  SPACE AREA       V-SIZE   GETVIS   V-ADDR   UNUSED NAME
AR 0015    S   SUP          636K                0          $$A$SUPX
AR 0015    S   SVA-24      2300K    1608K    9F000    1600K
AR 0015    0   BG V        1280K     256K   600000   10752K
AR 0015    1   F1 V         768K     832K   600000       0K POWSTART
AR 0015    2   F2 V        2048K    5120K   600000       0K CICSICCF
AR 0015    2   F2 V        2048K    5120K   600000       0K CICSICCF
   .
   .
   .
AR 0015    B   FB V         256K     256K   600000       0K
AR 0015    S   SVA-31      2888K    1208K  1200000
AR 0015        DYN-PA         0K
AR 0015        DSPACE      2048K
AR 0015        SYSTEM       192K
AR 0015        AVAIL       3904K
AR 0015        TOTAL      40960K   <----'
map f2
AR 0015  PARTITION: F2        SPACE-GETVIS......:    (N/A)
AR 0015  SPACE....: 2         ALLOC (VIRTUAL)...:    7168K  ADDR: 600000
AR 0015  STATUS...: VIRTUAL     SIZE............:    2048K
AR 0015  POWER-JOB: CICSICCF     EXEC-SIZE......:    3072K
AR 0015  JOBNUMBER: 1510        GETVIS..........:    5120K
AR 0015  JOBNAME..: CICSICCF     EXEC-GETVIS....:    4096K  ADDR: 900000
AR 0015  PHASE....: DFHSIP
AR 0015                       PFIX(BELOW)-LIMIT :     144K
AR 0015                                  -ACTUAL:      56K
AR 0015                       PFIX(ABOVE)-LIMIT :       0K
AR 0015                                  -ACTUAL:       0K
```

PD/PSI Console Commands

NOASSGN

Instructs VSE not to issue a Tape CCW Assign command when a shared tape drive is used.
(See write-up on page 30.)

NOAVR

Turns off Automatic Volume Recognition (dasd and tape). Requires an IPL to re-enable.
(See write-up on page 30.)

ONLINE [cuu] [,FORCE]

Forces an update of Device Characteristics (issues Read RDC CCW command, x'64') even if device was not previously in "device down" status. Available via APAR DY45299 for VSE/ESA 2.3 & 2.4.

READY [cuu]

Resets I/O status flag for this device, and generates a pseudo-interrupt.

PAUSE

See "*VSE/ESA System Control Statements*" (SC33-6613)

PAUSE [partition | luname | job] [,EOJ]

Pauses execution in the indicated partition at the next job control boundary (i.e. after currently executing phase / command completes) or at end-of-job, if "**,EOJ**" is specified.

SUSPEND

SUSPEND [task id | partition syslog id]

Pauses execution for the indicated task / partition immediately. Effectively moves partition out of dispatch pool. If a partition syslog id is specified (e.g. "BG", "F7"), the main task and all VSE sub-tasks are suspended.

RESUME

RESUME [task id | partition syslog id]

Resumes execution for the indicated task / partition immediately. Effectively moves partition back into dispatch pool.

PD/PSI Console Commands

REIPL

See "*Hints and Tips for VSE*" p. 62.

REIPL [CUU [, NOPROMPT]]

Does exactly what it sounds like. Use with extreme care. In fact, you may want to use the STACK command to disable this command.

```
reipl
AR+0015 1IxxD RE-IPL FROM CUU=120? REPLY YES OR NO
15 yes

 ************ LOGGING RESUMED AFTER VSE SYSTEM RE-IPL *************
BG 0000 0I04I IPLDEV=X'120',VOLSER=CTS120,CPUID=FFFF10009672
BG 0000 0J01I IPL=$IPLESA ,JCL=$$JCL
BG 0000 $$A$SUPX,VSIZE=120M,VIO=512K,VPOOL=64K,LOG
```

PWROFF

Initiates a hardware power off sequence. Designed for remote operations. Under VM, logs off the VSE virtual machine. Also a good candidate for the STACK command disabling.

```
pwroff
AR+0015 1I57D POWER-OFF THE CPU? REPLY YES OR NO
15 yes
 ... VSE is logged off, if under VM. ...
```

SHOW

See "*VSE/ESA Supervisor Diagnosis Reference*" (LY33-9164)

Displays up to 4K on the console. Memory will be paged in, if required.

SHOW	{id}	address [,length]
	{id,DSPNAME=name}	
	{id,SGT}	
	{id,PT}	

```
show f2,600000.3F
AR 0015   DATA  FOUND AT 00600000
V00600000 C4C6C8E2 C9D74040 07BD0000 00606B0A B6 *DFHSIP   ]   -, *
V00600010 00000000 00603F78 006000A8 00603280 B6 *     - ¹ - y - °*
V00600020 00601980 00000000 00608A18 008F1150 B6 * - °     -û    &*
V00600030 006C2800 00603380 00601A24 00605F78 B6 * %   - ° -   -^¹*
AR 0015 1I40I  READY
```

PD/PSI Console Commands

SIR

See "*Hints and Tips for VSE*" p. 42.

SIR **(blank)** (Full status display)
HELP | ? (List available SIR commands)
SYS | SYSLEVEL (Static status information)
RESET (Reset dynamic counters)
SMF [,VSE] = [ON | OFF | cuu] (Subsystem Measurement Facility)
(Some hardware SMF counters are inaccurate. "**,VSE**" counts from the VSE side.
MON [= id | ON [, NOSYM] | OFF [,FAST | COUNTER | BOUND]
(Turbo Dispatcher statistics)
MIH [= nnnnnn | ON | OFF] (Missing Interrupt interval)
CHPID [= id] (Channel paths defined to system)
CRWMSG = ON | OFF (Channel Report Word information)
VENDOR (Vendor product information)

```
sir
AR 0015 CPUID  VM = 9C24941096720000      VSE = FF04941096720000
AR 0015 VM-SYSTEM = VM/ESA  (LPAR)  2.4.0       0001
AR 0015 PROCESSOR = 9672-9C         USERID    = VSETEST3
AR 0015 PROC-MODE = ESA             IPL(140)    01:09:11   03/30/2000
AR 0015 SYSTEM    = VSE/ESA         2.4.1                  10/21/1999
AR 0015             VSE/AF          6.4.0       DY45196    09/27/1999
AR 0015             VSE/POWER       6.4.0       DY45131    07/19/1999
AR 0015 IPL-PROC  = $IPLESA         JCL-PROC  = $$JCL
AR 0015 SUPVR     = $$A$SUPX        TURBO-DISPATCHER (18) ACTIVE
AR 0015                             HARDWARE COMPRESSION ENABLED
AR 0015 SEC. MGR. = BASIC           SECURITY  = ONLINE
AR 0015 VIRTCPU   = 0000:01:31.742         CP = 0000:00:07.510
AR 0015 CPU-ADDR. = 0000(IPL)   ACTIVE
AR 0015   ACTIVE  = 0000:01:23.801   WAIT = 0063:11:51.958
AR 0015   PARALLEL= 0000:00:19.343   SPIN = 0000:00:00.000
AR 0015 CPU timings MEASUREMENT INTERVAL    0063:13:29.564
AR 0015 TASKS ATT.= 00015           HIGH-MARK = 00016    MAX = 00200
AR 0015 DYN.PARTS = 00000           HIGH-MARK = 00000    MAX = 00012
AR 0015
AR 0015 COPY-BLKS = 00003           HIGH-MARK = 00027    MAX = 01500
AR 0015 CHANQ USED= 00019           HIGH-MARK = 00025    MAX = 00226
AR 0015 PGIN  TOT.= 0000000000      EXP.AVRGE.= 0000000000/SEC
AR 0015 PGOUT TOT.= 0000000000
AR 0015       UNC.= 0000000000      EXP.AVRGE.= 0000000000/SEC
AR 0015       PRE = 0000000000      EXP.AVRGE.= 0000000000/SEC
AR 0015 LOCKS EXT.= 0000000554      LOCKS INT.= 0000006248
AR 0015      FAIL = 0000000014           FAIL = 0000000063
AR 0015 LOCK I/O  = 0000000000      LOCK WRITE= 0000000000
AR 0015 1I40I  READY
```

PD/PSI Console Commands

System Level (VSE/ESA 2.4.1)

// EXEC PROC=SPLEVEL

```
BG 0000 * ****************************************
BG 0000 *
BG 0000 *         VSE/ESA 2.4.1      10/21/1999
BG 0000 *
BG 0000 * ****************************************
BG 0000 * LICENSED MATERIALS - PROPERTY OF IBM      *
BG 0000 * 5686-066 AND OTHER MATERIALS (C) COPYRIGHT*
BG 0000 * IBM CORP. 1997 AND OTHER DATES            *
BG 0000 * ALL RIGHTS RESERVED.                      *
BG 0000 * US GOVERNMENT USERS RESTRICTED RIGHTS -   *
BG 0000 * USE, DUPLICATION OR DISCLOSURE            *
BG 0000 * RESTRICTED BY                            *
BG 0000 * GSA ADP SCHEDULE CONTRACT WITH IBM CORP.
BG 0000 * ****************************************
```

Updated for each FSU. Reflects level of last FSU installed. Never updated via PTF. Used by SIR.

Visible in any dump. Release updated only at release boundary. APAR num updated by each PTF. Used by SIR.

SHOW 2CC.40 (SGSRVLVL MACRO)

```
AR 0015   DATA  FOUND AT 000002C0
V000002C0 00000F4C 0024F004 0000FB89 0008E5E2 06 *...<..0....i..VS*
V000002D0 C561C1C6 40F5F6F8 F6F0F6F6 F0F640F4 06 *E/AF 568606606 4*
V000002E0 F5C34040 F6F4F040 C4E8F4F5 F1F9F640 06 *5C  640 DY45196 *
V000002F0 E5E2C54B C5E2C14B E2E4D7E7 40404040 06 *VSE.ESA.SUPX    *
```

SUBSID MACRO (SGSSID SUPERVISOR GENERATION MACRO):

Callable function. Updated at release boundary. Used by vendor products.

```
                  93214+*-------------------------------
                  93215+*   SUBSYSTEM LIST STARTING WITH SUPERVISOR ENTRY
                  93216+*-------------------------------------------------
03E8C8            93218+SSIDLIST DC 0D'0'               LIST AREA
03E8C8 0000       93219+         DC    XL2'00'          FLAGS
03E8CA 0000       93220+SSIDSUP  DC    XL2'00'          PARTITION-ID
03E8CC E2E4D740   93221+         DC    CL4'SUP '        NAME
03E8D0 06         93222+         DC    AL1(6)           VERSION NUMBER
03E8D1 01         93223+         DC    AL1(4)           RELEASE NUMBER
03E8D2 03         93224+         DC    AL1(0)           MODIFICATION NUMBER
03E8D3 06         93225+        DC    AL1(SSIDVLE-SSIDFB01) LENGTH OF VARIABLE PART
03E8D4 B8         93226+SSIDFB01 DC    BL1'10111000'
03E8D5 D0         93227+SSIDFB02 DC    BL1'11010000'    FLAGBYTE 2
03E8D6 60         93228+SSIDFB03 DC    BL1'01100000'    FLAGBYTE 3
03E8D7 00         93229+SSIDFB04 DC    BL1'00000000'    FLAGBYTE 4 RESERV.
03E8D8 000F       93230+         DC    H'15'     LENGTH OF LIB.CONCAT.CHAIN
```

LOCATE E2E4D74006

```
AR+0015   MATCH FOUND AT 0003E8CC
V0003E8C0 ........ ........ ........ E2E4D740 06 *............SUP *
V0003E8D0 06040006 B9DA2000 000F2C01 0020E2C5 06 *..............SE*
V0003E8E0 C3E20000 00002280 00C0D7E6 D9400604 06 *CS.......{PWR ..*
V0003E8F0 00002440 00A0E5E3 C1D40000 00002310 06 *...  ..VTAM......*
V0003E900 00B0C3C9 C3F40000 00004301 00B0E2E2 06 *..CIC4........SS*
V0003E910 E7400000 00004520 00B0C9C3 C3C60100 06 *X ........ICCF..*
```

PD/PSI Console Commands

STACK

See "*Hints and Tips for VSE/ESA*" p. 42

- Enables creation of a shortcut to a series of commands.
- Allows suppression or changing any VSE command.
- Abbreviates long VSE commands to just a few characters.
- Entire **STACK** command must not exceed 126 characters.

STACK	**(blank) | CLEAR**	(Clears all active synonym definitions)
	CLEAR=name	(Clears only specified synonym definition)
	SHOW [=name]	(Lists currently defined synonym(s))
	name | command | command | …	(may include special variables &0 .. &9 **name** is the synonym name. A series of commands may be associated with this name by separating them with "|".)

There is a sample program in Hints and Tips showing how to define synonyms during IPL using SVC 30.

```
stack bg | r rdr,pausebg
AR 0015 1I40I  READY
bg
AR 0015 1I40I  READY
AR 0015  R RDR,PAUSEBG
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R88I  OK
BG 0001 1Q47I   BG PAUSEBG 00023 FROM (SYSA) , TIME= 1:16:20
BG 0000 // JOB PAUSEBG
        DATE 03/16/1999, CLOCK 01/16/20
BG-0000 // PAUSE


stack pause | r rdr,pause&0
AR 0015 1I40I  READY
pause bg
AR 0015 1I40I  READY
AR 0015  R RDR,PAUSEBG
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R88I  OK
BG 0001 1Q47I   BG PAUSEBG 00261 FROM (SYSA) , TIME= 1:19:54
BG 0000 // JOB PAUSEBG
        DATE 03/16/1999, CLOCK 01/19/54
BG-0000 // PAUSE
```

PD/PSI Console Commands

STATUS

See "*VSE/ESA System Control Statements*" (SC33-6613) and
"*Hints and Tips for VSE/ESA*" p. 68

```
status
AR 0015 M12 CST          82 WAITING FOR I/O ON DEVICE=009
AR 0015 M20 AR           83 READY TO RUN
AR 0015 M21 BG           57 VSE/POWER WAITING FOR WORK
AR 0015 S3B     -F1      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 M22 F1           82 WAITING FOR I/O, OR ECB POSTING   POWER MAIN TASK
AR 0015 S40     -F2      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S41     -F2      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S42     -F2      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S43     -F2      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S44     -F2      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S45     -F2      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S46     -F2      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S47     -F2      82 VSE/POWER WAITING FOR WORK
AR 0015 M23 F2           82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S3C     -F3      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S3D     -F3      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S3E     -F3      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S3F     -F3      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 M24 F3           82 WAITING FOR I/O, OR ECB POSTING
AR 0015 M25 F4           57 VSE/POWER WAITING FOR WORK
AR 0015 M26 F5           57 VSE/POWER WAITING FOR WORK
AR 0015 M27 F6           57 VSE/POWER WAITING FOR WORK
AR 0015 M28 F7           57 VSE/POWER WAITING FOR WORK
AR 0015 M29 F8           57 VSE/POWER WAITING FOR WORK
AR 0015 M2A F9           57 VSE/POWER WAITING FOR WORK
AR 0015 M2B FA           57 VSE/POWER WAITING FOR WORK
AR 0015 S39     -FB      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 S3A     -FB      82 WAITING FOR I/O, OR ECB POSTING
AR 0015 M2C FB           82 WAITING FOR I/O, OR ECB POSTING   SECURITY SERVER
AR 0015 1I40I  READY

status f2
AR 0015 S40-F2 EVA10MST 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002C461C TIB=002C45A0 SAV=00405260
AR 0015 S41-F2 DFHEVID2 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002C48EC TIB=002C4870 SAV=00292400
AR 0015 S42-F2 DFHEVID1 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002C4BBC TIB=002C4B40 SAV=00292480
AR 0015 S43-F2 DFHEVID1 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002E607C TIB=002E6000 SAV=00292500
AR 0015 S44-F2 DFHSKTSK 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    SCB=0004980C PCB=0004A200 TCB=002E634C TIB=002E62D0 SAV=00292580
AR 0015 S45-F2 DTSC0PCM 82 WAITING FOR I/O, OR ECB POSTING
        .
        .
        .
```

PD/PSI Console Commands

```
status 240
AR 0015 SCHIB DEV  INT-PARM ISC FLG LP PNO LPU PI MBI  PO PA CHPID0-3 CHPID4-7
AR 0015 000A  0240 00003418   3  83 E0  00  20 F0 0000 F0 E0 24272A28 FFFFFFFF
AR 0015 CHPIDS NOT ACCESSIBLE:     28

status 480,all
AR 0015 SCHIB DEV  INT-PARM ISC FLG LP PNO LPU PI  MBI PO PA CHPID0-3
AR 0015 0010  0480 00003710   3  83 80  00  80 80 0000 80 80 0AFFFFFF
AR 0015            KEY SLCC FPIAUZEN FCTL ACTL SCTL CCW-ADDR DS CS CNT
AR 0015             0   0      00     40  0000  07  00027358 0C 00 000C
AR 0015 REQUEST IS STARTED             DEVICE IS BUSY
AR 0015   PUB=00003710   PUBX=0007A228   PUB2=00072288   P0WN=00003BBC
AR 0015  VCTE=000712FA  P0WNX=00237BC4
```

SCHIB	Subchannel number of the device that has been inspected.
DEV	Device number
INT-PARM	Interrupt parameter
ISC	Interruption subclass code
FLG	Flag field
LP	Logical Path Mask
PNO	Path Not Operational Mask
LPU	Last Path Used Mask
PI	Path Installed Mask
MBI	Measurement Block Index
PO	Path Operational Mask
PA 5	Path Available Mask
CHPID0-3	Channel Path Identifiers 0 through 3
CHPID4-7	Channel Path Identifiers 4 through 7
KEY	Storage protection key used at SSCH time
SLCC	Progress of I/O operation (cc = deferred condition code)
FPIAUZEN	Control bits
FCTL	Function control information (40 = start, 20 = halt, 10 = clear)
ACTL	Activity control information (04 = start pending, 02 = halt pending, 01 = clear pending, byte 2: 80 = subchannel active, 40 = device active)
SCTL	Status control information (10 = Alert status, 04 = primary I/O interrupt status, 02 = secondary interrupt status, 01 = status pending)
CCW-ADDR	Address+8 of last CCW that was executed
DS	Device Status information
CS	Channel status information
CNT	Residual count

PD/PSI Console Commands

TAPE UNL={UNL | REW}, DSPLY={VOL | JNM}

See "*Hints and Tips for VSE/ESA*", p. 71

UNL Prevents rewind-unload tape commands. Helpful with test systems, where you do not want to unload the tape.

DSPLY Changes the information normally displayed on the Load-Display (LED) of 3480, 3490, and 3590 tape cartridge devices. "JNM" will cause the volume and job name to alternately flash each 2 seconds.

TIME

See "*Hints and Tips for VSE/ESA*", p. 71

TIME [DATE=MM/DD/YYYY, CLOCK=HH/MM/SS]
[ZONE = EAST | WEST | zone-id/hh/mm

Allows operator to display or alter the current time or time zone. Extreme care must be exercises in using this command. Setting the time back may result in system mal-function, and render job accounting and/or journaling data invalid.

Requires APAR-# DY45385 (PTF-# UD51330) on VSE/ESA 2.4.x

VOLUME (tape)

See "*Hints and Tips for VSE/ESA*" p. 73

Enhancement to traditional VOLUME command, to show additional information for tape units. Output will be similar to following:

```
volume tape
AR 0015 CUU  CODE DEVICE-ID  VOLID  USE/STATUS - INFORMATION   CAPACITY
AR 0015 480  5400  3490-40   TAP634 BG                 BUFD      22356
AR 0015 481  5400  3490-40   *none* F4                 SYNC          0
AR 0015 482  5400  3490-40   ISMINE BG             2XF SYNC          1
AR 0015 483  5400  3490-40   TAP634 BG           NOT READY
AR 0015 484  5400  TAPE             SHARED         NOT OPER.
```

CUU device number as known to VSE
CODE device type code and mode setting currently active for this device
DEVICE-ID the device-type and model
VOLID VOL1 label (if any) of the media currently or last mounted on tape drive
USE/STATUS contains information about owner of the tape drive (if an owner exists)
INFORMATION media format mode (2XF vs XF) and whether operating as SYNC or BUFD
CAPACITY last block (record) read from, or written to the device. Zero indicates at load point. This value increments, allowing you to monitor tape job run progress.