B47

Debugging IMS Abends with IPCS

Jeff Maddix - (408)463-4956

IMS
Technical Conference

Sept. 27-30, 2004

Orlando, FL

RETURN TO INDEX

Agenda

- Although IMS rarely suffers outages due to abend conditions in production environments, should an abend occur, it is important to understand the source of the problem as quickly as possible so that conditions that caused the abnormal termination can be avoided in the future
- During this presentation we will:
  - Discuss abends
  - Explain the usefulness of key abend information elements common to most abends and indicate in which diagnostic sources you can find these elements
  - Describe how to obtain the diagnostic sources in which the key abend information elements exist

Abend Discussion

What is an Abend?

- What is an Abend?
  - The MVS system or application program issues a completion code when abnormally ending processing by a task, address space or system module
  - The completion code can be specified by
    - ABEND macro
    - CALLRTM macro
    - SETRP macro
    - Once one of the above macros have been issued, the code can be changed in recovery routines (FRR, ESTAE, SPIE, ESPIE, etc)

Abend Code Specification

- Abend codes can be specified as one of two basic types
  - SYSTEM
    - ABEND Macro Example:
      - ABEND X'0C4',,,SYSTEM
  - USER
    - ABEND Macro Example:
      - ABEND 476,,,USER

Abend Code Specification, Continued

- Based on the code type specification, the code is then represented in the completion code of various control blocks as follows:
  - 1 word hexadecimal completion code layout:
    - xxSSSUUU
      - Where
        - xx = Dump request information
        - SSS = System completion code
        - UUU = User completion code
- Completion code values based on the ABEND macro specification
  - ABEND X'0C4',,,SYSTEM
    - 000C4000
    - ABEND 476,,,USER
      - 000001DC

System Abend Codes

- System Abend Codes are normally issued by the MVS operating system
  - System-Detected Problems
    - The MVS system abnormally ends a task or address space if the system determines that the task or address space cannot continue processing and produce valid results
  - Hardware-Detected Problems
    - The MVS system abnormally ends a task with a completion code when the system receives control after a hardware-generated interruption that indicates an error in the task
  - Although not recommended, non MVS operating system applications could specify the abend code type as SYSTEM
    - This could be misleading because it is expected that MVS will issue system abend codes

System Abend Codes, Continued

- System Abend Codes are documented in the MVS Systems Codes Manual
- System Abend Codes are presented in hexadecimal within manuals, messages and formatted dump output
  - Example: Completion Code x'000C4000'
    - 0C4
    - S0C4
    - ABEND0C4

User Abend Codes

- User abend codes are normally issued by non MVS System routines. These are considered to be applications and include IMS, DB2, CICS, IBM Tools, non-IBM tools, user applications, etc.
  - Application-Detected Problems
    - An application program abnormally ends itself when it determines that it cannot continue processing and produce valid results.
    - The IMS subsystem might abnormally end a user application in an IMS dependent region if IMS determines that the application cannot continue processing and produce valid results

User Abend Codes, Continued

- User Abend Codes can be further categorized
  - Standard IMS control region user abend
    - Abends issued by IMS due to conditions which IMS determines that it cannot continue processing and produce valid results
      - Example: ABENDU0758, ABENDU3058
  - Control region sympathy abends
    - Issued to terminate the IMS subsystem due to dependent region abends received while in a sensitive state such as being in cross memory mode to the IMS control region or IMS DLI region, or an abend is received while holding an unrecoverable IMS latch.
      - Example: ABENDU0113 and ABENDU0780

User Abend Codes, Continued

- User Abend Codes can be further categorized, Continued
  - IMS pseudo abends
    - An IMS module detects a condition for which it determines that a user application cannot continue processing and produce valid results. Rather than issuing the abend macro immediately (which would cause an IMS control region abnormal termination), it sets codes in control blocks.
    - These codes are later examined by other IMS modules, which then issue the ABEND macro so that it can be attributable to the application program while not abnormally terminating the IMS control region
    - For some user abends, pertinent IMS control blocks from the control region are snapped to the IMS log in type X'67' log records. Described in IMS FAST and MESSAGES and Codes manuals
      - Example: Some ABENDU0476s, ABENDU08xx, etc,

User Abend Codes, Continued

- User Abend Codes can be further categorized, Continued
  - Non-IMS user abends
    - User abends issued by user applications running in IMS dependent regions
    - User abends issued by LE (Language Environment) while running in IMS dependent regions

User Abend Codes, Continued

- IMS user abend codes are documented in the IMS Failure Analysis Structure Tables (FAST) Manual and the IMS Messages and Codes Vol 1
- Other subsystem (DB2, MQSeries, CICS, etc.) user abend codes are documented in their associated product library manuals
- User application user abend codes are generally documented in the source code, but larger application systems may have manuals which describe user abend codes
- User abend codes are presented in decimal within manuals, messages, and formatted dump output
  - Example: Completion Code x'00000**1DC**'
    - 476
    - U0476
    - ABENDU0476

Key Abend Information Elements and Their Sources

Key Abend Information Elements - Overview

- Although the variety of possible abend codes is great, the core information required to solve each is very similar
- The following key abend information elements are useful to obtain across the vast majority of abends that occur in IMS environments.
  - Completion Code at time of abend
  - Failing ASID
  - Failing Module / Instruction / Offset
  - Messages
  - MVS System Trace Table Flow
  - MVS TCB/RB Flow
  - Primary Addressing Mode
  - PSW at abend
  - Registers at abend
  - Save Area Flow (Active SAP or REG13 area)
  - WLIC - Wait, Length, Interrupt Code field

Key Abend Information Elements - Overview, Continued

- Each of the key abend information elements is presented within this section along with a list of the various sources from which this information can be obtained.
  - The key abend information elements are presented in the same alphabetical sequence as shown on the prior page
- You will find that each key abend information elements can be found in multiple areas within a dump or system related datasets
  - Due to the complexity of today's computer environments and the varied timing of abend conditions, the documentation obtained at time of error may vary. Therefore, it is not always possible to find information related to an abend in one consistent manner. As a result, it is important to be able to search for information regarding the failure using various techniques in many locations.
  - You should use the eyecatchers presented along with an information source in any find command used to position the screen to the key abend information item being explained
    - Example: For completion code, Eyecatcher: "CODE SYS =" is shown
      - Appropriate find command would be: F "CODE SYS ="

Key Abend Information Elements - Completion Code Description

- The completion code indicates which abend code was issued. Once the completion code is obtained, analysis specific to the abend can begin
  - Hexadecimal completion code layout: xxSSSUUU
    - where xx = Dump request info

      SSS = System Abend Code

      UUU = User abend Code
- Descriptions for system abend codes can be found in the MVS Systems Codes Manual. System Abend Codes are presented in hexadecimal within manuals, messages and formatted dump output
- IMS user abend descriptions can be found in the IMS FAST Manual and the IMS Messages and Codes Vol 1
  - User abends should be translated to decimal prior to looking up the code in the manuals.
  - User abend codes are presented in decimal within manuals, messages, and formatted dump output - but not in the hexadecimal completion code field itself

COMPLETION CODE SOURCES

- Completion code sources
  - IMS Diagnostic Area
    - Eyecatcher: "SYSTEM ="
    - Eyecatcher: "USER ="
  - IMS Type 6705 / Type 67FD / Type 67FF Log Record
    - Eyecatcher: "CODE SYS ="
    - Eyecatcher: "USER ="
  - IPCS STATUS

Most reliable

    - Eyecatcher: "abend code:"
  - IPCS SUMDUMP
    - Eyecatcher: "Completion Code"

Completion Code Sources, Continued

- Completion code sources, continued
  - IPCS WORKSHEET
    - Eyecatcher: "ABEND"
  - MSGBPE0006I
    - Eyecatcher: "ABEND"
  - MSGDFS629I
    - Eyecatcher: "SYS"
    - Eyecatcher: "IMS"
  - MVS LOGREC
    - Eyecatcher: "ABEND CODE"

Completion Code Sources, Continued

- Completion code sources, continued
  - MVS RTM2WA Summary — Great for multiple abends
    - Eyecatcher: "Completion code"
  - MVS SDWA
    - Eyecatcher: "ABCC"
  - MVS System Trace Table
    - Eyecatcher: "SVC D" see UNIQUE-3 column
  - MVS TCB / RB Flow
    - Eyecatcher: "CMP"
    - Eyecatcher: "T C B S U M M A R Y" see CMP column

Key Abend Information Elements - Failing ASID Description

- The failing ASID is required to ensure that the dump formatting options are correct when an ASID is required as a parameter specification
  - ASID is used as a parameter in the "SUMMARY FORMAT ASID(x'nn')" command
  - ASID is used as a search parameter when viewing the IMS DWAs (Dispatch Work Areas)
    - Example: F 'ASIDS 00D2'
  - Specified in IPCS Browse mode when formatting diagnostic blocks such as the SDWA
  - etc....
- The failing ASID is the ASID under which the failing work unit was running when the abend was issued. The failing ASID is the 'Home ASID' for the work unit from an MVS perspective.

Failing ASID Sources

- Failing ASID sources (Failing ASID = Home ASID)
  - ASCB
    - Eyecatcher: "ASID"
  - IPCS STATUS  Most reliable
    - Eyecatcher: "Home ASID:"
  - IPCS SUMDUMP
    - Eyecatcher: "Error ASID"
  - IPCS WORKSHEET
    - Eyecatcher: "HASID"

Failing ASID Sources, Continued

- Failing ASID sources (Failing ASID = Home ASID), continued
  - MVS LOGREC
    - Eyecatcher: "HASID"
  - MVS RTM2WA Summary
    - Eyecatcher: "Error ASID"
  - MVS SDWA
    - Eyecatcher: "ASID"
  - MVS System Trace Table
    - Eyecatcher: See the "ASID" column of the failing trace entry

Key Abend Information Elements - Failing Module / Instruction / Offset Description

- Knowing the failing module, instruction, and offset aids you when resolving the source of the abend
  - Often necessary when referencing manuals for abend descriptions
    - Many IMS user abends are issued by more than one module and/or at different offsets within a module
    - Aids in finding APARs that might resolve your problem
      - APAR descriptions usually mention the failing module/instruction/offset
        - APARS can be searched at the IMS Support website

          Goto URL: http:// www.ibm.com/ims

          From the IMS home website

          Select the Support option in the left menu

          From the IMS Support page,

          Select Authorized Program Analysis Reports (APARS)

Failing Module / Instruction / Offset Sources

- Failing module sources
  - IMS Diagnostic Area
    - Eyecatcher: "IN MODULE"
  - IPCS Browse Mode

Most control, reliable

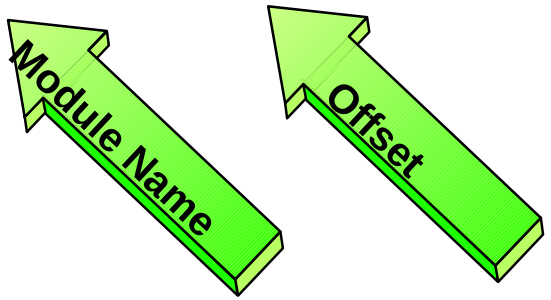
    - Using the Primary ASID, scan backwards from the PSW address looking for the module eyecatcher.
    - Be aware that some non-IMS software products place their eyecatcher at the end of the module
  - IPCS STATUS
    - Eyecatcher: "Load module name"
  - IPCS SUMDUMP
    - Eyecatcher: "MODN"

Failing Module / Instruction / Offset Sources, Continued

- Failing module sources, continued
  - IPCS WHERE aaaaaaaa ASID(x'nn')
    - Use the PSW at time of error as the object of the WHERE command
      - Failing PSW = 323E5A5A B23E5A5A
        - Command "IPCS WHERE 323E5A5A ASID(x'57B') resulted in:

          ASID(X'057B') 323E5A5A. DFSYFD00+3B52 IN EXTENDED CSA
  - MSGBPE0006I
    - Eyecatcher: "MODULE ID ="
  - MSGDFS629I
    - Eyecatcher: "MODID ="

Quick and easy when it returns output



Failing Module / Instruction / Offset Sources, Browse Mode Example

- **Failing module browse mode example**
  - **The failing module can be found in IPCS BROWSE mode by locating the PSW at Abend address for the primary ASID, then scanning backwards for the module eyecatcher in the EBCDIC display portion of the STORAGE panel output.**
  - **NOTE: Be sure to use the Primary ASID when accessing storage, which may be different than the Home ASID.**

Example: From the RTM2WA

Completion code **840C4000**

EC PSW at time of error **077C0000 8DCD8FDE 00040011 0F33D400**

```
0DCD8EA0 47F0F034 2EC4C6E2 C3E2E3F0 F04EE2F2  |.00..DFSCST00_S2|
0DCD8EB0 F0F34EE2 D7F5F34E F5F1F04E F0F461F0  |03_SP53_710_04/0|
0DCD8EC0 F261F0F2 4EF1F54E F4F1D7D8 F5F3F9F4  |2/02_15_41PQ5384|
0DCD8ED0 F440C100 90ECD00C 18CF58D0 D0081891  |4 A.../...//...j|
```

The failing module in this case is DFSCST00 at APAR level PQ53844. Note that most of the modules in IMS will begin with a x'47F0nnnn' instruction, aligned on a double word boundary, to branch around the module eyecatcher.

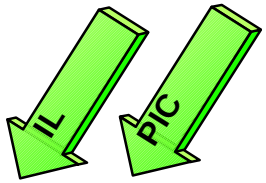
Failing Module / Instruction / Offset Sources, Browse Mode Example, Cont.

- **Failing instruction**
  - **The Failing Instruction can be found by locating the PSW at abend in IPCS BROWSE mode. The Instruction Length should be subtracted (except in the case of ABEND0C4 or ABEND0C5 PIC10, PIC11, PIC38, PIC39, PIC3A, PIC3B) from the PSW address.**

Example:

Completion code 840C4000

EC PSW at time of error 077C0000 8DCD8FDE 00040011 0F33D400




```
0DCD8FD0  100001255 47D0C14E 1B665060 10005860    |. . . . . / A. . .&- . . . - |
0DCD8FE0  50005010  50001815  12564720  C13E1841   |&.&.&. . . . . A . . . |
```

Using the PSW information above, the failing instruction from the block of storage is 58605000. The instruction length of 4 does not need to be subtracted because the abend is ABEND0C4 and the Program Interrupt Code is PIC11.

Failing Module / Instruction / Offset Sources, Browse Mode Example, Cont.

Failing Offset

- The Failing Offset can be determined by subtracting the Module entry point address from the PSW at Abend, minus the instruction length (except in the case of ABEND0C4 or ABEND0C5 PIC10, PIC11, PIC38, PIC39, PIC3A, PIC3B)

Example:

Completion code **840C4000**

EC PSW at time of error **077C0000 8DCD8FDE 00040011 0F33D400**

Because the abend code is ABEND0C4 and the PIC (Program Interrupt Code) is 11, the instruction length does not need to be subtracted.

0DCD8FDE

- 0DCD8EA0

x'13E'

MVS System Trace Table Flow Description

- The MVS System Trace is useful for many varieties of MVS problems. It is often the only means of reconstructing problem origins. The larger the trace table specification, the better our chances of diagnosing some of the more intricate problems encountered while running IMS
  - IMS recommends setting the MVS System Trace table size to 999K
  - The MVS command "TRACE ST,999K" can be specified in the MVS COMMNDxx PARMLIB member so that the trace table size is in effect at IPL time. Without this specification, the MVS default size is 64K
  - Note that the System Trace Table is page fixed. Installations that are tight on real page frames should take this into consideration.

MVS System Trace Table Flow Sources

- MVS System Trace Table flow sources
  - MVS System Trace Table
    - Eyecatcher: "SYSTEM TRACE TABLE"
    - Eyecatcher: "SVC D"
      - This represents the ABEND SVC
      - With an ABEND0C4, the "PGM" entry prior to the ABEND SVC should indicate the program interrupt code which caused the ABEND0C4
        - See trace example following the source information

MVS System Trace Table Flow Sources

- MVS System Trace Table flow sources
  - MVS System Trace Table
    - Eyecatcher: "*"
      - Unusual conditions are marked by * (asterisk) before the trace identifier
    - The address portion of the PSW at time of error can be used to find the associated PGM entry for ABEND0C4
      - Eyecatcher: "aaaaaaaa"
        - where aaaaaaaa is the address portion of the PSW at time of error

MVS System Trace Table Flow Example 1

- IPCS STATUS abend info for MVS System Trace Table example on next page

```
IPCS OUTPUT STREAM ------------------------------------- Line 71 Cols 1
Command ===> _                                            SCROLL ===> C
   Symptom              Description
   -------              -----------
   RIDS/DFSYFD00#L      Load module name: DFSYFD00
   RIDS/#UNKNOWN        Csect name: #UNKNOWN
   AB/S00C4             System abend code: 00C4
   PRCS/00000011        Abend reason code: 00000011
   REGS/0F018           Register/PSW difference for R0F: 018
   REGS/0A012           Register/PSW difference for R0A: 012

 PSW: 077C2000 B23E5A5A   Instruction length: 04   Interrupt code: 0011
 Failing instruction text: B0004780 A02E9120 20404780
 Translation exception address: 7FFFF000

 Registers 0-7
 GR: 00000000 25ACB05C 7FFFF800 000000DC  25ACB10C 00000000 0000007E 25ACB0DC
 AR: 00000000 00000000 00000000 00000000  00000000 00000000 00000000 00000000
 Registers 8-15
 GR: 323E5A84 00204000 B23E5A48 25ACB10C  B23E1F08 25ACAE88 B23E2DEA 323E5A42
 AR: 00000000 00000000 00000000 00000000  00000000 00000000 00000000 00000000

 Home ASID: 057B    Primary ASID: 057B    Secondary ASID: 057B
```

Abend Code

PSW

PIC

PASID

MVS System Trace Table Flow Example 1, Continued


```
IPCS OUTPUT STREAM ----------------------------------------------------
Command ===> ip systrace asid(x'57b') exclude(BR)_            SCROLL ===> CS
                                                  80000000
 05 057B 008AE890   DSP        070C0000 B23CE088  00000000 008AE890 008AF450
 05 057B 008AE890   SVC     3C 077C0000 B54C9F6C  00000000 00000100 39D2AA6C
 05 057B 008AE890   SVCR    3C 077C0000 B54C9F6C  00000000 00000000 39D2AA6C
 05 057B 008AE890   SVC     78 077C0000 B54CC46E  00000070 000000FC 00000000
 05 057B 008AE890   SVCR    78 077C0000 B54CC46E  00000000 00000100 39200060
 05 057B 008AE890   SVC     78 077C1000 B54CC526  00000001 000000FC 39200060
 05 057B 008AE890   SVCR    78 077C1000 B54CC526  00000000 000000FC 39200060
 05 057B 008AE890   SVC      A 077C1000 B54D730A  0043B690 00001000 B54D7000
 05 057B 008AE890   SVCR     A 077C1000 B54D730A  00000000 00001000 B54D4000
 05 057B 008AE890   PGM    011 077C2000 B23E5A5A  00040011 7FFF
 05 057B 008AE890  *RCVY PROG                     940C4000 00000011 00000000
 05 057B 008AE890   SSRV   12D          8115ACA8  008AE890 000C8000 FF3A0000
                                                  00000000
 05 057B 008AE890   SSRV   12D                    008AE890 000B8000 00000000
                                                  00000000
 05 057B 008AE890   DSP                8115D2FA  00000000 00000000 25ACB05C
 05 057B 008AE890  *SVC      D 077C2000 8115D2FC  323E5A42 00000000 25ACB05C
```

PIC

PSW

Abend Completion Code

Error entries marked with *

Error entries marked with *

MVS System Trace Table Flow Example 2

- ABEND0C1 example for SLIP dump with SYSTRACE BR=ON Specified
  - SLIP Command:

```
SLIP SET,IF,ID=IMS1,A=SYNCSVCD,RA=(00000000,00000004),
 JOBLIST=(IMSCTL,IMSDLI,DBRC),JSPGM=DFSRRC00,
 SDATA=(CSA,PSA,RGN,SQA,SUM,TRT),END
```

  - IPCS Command: IP LIST TITLE

```
LIST 00. LITERAL LENGTH(X'11') CHARACTER
00000000 | SLIP DUMP ID=IMS1
```

MVS System Trace Table Flow Example 2, Continued

- **ABEND0C1 example for SLIP dump with SYSTRACE BR=ON Specified, Continued**
  - **IPCS ANALYSIS, STATUS PANEL OUTPUT:**

```
IPCS OUTPUT STREAM ----------------------------------- Line 27
Command ===> _                                         SCROLL
 PSW=477C0C00 80000002
     (Running in PRIMARY, key 7, AMODE 31, DAT ON)
     ENABLED FOR PER I/O EXT MCH
    ASID(X'00C3') 02. STRUCTURE(Srb)+02 IN PSA
    ASID(X'00C3') 02. STRUCTURE(Psa)+02 IN PSA
    ASID(X'00C3') 02. IEAVFX00+02 IN PSA
   ASCB205 at EB9880, JOB(IMSRRM20), for the home ASID
   ASXB205 at 9FDE88 and TCB205F at 9E2A80 for the home ASID
   HOME ASID: 00CD PRIMARY ASID: 00C3 SECONDARY ASID: 00CD

   General purpose register values
     Left halves of all registers contain zeros
      0-3  00000000  1A7F7060  1A94A5EA  1A949630
      4-7  20231D40  20231F68  00000004  1A7F7060
      8-11 1FDE4534  1A9B2218  20231D40  0003C9C0
     12-15 00043C70  1A7F76F0  800440D6  00000000
```

PSW

Primary ASID - Instructions

BALR

MVS System Trace Table Flow Example 2, Continued

- **ABEND0C1 example for SLIP dump with SYSTRACE BR=ON Specified, Continued**
  - **SYSTRACE OUTPUT: IPCS SYSTRACE ASID(x'CD')**

```
IPCS OUTPUT STREAM ------------------------------------------------------------------------------------------ Line 0 Co
Command ===> _                                                                                               SCROLL =
 ********************************************************* TOP OF DATA ***********************************************

 ------------------------------------------------- SYSTEM TRACE TABLE ------------------------------------------------
 --                                                 ------------------                                              --
 --                                                 ------------------                                              --
 PR ASID WU-ADDR-  IDENT CD/D PSW----- ADDRESS-  UNIQUE-1 UNIQUE-2 UNIQUE-3  PSACLHS- PSALOCAL PASD SASD TIMESTAMP-RECORD
                                                UNIQUE-4 UNIQUE-5 UNIQUE-6  PSACLHSE

 09-00CD 009E2A80   DSP       478D3C00 86754590  00000000 000CA710 00000000  00000000 00000000 00CD 00CD BA66BBF81AE16541
 09-00CD 009E2A80   BR          0B71F0   0BC2A8 06754398   0B71F0   0BC2A8 06754398   097610 1FD3B8F8 000BFAAC 066B86C0
                    BR          0B71F0   0BC2A8 06754398   0B71F0   0BC2A8 06754398   0B71F0   0BC2A8 06754398   0B71F0
                    BR          0BC2A8 06754398   019FE0   01DC40   0A8728   0A7EE8 0009D308
 09-00CD 009E2A80   SVC    78 478D0C00 8009D378  00000214 00020000 00000000                                BA66BBF81AE37201
 09-00CD 009E2A80   BR        00FC6620 00FE250C 02FE9E88 00FFFB88 00FFF67E 00FE2A00 014DA0E4 013CC6C4 016AE5E0 00FE240C
 09-00CD 009E2A80   MODE  ... 64       016AF038
 09-00CD 009E2A80   MODE  ... 24 OR 31       00_016AF0C0
 09-00CD 009E2A80   BR        016383F0
 09-00CD 009E2A80   MODE  ... 64       016AF1AA
 09-00CD 009E2A80   MODE  ... 24 OR 31       00_016AF1C8
 09-00CD 009E2A80   MODE  ... 64       016AF1E0
 09-00CD 009E2A80   MODE  ... 24 OR 31       00_016AF1E6
 09-00CD 009E2A80   MODE  ... 64       016AF038
 09-00CD 009E2A80   MODE  ... 24 OR 31       00_016AF0C0
 09-00CD 009E2A80   MODE  ... 64       016AF038
 09-00CD 009E2A80   MODE  ... 24 OR 31       00 016AF0C0
```

MVS System Trace Table Flow Example 2, Continued

- **ABEND0C1 example for SLIP dump with SYSTRACE BR=ON Specified, Continued**
  - **Find the error entry: F '*PGM'**
  - **Branches show target BAL, BALR, BAS, BASSM, etc.**
    - **Does not show return BR 14's**

```
IPCS OUTPUT STREAM ------------------------------------------------------------------------------------------ Line 4937 Cols 1
Command ===> _                                                                                                SCROLL ===>
 02 00CD 009E2A80   DSP       470C0C00 9B0B5FE0  00000000 009E2A80 009E2840  00000000 00000000 00C3 00CD BA66BBF8F2B67802
 02 00CD 009E2A80   BR        1B0B4388 00B97648 1AB90000 1ABB110C 00FE250C 1AB9B538 1AB90000 1AB9BD20 00FE2A00 0000EEA6
                    BR        1ACA3218 1ABABB50 0000DB28 0000DD3A 0000EEA6 1ACA3218 1B13FE70 1ABABB50 0000DB28 0000F6C4
                    BR        1ACA3218 1B0B6508 1B140354 00B97648 1AB90000 1ABB110C 1AB91E88 1AB90000 1AB9218C 1B0B4388
                    BR        1AB8EBBC 1AB8F184 1AB8E960 00B97648 1AB90000 1ABB110C 00FE250C 1AB95780 1AB90000 1AB961C0
                    BR        1AB91458 1AB90000 1AB91C3A 1AB91040 1AB90000 1AB91340 0149BB10 00FE250C 00FE2A00 00FE2A00
                    BR        1AB8FA30
 02 00CD 009E2A80   PT    ...   8      00BB06F7           00CD
 02 00CD 009E2A80   BR        1FD03C40   0B76A0   0A7EE8   01CC00   01A1F0   0750F6   079150   0A8728   0A7EE8   0971C8
                    BR          0A6560   01A150   0A6560   01A268   0A8728   0A7EE8   096CA2 00BB0620 1FD03C40
 02 00CD 009E2A80   PC    ...   8      00BB06F7           0FE01
 02 00CD 009E2A80   BR        1AB8E710 1AB8E960 00B97648 1AB90000 1ABB1[illegible]250C 1AB97170 1AB90000 1AB97C38 1[illegible]
                    BR        1AB90000 1AB91C3A 1AB91040 1AB90000 [illegible] 0149BB10 00FE250C 00FE2A00 00FE[illegible]FA30
                    BR        1B0B6E50 1B0B4388 1AB8E9E4 1B13[illegible]3E77C 1ACA3218 1AB8FC78 1B1418D8 1[illegible]1AB8EA6A
                    BR        1AB8EAC0 1B13C4D8 1B1402F[illegible] 1AB90000 1ABB110C 00FE250C 1AB[illegible]90000 1AB95480
                    BR        00FE2A00 1B13E8AE 1ACA3[illegible]3FE70 0003C9C0 1ABABB50 0000DB28 0[illegible]0000E58C 0000E8E2
                    BR        0000EEA6 1ACA3218 00000000 0147A0A8 0147A842 0147A8CC 0147A95E 0147A7CA 0147A604 01034846
 02 00CD 009E2A80  *PGM   081 477C0C00 80000002  00020081 00000000           00000000 00000000 00C3 00CD BA66BBF8F2BCB422
                                                          009AA000           00000000
 02 00CD 009E2A80   BR        00FC6620 016B2B60 00FE250C 00FFF2B2 017E29F8 00FE230C 017DC458 01689320 01689320 01689320
                    BR        01689320 01689320 01689320 01689320 017E0C98 0168A6E0 017E41C0 00FE20D8 00FE240C 00FD2C2C
```

*PGM

Branch to zeros*

MVS Gaining conrol*

MVS TCB / RB Flow Description

- The TCB / RB flow is very useful for the full spectrum of system abend codes.
  - Concentrate on the OPSW, WLIC, RB type, CDE, and register contents when analyzing a TCB / RB flow. Remember that a WLIC content of 000200xx usually indicates an SVC was issued (SVC xx), especially when the RB is followed by an SVRB.
  - Generally, the failing TCB will contain the abend completion code in the TCBCMP field at x'10' into the TCB.
  - The TCBRBP field (TCB offset x'00') will point to the last RB in the RB chain. The RB LINK field can be used to obtain the remaining RBs. Each RB LINK field points to the prior RB. The first RB will contain the TCB address in its LINK field.
    - Note it is possible to have only one RB in the TCB/RB flow. In the case, the RB pointed to by the TCBRBP field will contain a LINK field that points to the TCB
  - Keep in mind that the registers for any given RB are stored in the following RB. The registers for the last RB (TCBRBP TCB offset x'00') are stored in the TCB.
  - RB types include PRB (Program Request Block, IRB (Interrupt Request Block) and and SVRB (Supervisor Request Block).

MVS TCB / RB Flow Sources

- MVS TCB/RB flow sources
  - IPCS CMD: CBF aaaaaaaa STR(TCB) and CBF aaaaaaaa STR(RB)
    - Eyecatcher: "TCB:"
    - Eyecatcher: "RB:"
  - MVS Formatted Dump
    - Eyecatcher: "TCB:" - Find the TCB(s) with a non-zero CMP field
      - The formatted RB flow will follow the TCB
      - The "T C B S UM M A R Y" can be referenced to determine which TCBs contain the non-zero CMP field
  - SYSUDUMP/SYSABEND Dump
    - Eyecatcher: "TCB:" - Find the TCB(s) with a non-zero CMP field
      - The formatted RB flow will follow the TCB
      - The "T C B S U M M A R Y" can be referenced to determine which TCBs contain the non-zero CMP field

Primary Addressing Mode Description

- The primary addressing mode is extremely important because it indicates which address space the failing task was accessing storage within for instruction fetch.
  - Much analysis could be performed incorrectly unless the Primary ASID is used as the ASID specification for storage access in IPCS BROWSE mode.
- NOTE: The primary ASID information in the RTM2WA and IPCS WORKSHEET may not be correct for the actual "Time of Error" event.

Primary Addressing Mode Sources

- Primary Addressing Mode Sources
  - IPCS STATUS  Most reliable
    - Eyecatcher: "PASID"
  - MVS LOGREC
    - Eyecatcher: "PRIMARY ASID"
  - MVS SDWA
    - Eyecatcher: "PRIM"
  - OS/390 System Trace table
    - PASD column of trace entry
  - MVS TCB / RB Flow
    - For the OPSW in given RB, use it's associated XSB block.
    - XSB field eyecatcher: "PASID"

Primary Addressing Mode Example

- **IPCS Analysis, Status Panel Output:**

```
IPCS OUTPUT STREAM ------------------------------------- Line 27
Command ===> _                                            SCROLL
 PSW=477C0C00 80000002
     (Running in PRIMARY, key 7, AMODE 31, DAT ON)
     ENABLED FOR PER I/O EXT MCH
    ASID(X'00C3') 02. STRUCTURE(Srb)+02 IN PSA
    ASID(X'00C3') 02. STRUCTURE(Psa)+02 IN PSA
    ASID(X'00C3') 02. IEAVFX08      IN PSA
   ASCB205 at EB9880, JOB(BRRM20), for the home ASID
   ASXB205 at 9FDE88, TCB205F at 9E2A80 for the home ASID
   HOME ASID: 00CD PRIMARY ASID: 00C3 SECONDARY ASID: 00CD

   General purpose register values
     Left halves of all registers contain zeros
      0-3  00000000  1A7F7060  1A94A5EA  1A949630
      4-7  20231D40  20231F68  00000004  1A7F7060
      8-11 1FDE4534  1A9B2218  20231D40  0003C9C0
     12-15 00043C70  1A7F76F0  800440D6  00000000
```

Home ASID - TCB Resides

Primary ASID - Instructions

PSW at Abend Description

- The PSW address is, of course, essential because it can be used as a starting point to look for the module that failed or to verify module information presented in other control block formatting.
  - This assumes that the failure is not due to the PSW itself being invalid. If the registers look in synch with the PSW, then dumped storage can be accessed and eyecatchers can be examined while scanning backwards from the PSW address until you find a module identifier.
  - To relate a module identifier to a module name, see the "Save-Area-ID-To-Module Cross-reference Table" in the IMS Diagnosis Guide and Reference.
- The PSW address at time of abend usually points to the next instruction to be executed.
  - If ABEND0C4 or ABEND0C5 occurs and the PIC (Program Interrupt Code) field contains X'0010', X'0011', X'0038', X'0039', X'003A', or X'003B', the PSW points directly to the instruction that failed.

PSW at ABEND Sources

- PSW at abend sources
  - IMS Diagnostic Area
    - Eyecatcher: "PSW AT ERROR"
  - IMS Type 6705 / Type 67FD / Type 67FF Log records
    - Eyecatcher: "SDWA"
    - Add x'76' to the beginning of the SDWA eyecatcher to reach the PSW
  - IPCS STATUS
    - Eyecatcher: "PSW:"
  - IPCS SUMDUMP
    - Eyecatcher: "EC PSW at time of error"

PSW at Abend Sources, Continued

- PSW at abend sources, continued
  - IPCS WORKSHEET
    - Eyecatcher: "PSW"
  - MSGBPE0006I
    - Eyecatcher: "PSW ="
  - MSGDFS629I
    - Eyecatcher: "PSW AT ERROR"
  - MVS LOGREC
    - Eyecatcher: "PSW:"

PSW at Abend Sources, Continued

- PSW at abend sources, continued
  - MVS RTM2WA summary
    - Eyecatcher: "EC PSW at time of error"
  - MVS SDWA
    - Eyecatcher: "PSW ="
  - MVS System Trace Table
    - Eyecatcher: See the "PSW - - - - - Address -" column of the failing trace entry.
  - MVS TCB/RB flow
    - Eyecatcher: "OPSW"

Registers at Abend Description

- Registers can be useful while examining the code near the failure to determine the register that is invalid or in error.
  - Example: In the event of an ABEND0C1 with PSW = 00000004, if register 14 points just after a BALR (05EF) instruction, where R15 = 00000000, then, most likely, register 15 is in error. The load source of R15 must then be determined.
- In performing system abend analysis, another module might have passed the register in error.
  - You might be able to determine this by looking at the registers on entry to the failing module. If the same value is in one of the registers, then that value might have been passed.
- Also see the save area pages in the IMS Structure presentation to help understand normal save area conventions. This could also shed light on the problem.

Registers at Abend Sources

- Registers at Abend Sources
  - IMS Diagnostic Area
    - Eyecatcher: "ERROR REGISTERS"
  - IMS Type 6705 / Type 67FD / Type 67FF Log records
    - Eyecatcher: "SDWA"
      - Add x'26' to the beginning of the SDWA eyecatcher to reach REG 0.
  - IPCS STATUS
    - Eyecatcher: "Registers 0-7" "Registers 8-15"
  - IPCS SUMDUMP
    - Eyecatcher: "GPRs at time of error"

Registers at Abend Sources, Continued

- Registers at abend sources, continued
  - MSGBPE0006I
    - Eyecatcher: "TR00-03" "R04-07" "R08-11" "R12-15"
  - MSGDFS629I
    - Eyecatcher: "R0-3" "R4-7" "R8-11" "12-15"
  - MVS LOGREC
    - Eyecatcher: "Registers 0-7" "Registers 8-15"
  - MVS RTM2WA Summary
    - Eyecatcher: "GPRs at time of error"

Registers at Abend Sources, Continued

- Registers at abend sources, continued
  - OS/390 SDWA
    - Eyecatcher: "GPRs at time of error"
  - OS/390 System Trace Table (Regs 15, 0 and 1)
    - Eyecatcher: See the "UNIQUE-1 UNIQUE-2 UNIQUE-3" columns of the failing trace entry.
  - OS/390 TCB / RB Flow
    - Eyecatcher: For TCB "GENERAL PURPOSE REGISTER VALUES"
    - Eyecatcher: For RBs "GPR0-3" "GPR4-7" "GPR8-11" "GPR12-15"

Save Area Flow Description

- The save area flow can be analyzed to determine what circumstances led up to the failure.
  - Keep in mind that save areas with the low order bit turned on in the RET field, at offset x'C' (this makes that field contain an odd value, i.e.. nnnnnnn1, 3, 5, 7, 9, B, D, F) indicates it is residual.
  - Also, x'FF' in the high order byte indicates a residual flow
    - Still done by at least one non-IBM vendor product
- See the "Save Area Set Structure" diagrams earlier in the IMS Structure rpresentation as a reference to format unformatted save areas

Save Area Flow Sources

- Save Area Flow Sources (Active SAP or REG13 or DWA CECB)
  - IMS Diagnostic Area
    - Eyecatcher: "***SAVE AREA SET***"
  - IMS DPST formatting (If IMS dependent region)
    - Eyecatcher: "***SAVE AREA SET***"
  - IMS DFSABSAV - Only ABENDU0113 or ABENDU0780
    - Eyecatcher: "DPST"
    - Use the Jobname in the DPST as a reference point to reference the unformatted save area flow.
  - IMS SAP / Save Area
    - Eyecatcher: "***SAVE AREA SET***"
    - The abending save area flow is often marked "A C T I V E" - but not uniquely

Savearea Flow Sources, Continued

- Savearea Flow Sources, continued
  - IMS Type 6705 / Type 67FD / Type 67FF Log records
    - Eyecatcher: "DPST"
    - Use the Jobname in the DPST as a reference point to reference the unformatted save area flow.
  - IPCS Browse Mode
    - Use the REG13 value from Registers at Abend to reference the unformatted save area set.
  - MVS Save Area Trace
    - Eyecatcher: "SAVE AREA TRACE"

WLIC at Abend Discussion

- With ABEND0CX abends, the PSW address at time of abend usually points to the next instruction to be executed. The WLIC is needed to determine the number of bytes (length) to back up in order to find the failing instruction
  - WLIC layout: nnllcccc
    - where nn = wait count. This portion only exists in formatted RBs.
      - ll = Instruction length.
      - cccc = Interrupt code.
- If ABEND0C4 or ABEND0C5 occurs and the INTC (interrupt code) associated with the PSW AT ENTRY TO ABEND contains x'0010', x'0011', x'0038', x'0039', x'003A', x'003B' , the PSW points directly to the instruction that failed and the WLIC is not to be used for failing instruction resolution.
- In the event of other program interrupt abends (ABEND0Cx) the length value can be used to determine the length of the instruction to subtract from the failing PSW should the instruction not be a branch related failure

WLIC at Abend Sources

- WLIC (Wait, Length, Interrupt Code) Field at Abend Sources
  - IPCS STATUS
    - Eyecatcher: "Instruction length:"
    - Eyecatcher: "Interrupt code:"
  - IPCS SUMDUMP
    - Eyecatcher: "EC PSW at time of error"
      - The third word is the WLIC
  - MVS LOGREC
    - Eyecatcher: "INSTRUCTION LENGTH:"
    - Eyecatcher: "INTERRUPT CODE:"

WLIC at Abend Sources, Continued

- WLIC at Abend Sources, Continued
  - MVS RTM2WA Summary
    - Eyecatcher: "EC PSW at time of error"
      - The third word is the WLIC
  - MVS SDWA
    - Eyecatcher: "AEC1"
  - MVS System Trace Table,
    - PGM Entry
      - Interrupt code only
  - MVS TCB / RC Flow
    - Eyecatcher: "WLIC"

How to Obtain Diagnostic Sources

OBTAINING IMS DFSABSAV

- **IMS DFSABSAV:**
  - **This DSECT maps the IMS diagnostic area DFSABSAV.**
  - **This area includes a copy of the dependent region SDWA (if available), the entire PST IPAGE, SAP, LCRE, XPST, AND CLLE AREAS.**
  - **This pseudo module is mapped by assembling macro IPST ABSAV=YES.**
  - **DFSABSAV review is useful primarily for IMS sympathy abends such as ABENDU0113 and ABU0780**
  - **It is found in local Subpool 230 storage in a control region dump by performing the following steps:**
    - **IPCS SUMMARY FORMAT JOBNAME (nnnnnnnn) for the Control Region**
      - **Scan for the DFSABSAV MVS CDE**
      - **Use the EPA from the above MVS CDE in IPCS BROWSE Mode**
    - **IPCS browse mode. Scan for DFSABSAV eyecatcher in control private**
      - **Scan for the unformatted DFSABSAV MVS CDE**
      - **Use the EPA from the above MVS CDE in IPCS BROWSE Mode**

OBTAINING THE IMS DIAGNOSTIC AREA

- **The IMS Diagnostic Area can be obtained via the two methods listed below:**
  - **The IPCS command: VERBX IMSDUMP 'jobname, FMTIMS(options)'**
    - **Where jobname is the name of the IMS control region**
    - **ALL or SYSTEM or SUMMARY can be used in the FMTIMS(options) field.**
  - **Within the IMS Dump Formatter Hi-LEVEL Option**
    - **ALL**
    - **SYSTEM**
    - **SUMMARY**

OBTAINING IMS DPST FORMATTING

- **IMS DPST Formatting can be obtained via the following:**
  - **IPCS Command: VERBX IMSDUMP 'jobname, FMTIMS( )' with the following FMTIMS options:**
    - **ALL**
    - **DB,MIN**
    - **DPST,A=aaaaaaaa where aaaaaaaa = the DPST address**
    - **DPST,jobname where jobname is the region job name**
    - **DPST,N=bbbb where bbbb = the DPST region number**
  - **IMS IDF HI-LEVEL:**
    - **ALL**
    - **DB**
  - **IMS IDF LOW-LEVEL:**
    - **DPST ADDRESS**
    - **DPST NUMBER**
    - **DPST NAME**

OBTAINING IMS MASTER TERMINAL INFO

- **Some IMS Monitor packages will save all IMS messages to a dataset for later browsing. The IMS Master Terminal is often Spooled to a secondary master console. But the most obvious source would be to examine the Master Terminal screen.**

- **The IMS Secondary Master Terminal dataset can also be browsed using ISPF BROWSE mode**

OBTAINING THE IMS SAP / SAVE AREA

- **IMS SAP / SAVE AREAs can be obtained in a variety of manners:**
  - **IPCS Command: VERBX IMSDUMP 'jobname,FMTIMS( )' with the following FMTIMS options:**
    - **ALL**
    - **SYSTEM**
    - **SAVEAREA**
    - **SAP,A=aaaaaaaa where aaaaaaaa = the SAP address**
    - **SAP,E=eeeeeeee where eeeeeeee = the ECB address**
    - **DPST,A=aaaaaaaa where aaaaaaaa = the DPST address**
    - **DPST,jobname where jobname is the name of the region job name**
    - **DPST,N=bbbb where bbbb is the region number**

OBTAINING THE IMS SAP / SAVE AREA (continued)

- **IMS Interactive Dump Formatter HI-LEVEL:**
  - **ALL**
  - **SUMMARY**
  - **SAVEAREA**
- **IMS Interactive Dump Formatter LOW-LEVEL:**
  - **SAP ADDRESS**
  - **SAP ECB ADDRESS**

OBTAINING IMS TYPE 67 LOG RECORDS

- IMS Type 6705: Diagnostic information for dependent regions when a thread terminates abnormally
- IMS Type 67FD: A SNAP call was issued
- IMS Type 67FF: Abend occurred and SNAP dump was produced
- All of the above record types can be obtained via the following type of JCL and control statement stream run against the IMS Log (or a copy) in use at the time of the error or incident.

```
//PRINT67   JOB    .....
//JOBLIB    DD    DSN=IMS61.RESLIB,DISP=SHR
//PRTPGM    EXEC  PGM=DFSERA10
//SYSUT1    DD    DSN=INPUT1.LOG1,DISP=SHR
//SYSPRINT  DD    SYSOUT=*
//SYSUDUMP  DD    SYSOUT=*
//SYSIN     DD    *
CONTROL  CNTL H=EOF
OPTION    PRINT O=5,V=6705,L=2,C=E,E=DFSERA30
OPTION    PRINT O=5,V=67FD,L=2,C=E,E=DFSERA30
OPTION    PRINT O=5,V=67FF,L=2,C=E,E=DFSERA30
/*
```

OBTAINING INFORMATION USING GENERAL IPCS COMMANDS AND FUNCTIONS

- IPCS Browse Mode for examining addresses, control blocks, instructions etc.
  - IPCS Option 1 form the IPCS Primary Menu
- IPCS MTRACE
  - IPCS Command: VERBX MTRACE or:
  - IPCS Primary Menu Option 2 (Analysis) - then:
    - Option 6 (Component - MTRACE), or
    - Option 7 (Traces - and then Option 3 - MTRACE)
- IPCS STATUS
  - IPCS Primary Menu Option 2 (Analysis) - then:
    - Option 2 (STATUS)
- IPCS SUMDUMP
  - IPCS Command: VERBX SUMDUMP

OBTAINING IPCS INFORMATION (continued)

- **IPCS WORKSHEET**
  - **IPCS Command: STATUS or:**
  - **IPCS Primary Menu Option 2 (Analysis) - then:**
    - **Option 3 (Worksheet)**

OBTAINING Message BPE006I

- **Message BPE006I can be located from:**
  - **The CQS Job Log**
  - **IPCS MTRACE**
  - **The MVS Console listing**

OBTAINING MESSAGE DFS629I (continued)

- **Message DFS629I can be located from:**
  - **IMS Job Log**
  - **IPCS MTRACE**
  - **MVS Console listing**
  - **IMS Master Console Log**

OBTAINING MVS ASCBs

- **An MVS ASCB can be located from:**
  - **IPCS command: SUMMARY**
    - **PARMS: ASID (X'hh'), JOBNAME(nnnnnnnn) where hh is the ASID and nnnnnnnn is the name of the region in question.**
    - **This provides a summarized version of the error ASCB.**
  - **IPCS command: SUMMARY FORMAT**
    - **This command provides a completely formatted ASCB**
    - **Note: The performance of this command can be very poor for the IMS control region due to the excessive number of CDE entries being processed.**
  - **IPCS command: CBF aaaaaaaa STR(ASCB) where aaaaaaaa is the address of the ASCB.**

OBTAINING AN MVS FORMATTED DUMP

- An MVS formatted dump can be obtained via::
  - IPCS command: SUMMARY FORMAT
    - This provides a more complete control block formatting but it may not be effective for IMS control region formatting due to poor performance of the command while processing the many IMS CDE entries.
    - The parms are: ASID (X'hh'), JOBNAME(nnnnnnn)
  - IPCS command: SUMMARY
    - This command provides a summary of the major OS/390 control blocks. It offers better performance at the cost of producing less data.
    - The parms are: ASID(X'hh'), JOBNAME(nnnnnn)

OBTAINING MVS LOGREC

- **MVS LOGREC can be obtained via:**
  - **IPCS command: VERBX LOGDATA**
  - **The MVS EREP Utility**
    - **See MVS Diagnosis: Tools and Service Aids Manual**

OBTAINING MVS RTM2WA SUMMARY

- **MVS RTM2WA SUMMARY information can be obtained via:**
  - **IPCS command: SUMMARY FORMAT**
    - **The parms are: ASID (X'hh'), JOBNAME(nnnnnnnn)**
  - **IPCS command: VERBX SUMDUMP**
  - **MVS formatted section of a preformatted dump. The dump could be a SYSABEND or SYSUDUMP**

OBTAINING MVS Savearea Trace

- **MVS Savearea Trace information can be obtained via:**
  - **MVS formatted section of a preformatted dump.**
    - **The dump could be a SYSABEND or SYSUDUMP**
    - **Not available in machine readable dump**
      - **SVC DUMP or SYSMDUMP**

OBTAINING MVS SDWA

- **MVS SDWA can be obtained via:**
  - **IMS DFSABSAV control block**
    - **Eyecatcher: "SDWA"**
    - **This is only valid for IMS Abend codes U0113 or U0780**
    - **The SDWA is unformatted here. Use the MVS Data Areas manual for field offsets.**
  - **IMS TYPE 67 Log Records**
    - **The SDWA is unformatted here. Use the MVS Data Areas manual for field offsets.**

OBTAINING MVS SDWA (continued)

- **MVS SDWA can be obtained via:**
  - **IPCS command: SUMMARY FORMAT**
    - **Use parms: ASID(X'hh'),JOBNAME(nnnnnnnn)**
    - **Locate the pointer in Register 1 in the Request Block (RB) <u>following</u> the RB containing SYNCH macro WLIC value of X'0002000C'**
      - **Use the IPCS command: CBF aaaaaaaa STR(SDWA) - where aaaaaaaa = REG01 from above**
  - **The IPCS WORKSHEET**
    - **Look for eyecatcher: SDWA where the address can be obtained.**
    - **IPCS command: CBF aaaaaaaa STR(SDWA) using the obtained SDWA address.**

OBTAINING THE MVS SYSTEM TRACE TABLE

- **MVS SYSTEM TRACE TABLE can be obtained via:**
  - **IPCS command: SYSTRACE**
    - **The parms used are: ASID(x'hh'),JOBNAME(nnnnnnnn)**
    - **Where hh = The failing ASID in hex and nnnnnnnn = the failing jobname**
  - **Examining the MVS formatted section of a preformatted dump such as SYSABEND or SYSUDUMP**
    - **Eyecatcher: "System Trace Table"**

OBTAINING MVS TCB / RB FLOW

- **MVS TCB / RB Flow can be obtained via:**
  - **IPCS command: CBF aaaaaaaa STR(TCB)**
    - **IPCS command: CBF aaaaaaaa STR(RB) for chained RBs**
      - **TCBRBP field points to last RB in chain**
      - **RB LINK field points to the prior RB in the chain or TCB if the top RB**
  - **IPCS command: SUMMARY FORMAT ASID(x'....')**
    - **This provides a more complete TCB / RB control block formatting including the CDEs. It may not be effective for IMS control region formatting due to poor performance of the command while processing the many IMS CDE entries.**
  - **IPCS command: SUMMARY REGS**
    - **This command provides a summary of major OS/390 control blocks. It is a better performer as compared to the SUMMARY FORMAT ASID(x'....') but produces less data. Also the RB flow is presented in LIFO sequence with this form of the command.**
  - **Examining the MVS formatted section of a preformatted dump such as SYSABEND or SYSUDUMP.**

RETURN TO INDEX