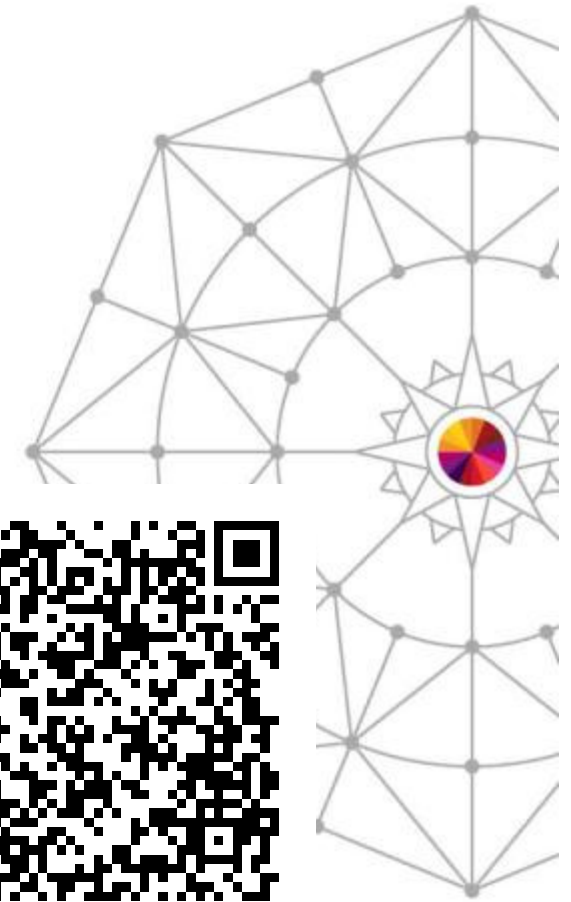




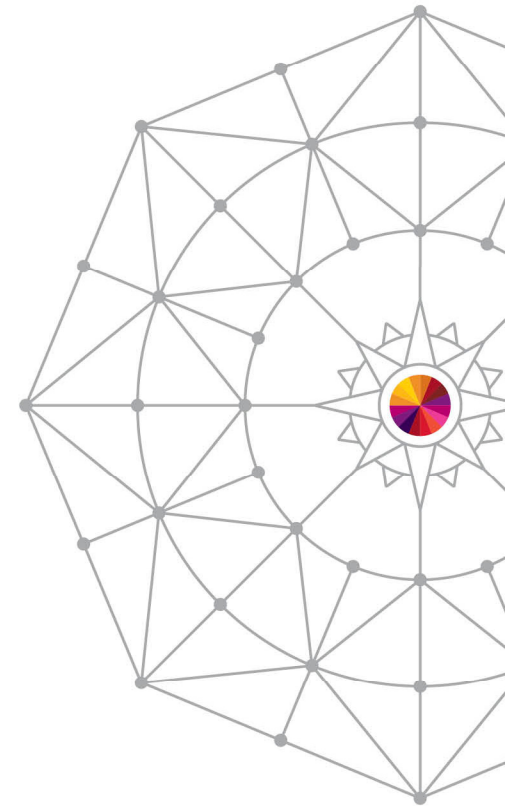
# DFSMS Advanced: PDSE Diagnostics and Recovery

Speaker: Thomas Reed  
IBM Corporation  
Session: **15084**



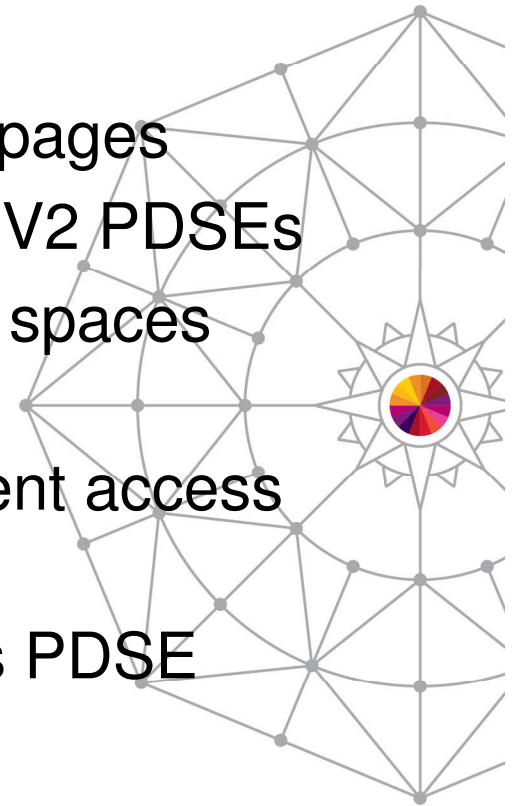
# Agenda

- How PDSE externalizes problems.
- PDSE's had a problem, now what?
- Identifying and solving the problem:
  - Tools
  - OF4's
  - Hangs
  - Corruption



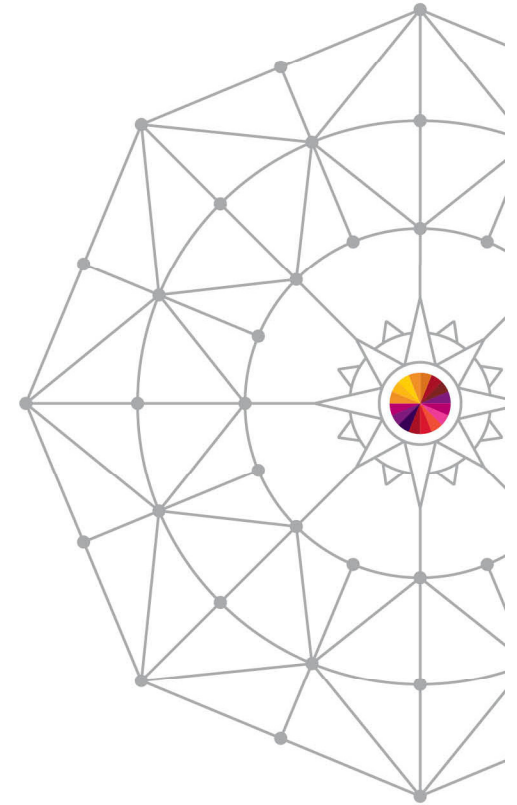
## What is a PDSE?

- PDSE: Partitioned DataSet Extended
- A PDSE is a collection of directory and data pages
- At V2R1 there are 2 dataset formats V1 and V2 PDSEs
- PDSE server consists of one or two address spaces (SMSPDSE and SMSPDSE1)
- The SMSPDSE(1) address spaces serve client access requests for PDSE datasets
- Under the hood SMSPDSE(1) also manages PDSE serialization and buffering



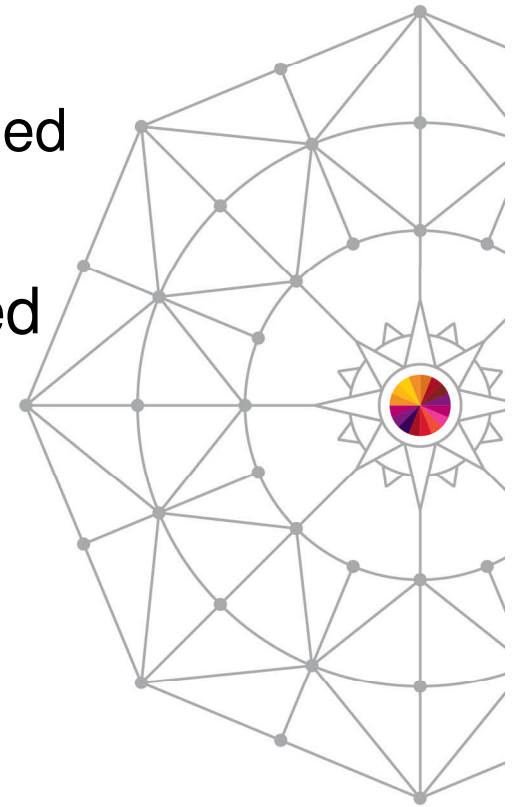
# How PDSE Externalizes Problems

- ABENDS
  - 0F4
  - 0Cx
  - 878
- IGW messages
  - 38A/31I
  - 702I
  - 007E/077E
- CSV031I



## 0F4 Abends

- Indicate a PDSE “Logic Error”
  - In other words something unexpected happened
- RC (Reg 15) is not very useful
- RSN (Reg 0) tells us where the error occurred and what the error was
  - Errors can percolate up the calling chain
  - Many error codes may be issued from the same place in code



## Reading the 0F4 Dump Title

- CompID
  - DF115 = PDSE
  - DF104 = FAMS
- CSECT + offset
- PTF number
- ABEND code
- Return code
- Reason code

```
COMPID=DF115,  
CSECT=IGWDAV00+10C4,  
MAINT ID=UA69488,  
ABND=0F4,  
RC=00000024,  
RSN=01188022
```

# Checking the Error Information

- Check Reg 0 for the reason Code
- Check Reg 15 for the return code

## Time of Error Information

PSW: 07541000 80000000 00000000 08251698  
 Instruction length: 02 Interrupt code: 000D  
 Failing instruction text: 58F05048 0A0DB219 0000A788

Breaking event address: 00000000\_00000000

Registers 0-7

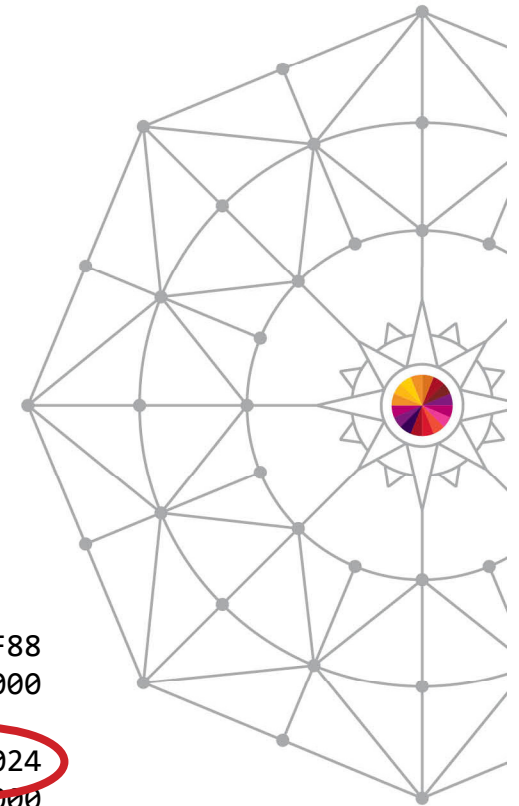
GR: 01188022 040F4000 00FD9A40 7C343268 00007703 7C348400 7C343060 00FDFF88

AR: 003FF6F8 00000000 00000001 00000000 00000002 00000000 00000000 00000000

Registers 8-15

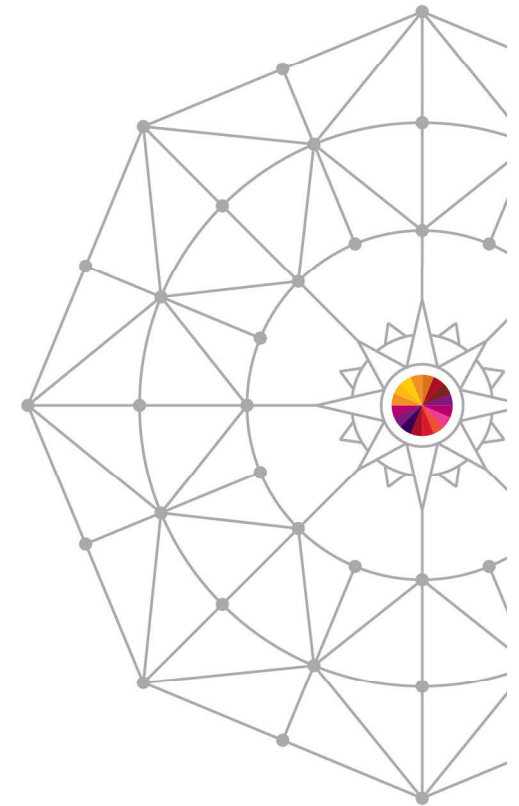
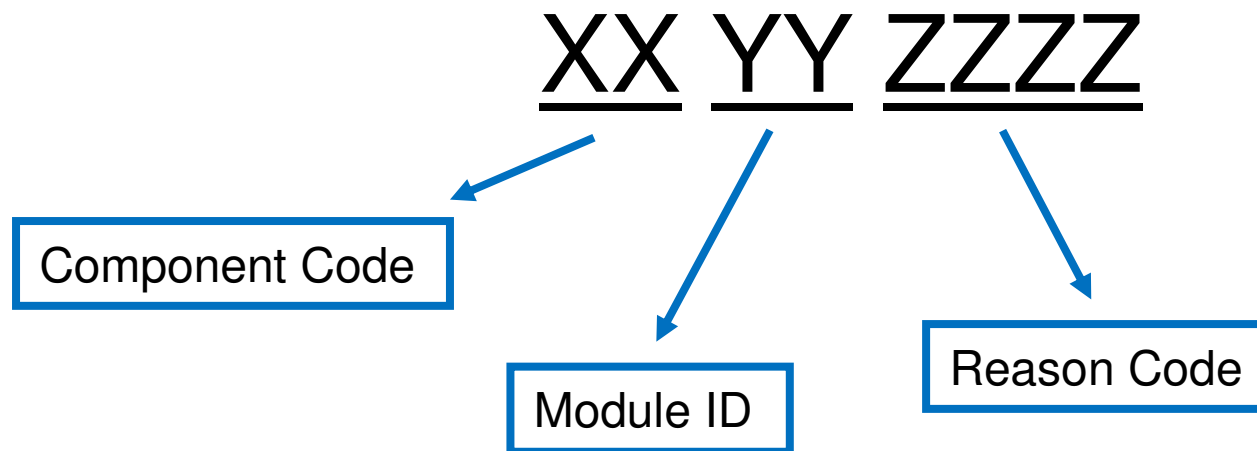
GR: 00FDFFF8 7C343268 7C348868 00000000 0825101A 7C348778 8825168A 00000024

AR: 00000000 00000001 00000000 00000000 00000000 00000000 00000000 00000000



# Reason Code Translation

Reason Code Format:





# Reason Code Translation Example

RSN: 150BC008

This error generally the result PDSE corruption due to being shared outside the SYSPLEX

**How it's broken down:**

**15** = SubComp = PDSE BMF IO Control

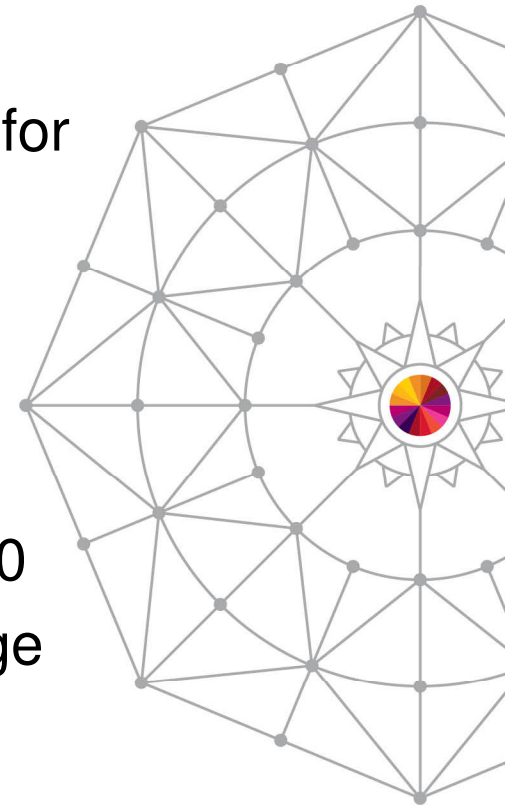
**0B** = Module ID = IGWBIEX1

**C008** = Reason = RSNS\_IO\_ERROR\_DIRECTORY



# IGW Messages and Meanings

- IGW038A and IGW031I
  - Indicates a PDSE latch or lock has been held for longer than the PDSE Monitor's set duration
- IGW702I
  - IEBPDSE (The PDSE Validation Tool) completion message
  - Indicates that IEBPDSE did not complete RC 0
  - IGW701I is the successful completion message





## IGW Messages and Meanings Cont'd

- IGW007E and IGW077E
  - SMSPDSE and SMSPDSE1 address space failures respectively
  - IGW007E will require an IPL to recover
  - IGW077E may be recoverable via the PDSE1,ACTIVATE command

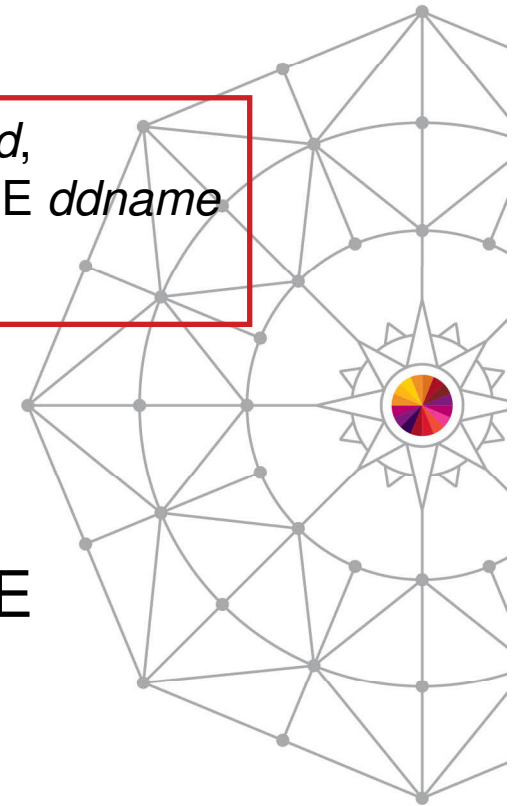
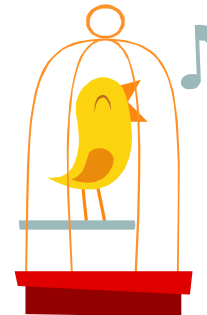


# IGW Messages and Meanings Cont'd

## CSV031I

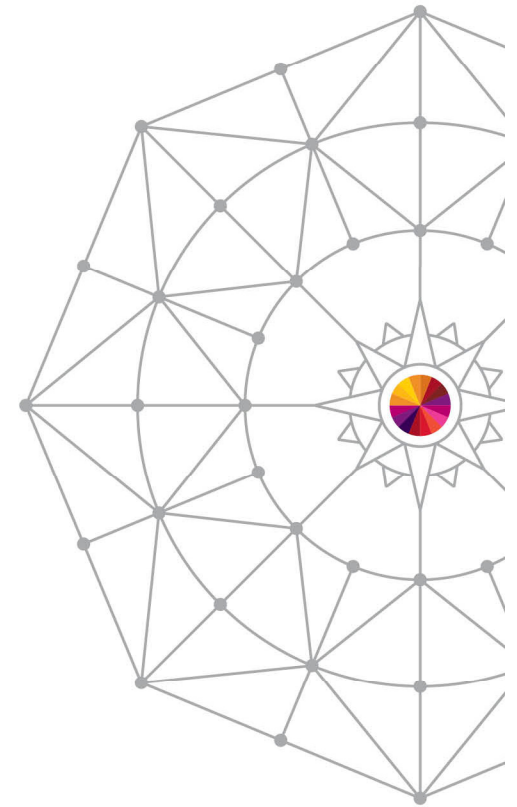
LIBRARY {SEARCH | ACCESS} FAILED FOR MODULE *mod*,  
RETURN CODE *xx*, REASON CODE *reason-code*, DDNAME *ddname*

- Not a PDSE message
- RC and RSN code are percolated from PDSE
- A “canary” for PDSE corruption

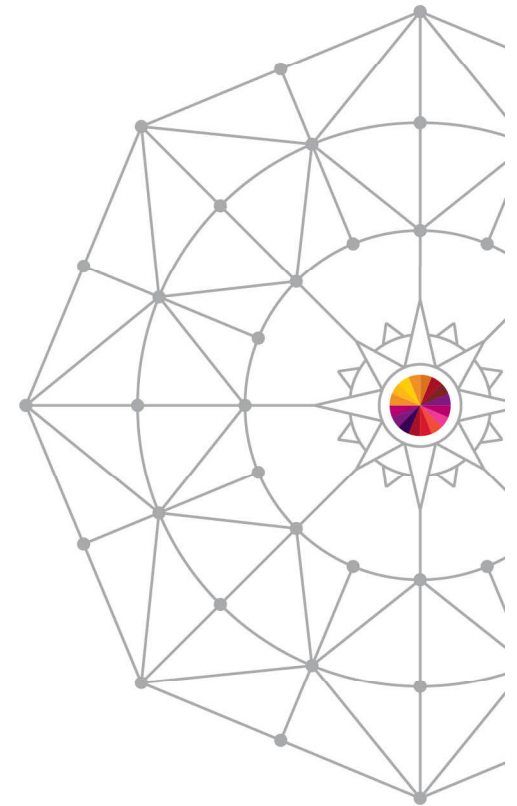


# OCx ABENDs

- 0C4 ABENDs
  - Generally due to a bad pointer
  - Variety of causes
  - Expect an APAR
- 0C1 ABENDs
  - Very rare
  - Some 0C1's are diagnostic branches to a 0h



**So, PDSE has just had a problem...**

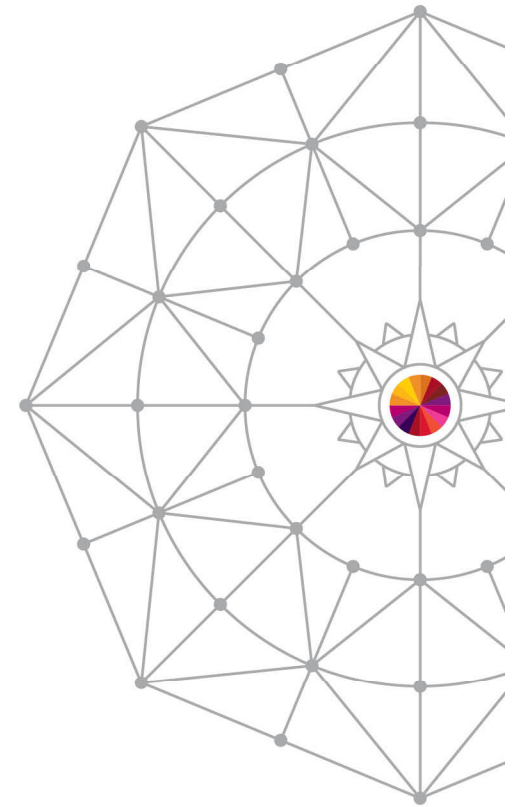


**DON'T  
PANIC**



# Step 1: Identify the Problem

- Start narrow by defining the symptoms
- What are we experiencing?
  - A 0F4 dump?
  - Many 0F4 dumps?
  - IGW messages?
  - Processing hangs?
  - Dataset corruption?
  - CPU/Storage consumption?



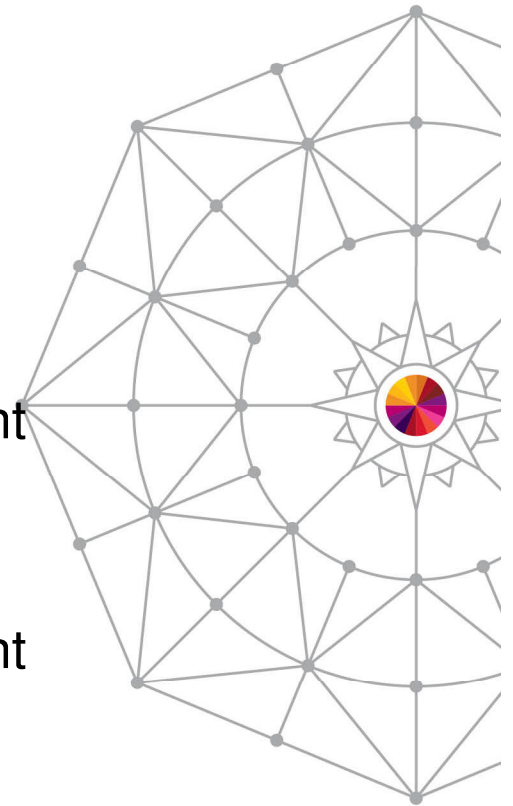
## Step 2: Gather Doc!

- Initial PDSE doc gathering comes down to 4 items
  1. An SVCDump including the PDSE[1] address space
  2. SYSLOG/OPERLOG
  3. LOGREC/RAW EREP
  4. DSS physical dump of the affected dataset



## Step 2 Cont'd

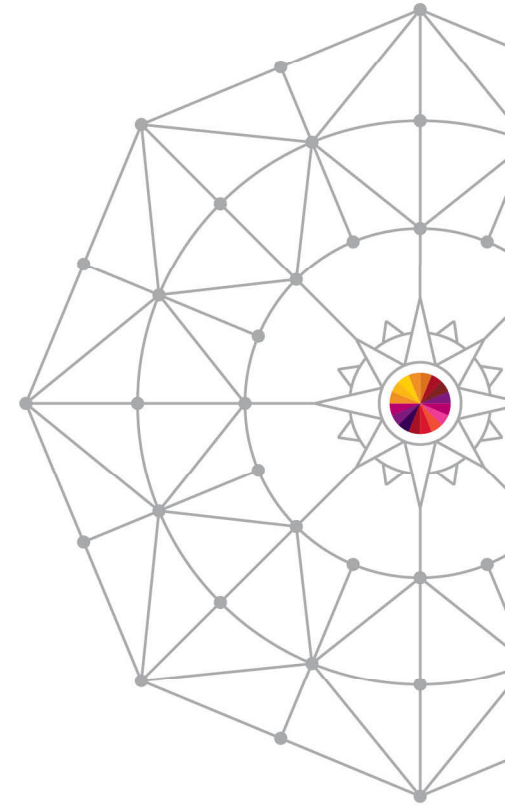
- For dumps, gather:
  - The SVC Dump (for multiple, start with the first)
  - SYSLOG and LOGREC covering the incident
- For hangs, gather:
  - A console dump of the PDSE[1] address space
  - SYSLOG and LOGREC starting prior to the incident
- For corruption, gather:
  - The SVC Dump (for multiple, start with the first)
  - SYSLOG and LOGREC starting prior to the incident
  - A DSS Physical dump of the dataset



\*See APPENDIX for JCL and Parameters

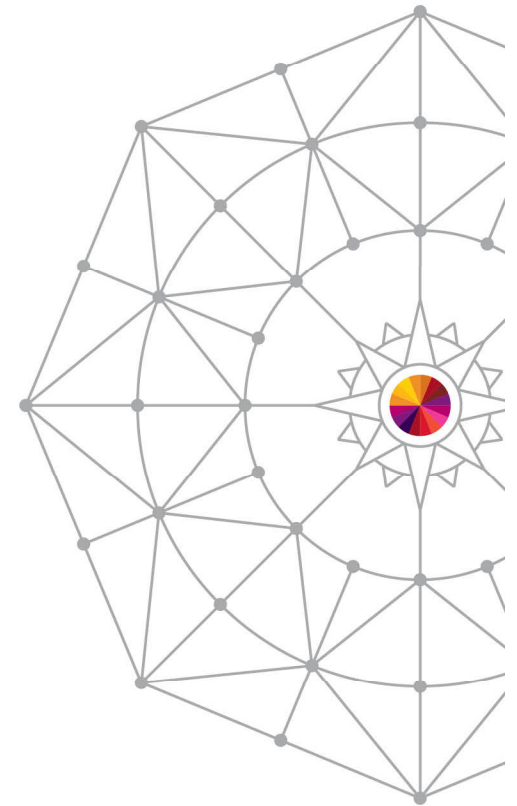
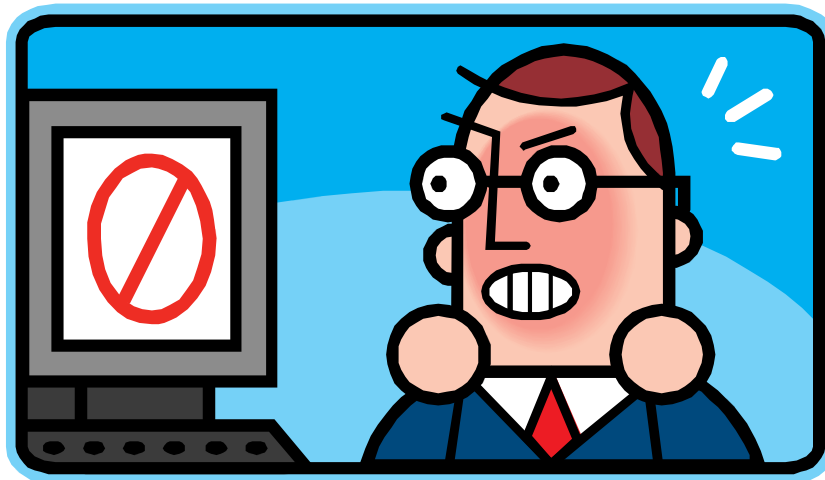
## Step 2 Cont'd

- DO NOT SUPPRESS 0F4 Dumps
- DO NOT SUPPRESS IGW038A messages

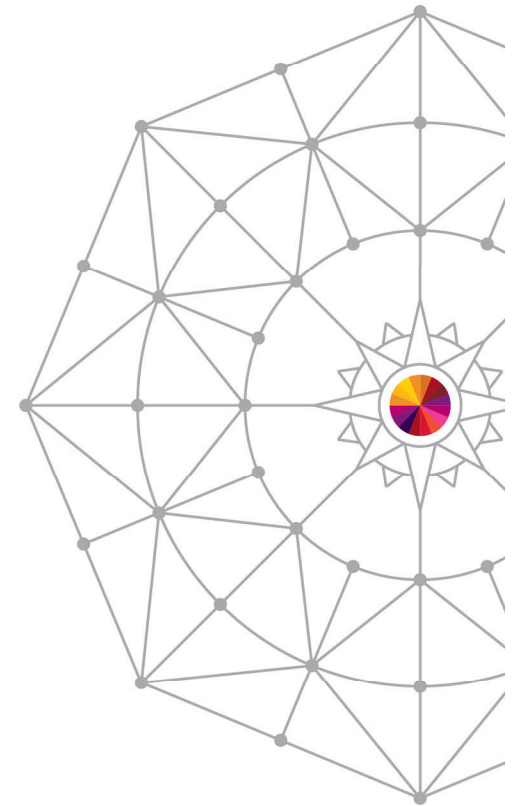


## Step 3: Call L2

- Shameless L2 plug!

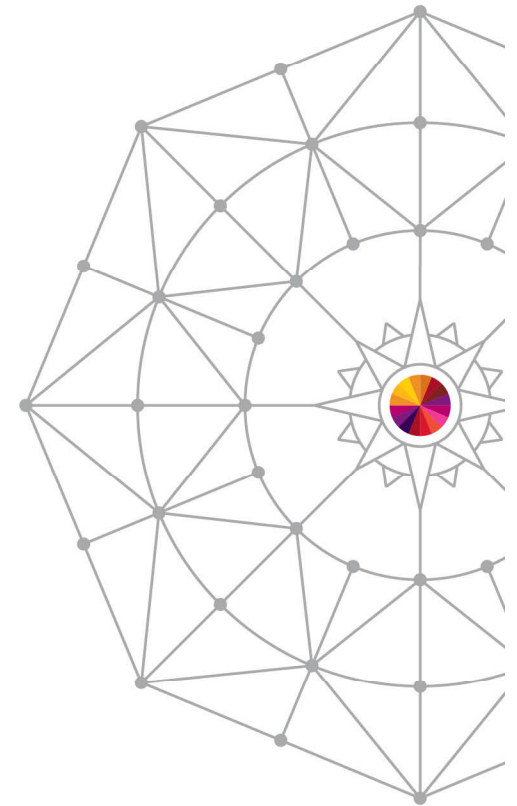


# Solving the Problem



# Tools of the Trade

- IPCS
  - IP ST (IP STATUS)
  - SMSXDATA
    - Q
    - AS
    - MSGS
    - POOLS
- IEBPDSE – PDSE Validation Tool



# IP STATUS

- Gives a quick overview of the error
- For PDSE 0F4s the most important are the registers
  - Return Code in REG 15
  - Reason Code in REG 0

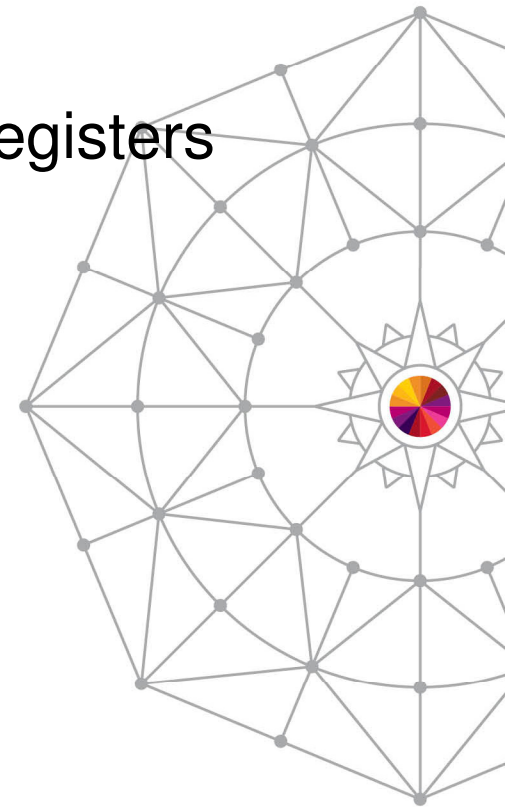
```

Time of Error Information

PSW: 075C1000 8BF686F0   Instruction length: 02   Interrupt code: 0000
Failing instruction text: 58F05048 0A0DB219 0000A788

Breaking event address: 00000000_00000000
AR/GR 0-1   008FE050/00000000_13F2FE1E   00000000/00000000_040F4000
AR/GR 2-3   00000000/00000000_00FDD098   00000000/00000000_7B923268
AR/GR 4-5   00000000/000000E4_00007703   00000000/00000000_7B925D18
AR/GR 6-7   00000000/00000000_7B923060   00000000/00000000_00FDC0D8
AR/GR 8-9   00000000/00000000_00FDC148   00000000/00000000_7B923268
AR/GR 10-11 00000000/00000000_7B9262D0   00000000/00000000_00000000
AR/GR 12-13 00000000/00000000_0BF68072   00000000/00000000_7B9261E0
AR/GR 14-15 00000000/00000000_8BF686E2   00000000/00000000_00000024

Home ASID: 0009   Primary ASID: 0009   Secondary ASID: 0009
PKM: 8040       AX: 0001       EAX: 0000
  
```

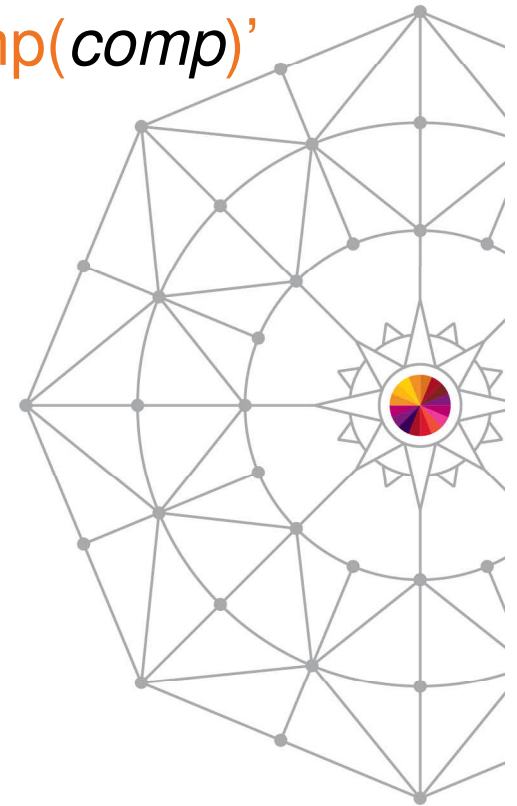




# SMSXDATA

VERBX SMSXDATA 'f(*func*) jobname(*job*) comp(*comp*)'

- 'comp' stands for component within PDSE
  - Many components with reports
  - CLM contains the most important
- 'f' stands for function
  - Q = Quick view of error TCB
  - AS = Address spaces summary

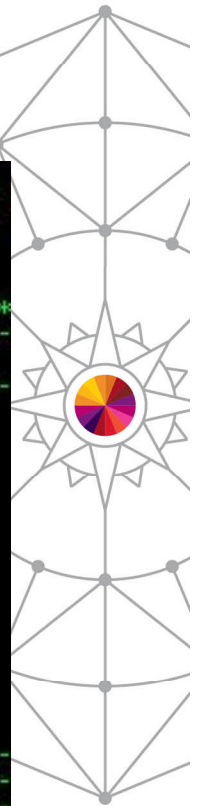


# SMSXDATA

VERBX SMSXDATA 'f(Q) jobname(job) comp(CLM)'

```
DFSMS verbexit processing
Title:COMPID=DF115,CSECT=IGWLHA10+244E,DATE=02/20/12,MAINTID=UA64274 ,ABND=0F4,RC=00000024,RSN=13F2FE1E
DSN:
Function=Q
Dump compatible with IGWFPAN version:HDZ1C10
-----
Primary Asid:0009 Secondary Asid:0009 PSATOLD:008FAE88
-----
Looking for SSF thread by Register 13:7B9261E0
FVSA:7B923040 Asid:0009 SMSPDSE1
FVSP:7B926190 IGWFPCPER <<<<<input FVSP
FVSP:7B924EE0 IGWLHA10
FVSP:7B924438 IGWLHA00
Displaying Recovery FVSA associated sequences

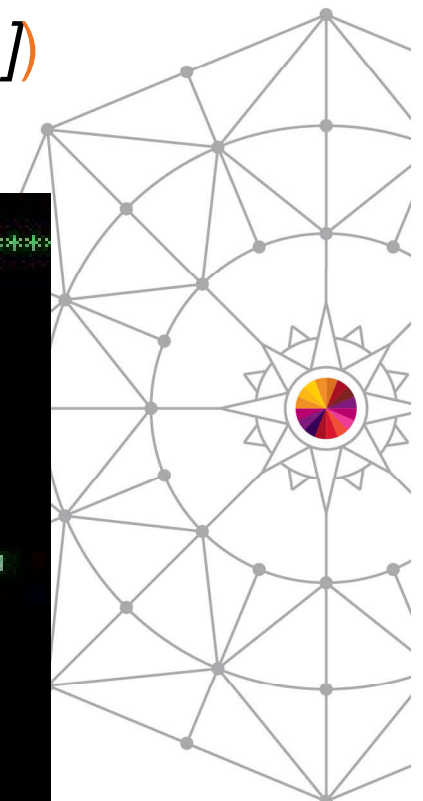
FVSA:73C26040 Asid:0009 SMSPDSE1
FVSP:73C284B8 IGWFDDMP
FVSP:73C27A80 IGWFCLRR
FVSP:73C27828 IGWFERTR
FVSP:73C27438 IGWFEFRR
Looking for SSF thread by Control Reg15:7F6E3388 Linkage Stack
-----
Tcbs Description-----
IGWFPCPER:IGWLHA10:IGWLHA00 -----
Tcb: 0
```



# SMSXDATA

VERBX SMSXDATA 'f(AS) jobname(SMSPDSE[1])  
comp(CLM)'

```
Latch Summary
*****
  Latch Name          Holder                Waiter(s)
1.  IGWASRB(7FF93380)+18 (81)0032:008BD4E0 (76)0032:0089D030
   (84)0032:008BD8E8
   (82)0032:008BD7E8
   (77)0032:0089D1C8
   (79)0032:008AE380
   (78)0032:008AE0F0
2.  IGWHTAB(7FF9F000)+38 (41)0009:008F0900 (16)0009:008F9E00
   (13)0009:008FF230
   (595)0200:008FF260
3.  IGWHTAB(7FF9F000)+48 (41)0009:008F0900 (14)0009:008FA1D0
   (81)0032:008BD4E0
   (34)0009:008F6CF0
   (33)0009:008F6E88
   (32)0009:008F7190
   (61)0009:008EA3B0
   (26)0009:008F8360
4.  IGWHTAB(7FF9F000)+58 (41)0009:008F0900 (1530)028E:008CA7B8
   (1242)0257:008FF260
5.  IGWHTAB(7FF9F000)+60 (41)0009:008F0900 (1295)0270:008FF260
6.  IGWHTAB(7FF9F000)+70 (41)0009:008F0900 (1243)0259:008CA488
*****
```



# SMSXDATA

VERBX SMSXDATA 'f(*POOLS*) jobname(*SMSPDSE[1]*)'

- Max to the bottom for the important parts

```

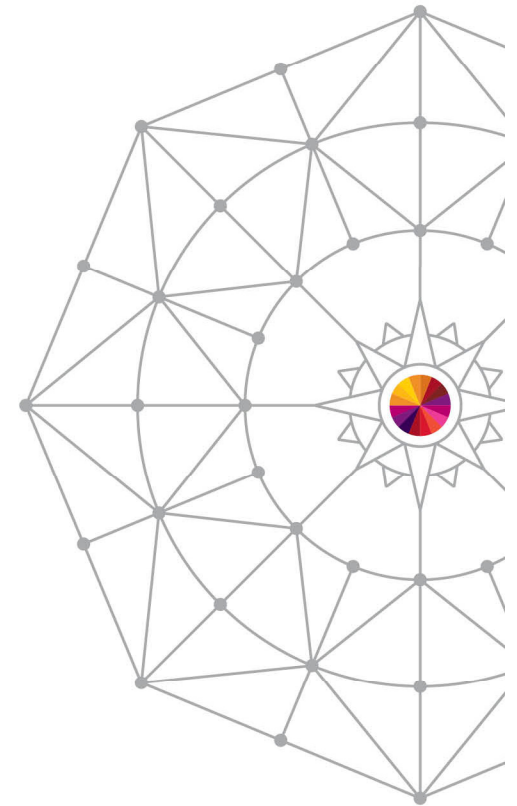
*****
*   Above the Bar Storage
*   _Extents  ___Total_Size_of_the_Pool  _TotCells  InUseCells  Fixed  FProt  Key  Description-----
*****
1.      45      47,185,920      245745      186334      NO      YES  50  IGWHL3B STORAGE POOL
2.       4       4,194,304       20164       4412       NO      YES  50  IGWAJB STORAGE POOL
3.      44      46,137,344      262108      197644      NO      YES  50  IGWLRE STORAGE POOL
4.       1       4,194,304         100        112       NO      YES  50  IGWHL1B Global Lock State trace
5.      33      2,214,592,512      540672       521       NO      YES  50  DFP BMF PAGE BUFFER POOL
6.      24      25,165,824      524280       521       NO      YES  50  DFP BMF BCB POOL
7.      41      42,551,616      288673      197823      NO      YES  50  JCDCM DSC POOL FOR PUB
8.      59      61,865,984      38645        198       NO      NO   50  JCDCM DSC POOL FOR FIB
9.     326      341,835,776      213530      186129      NO      NO   50  JCDCM DSC POOL FOR FIB DREFD
*****
Total Space Used: 2,788,163,584

```

- TotCells = Allocated storage in cells
- InUseCells = In use storage in cells

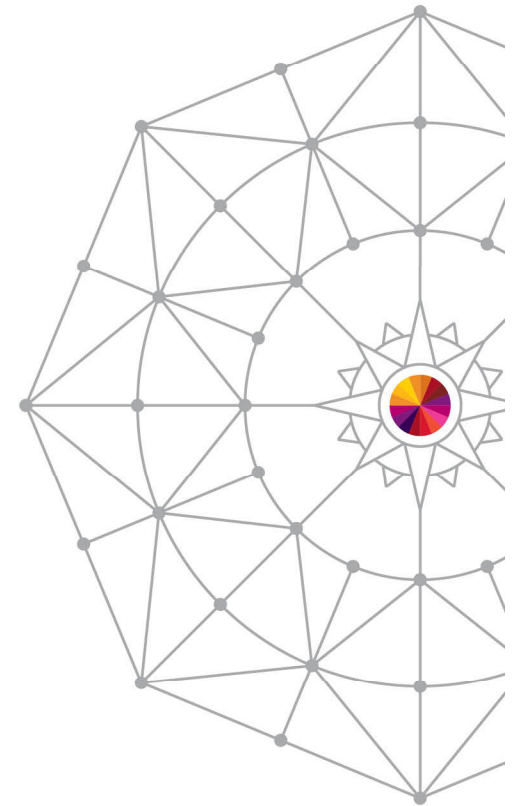
# IEBPDSE

- The PDSE validation tool
    - 1.13 and above
    - Use the highest available version if possible
    - 1.13 and 2.1 versions are NOT identical
- Do not expect identical results**
- Works on both V1 and V2 datasets



# 0F4 ABENDs

## PDSE Logic Errors

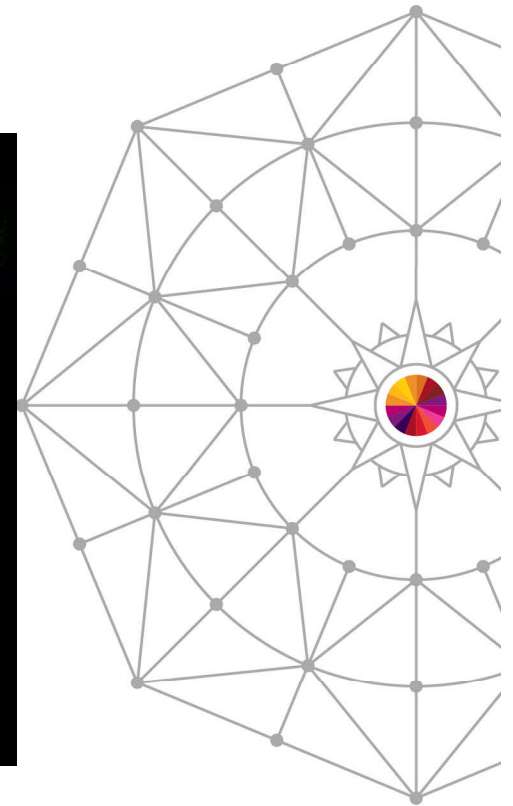




# Step 1: Getting the reason code

- IP ST [REGS]

```
CPU STATUS:  
PSW=07540000 80000000 00000000 06ACAAFA  
  (Running in PRIMARY, key 5, AMODE 31, DAT ON, SUPERVISOR STATE)  
  Disabled for PER  
  ASID(X'0009') 06ACAAFA. IGWDLAC0+BAFA IN EXTENDED PLPA  
  ASCB68 at FC0780, JOB(GUDB2035), for the home ASID  
  ASXB68 at BFD860 and TCBE8E at B99E88 for the home ASID  
  HOME ASID: 0044 PRIMARY ASID: 0009 SECONDARY ASID: 0044  
  
General purpose register values  
  0-1 00000000_150D5476 00000000_040F4000  
  2-3 00000000_150D5476 00000000_00000020  
  4-5 00000000_00000080 00000000_00002DFC  
  6-7 00000000_00000000 00000000_00000000  
  8-9 00000000_00FE0CF0 00000000_06ACC968  
 10-11 00000048_00FE0C80 00000000_06ACB26F  
 12-13 00000000_06ACA270 00000000_7EB510B8  
 14-15 00000000_86ACAAFA 00000000_00000020
```



- Try and identify the reason code
  - In this case: **JCDM\_NO\_LSSM\_RECORD**

## Step 2: Get the calling sequence

```

-----
Primary Asid:0009 Secondary Asid:0044 PSATOLD:00B99E88
-----
Looking for SSF thread by Register 13:7EB510B8
FVSA:7F180040 Asid:0009 SMSPDSE1
FVSP:7EB51068 IGWDLBAP <<<<<input FVSP
FVSP:7F187CD8 IGWDLRUP
FVSP:7F186820 IGWDLDES
  CDM LAC DESTROY_LS
    FDES:7F185BD8 DUB:7EA107F0
      FileToken:01-S28000-000104-00000000000003-00000000AA6E-0000
FVSP:7F185260 IGWDDDSF
  IGWDDIP1:IGWDDDSF
  CDM Destroy File:
    FDSF:7F184178 DirConnToken:000000007EA107F0 Defer:1 DestroyAll:0 KEEP_PRIMARY_NAME:0
      NOTESTLOCK:1 CreatePendingDelete:0
      FDSF_NAME_ID:01-S28000-000104-00000000000003-00000000AA6E
      Dub:7EA107F0 Dib:7F384720 HLib:7FF6AB30 DSN: Sys1.samp.pdse
FVSP:7F1830F8 IGWDADPD
  Update version running
FVSP:7F182140 IGWDACND
  Update version running
  CDM Connect Directory:
    Fcmd:7F24D428 Intent:OUTPUT Ysgt:01-S28000-000104
FVSP:7F181A00 IGWDBPAR
  Version 2 of IGWDBIP1 running
FVSP:7F181438 IGWFARR3
Displaying Recovery FVSA associated sequences
  
```

Requested function

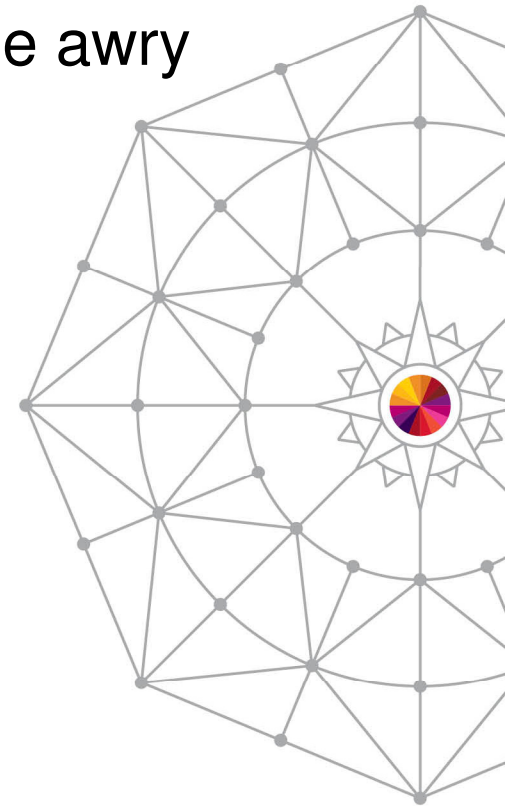
The dataset we're connecting to

Connecting for output



## Step 3: Now What!?

- Open a PMR, a 0F4 means something's gone awry and L2 should look at it
- Repeated 0F4's on access of the same dataset can indicate corruption



# Hangs

## PDSE Resource Serialization Issues

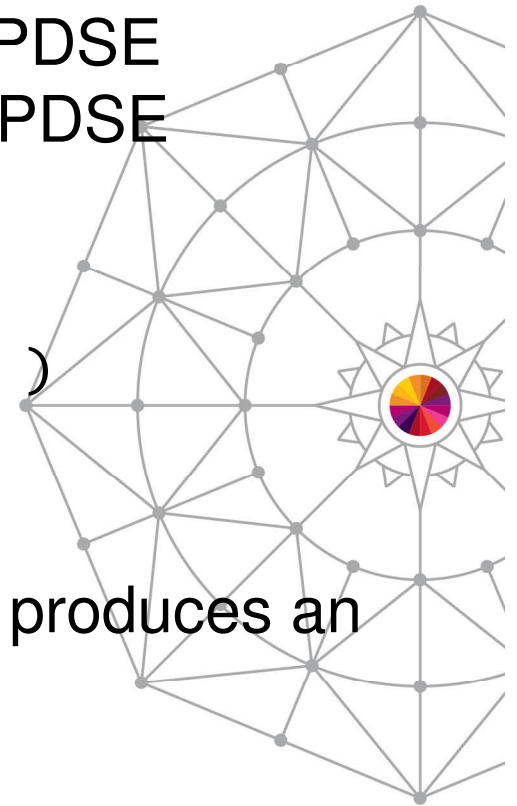


## Step 1: IGW038A Messages

- IGW038A messages indicate that there is a PDSE resource that has been held longer than the PDSE monitor's set duration time

IGW038A Possible PDSE Problem(s) (SMSPDSE )  
Recommend issuing V SMS,PDSE,ANALYSIS

- Issuing the suggested ANALYSIS command produces an IGW031I report



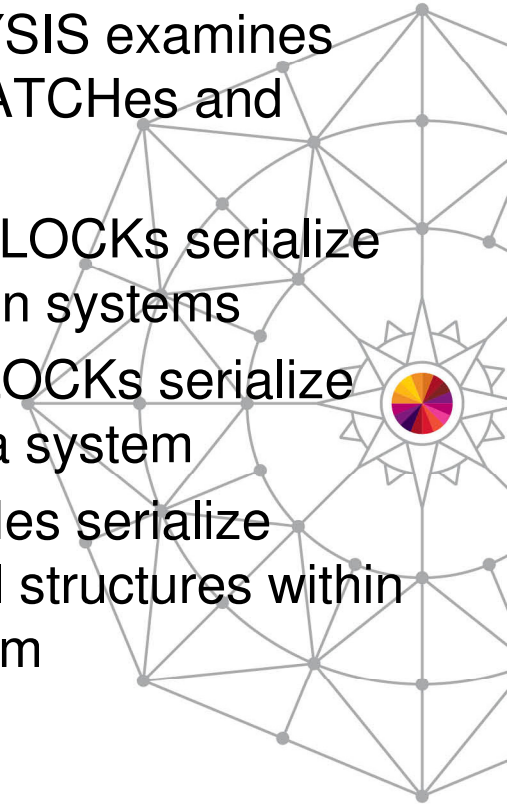
# Step 1: IGW031I Messages

```

IGW031I PDSE ANALYSIS  Start of Report(SMSPDSE )
++ Unable to latch ASRBULCH:000000007F04DE00
   Latch:000000007F04DE18 Ho1der(0008:009F6CF0)
   Holding Started Task:SMSPDSE
++ Unable to latch ASRBULCH:000000007F050560
   Latch:000000007F050578 Ho1der(0008:009F75D0)
   Holding Started Task:SMSPDSE
++ Unable to latch ASRBULCH:000000007F050640
   Latch:000000007F050658 Ho1der(0008:009F6AD0)
   Holding Started Task:SMSPDSE
++ Unable to latch ASRBULCH:000000007F0503A0
   Latch:000000007F0503B8 Ho1der(00E5:009AEE88)
   Holding Started Task:TREED12
-----data set name-----vsgt-----
SYS2.SOMETHING.LOADLIB          01-SYSQ01-000109
++ Unable to latch HL1b:000000007FF62E40
   Latch:000000007FF62E50 Ho1der(0008:009F6CF0) IGWLHPRG
   Holding Started Task:SMSPDSE
PDSE ANALYSIS  End of Report(SMSPDSE )

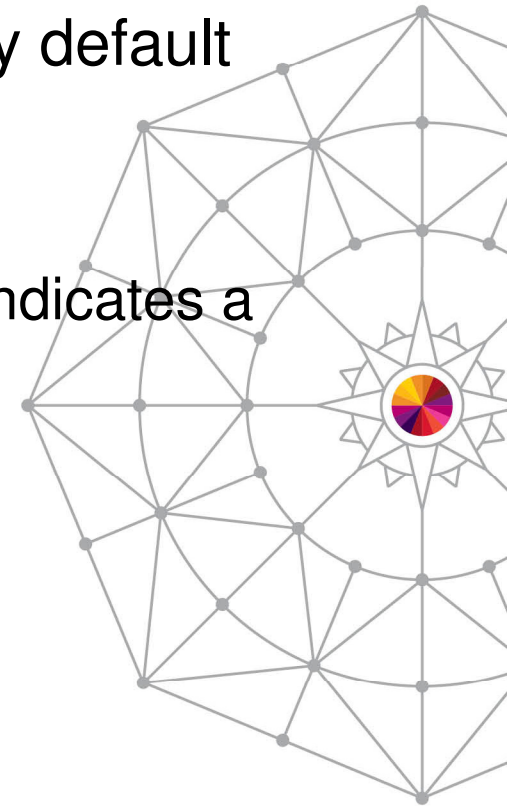
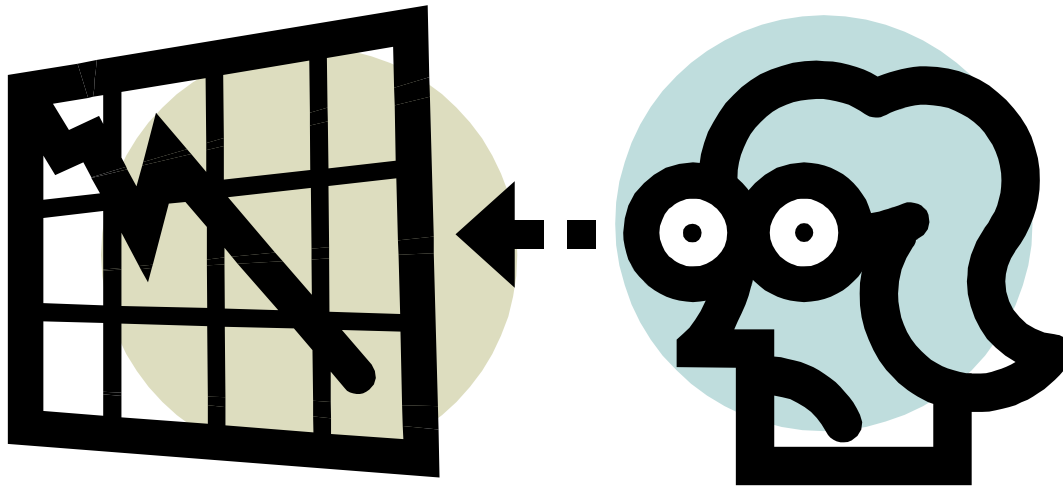
```

- ANALYSIS examines both LATCHes and LOCKs
- Global LOCKs serialize between systems
- Local LOCKs serialize within a system
- LATCHes serialize internal structures within a system



## Step 1: Reading the Trends

- The PDSE monitor runs every 90 seconds by default
  - A single IGW038A message is OK
  - Can be tripped in a busy system
  - Repeating messages every ~2min generally indicates a problem



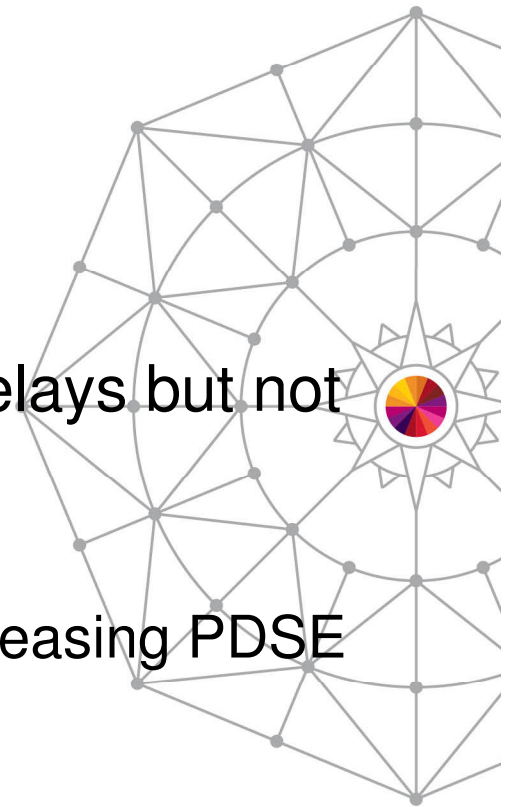
## Step 2: Get a Dump

- Using the **V SMS,PDSE[1],MONITOR,DUMPNEXT** makes getting a good dump easy
- Alternately taking a console dump of both PDSE address spaces works well
  - Make sure to get the involved PDSE ASID
- If there are unresponded messages to a system
  - Take a console dump from that system as well
  - Again, make sure to get the involved PDSE ASID



## Step 3: Finding the Root of the Hang

- Do we have a real hang?
  - Is it a single holder?
  - Are holders changing?
  - Are the IGW038A message ongoing?
- Changing holders can indicate processing delays but not necessarily a real hang
  - Check CPU utilization
  - CPU starvation can cause delays with jobs releasing PDSE resources



# Step 4: We have a real hang, now what? (it'd be boring if we didn't!)

- Finding the key latch
- IGW031I report is not what we need....
- AS report to the rescue!

```

*****
Latch Summary
*****
   Latch Name                Holder                Waiter(s)                Latch Caller(s)
1.  IGWASRB(7F04DE00)+18     (35)0008:009F6CF0      (34)0008:009F6E88
2.  IGWHL1B(7FF62E40)+10    (35)0008:009F6CF0      (402)00E5:009AEE88
*****
*****
Tcb Latch Summary
*****
   TCB                Held Latch(es)                AWaited Latch(es)
34.  0008:009F6E88                IGWASRB(7F04DE00)+18
35.  0008:009F6CF0                IGWASRB(7F04DE00)+18
35.  0008:009F6CF0                IGWASRB(7F04DE00)+18
                                         IGWHL1B(7FF62E40)+10
402.  00E5:009AEE88                IGWHL1B(7FF62E40)+10

```



# Step 5: Reading the AS Latch Report

- Track down the originating latch
- Note the TCB and ASID

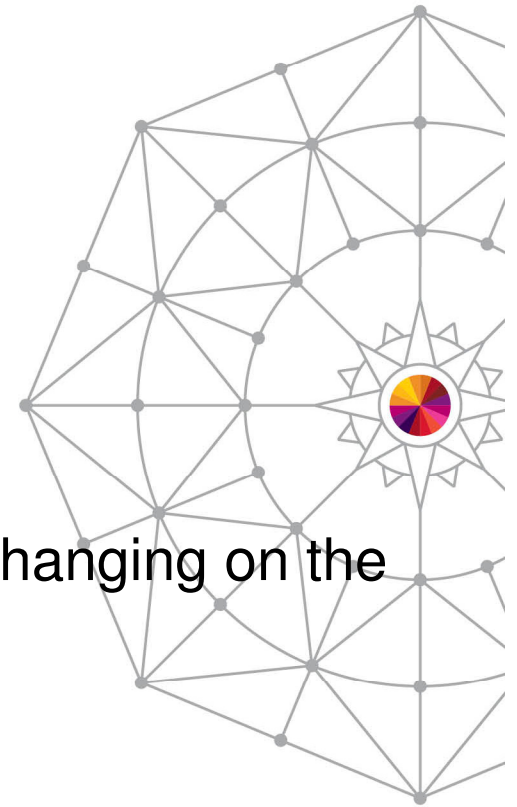
```

*****
Latch Summary
*****
  Latch Name                Holder                Waiter(s)                Latch Caller(s)
1.  IGWASRB(7F04DE00)+18    (35)0008:009F6CF0      (34)0008:009F6E88
2.  IGWHL1B(7FF62E40)+10   (35)0008:009F6CF0      (402)00E5:009AEE88
*****
*****
Tcb Latch Summary
*****
  TCB                Held Latch(es)                AWaited Latch(es)
34.  0008:009F6E88    IGWASRB(7F04DE00)+18          IGWASRB(7F04DE00)+18
35.  0008:009F6CF0    IGWASRB(7F04DE00)+18
35.  0008:009F6CF0    IGWASRB(7F04DE00)+18
                                IGWHL1B(7FF62E40)+10
402. 00E5:009AEE88    IGWHL1B(7FF62E40)+10
  
```

HL1B latches are ALWAYS suspicious

## Step 6: Learn about the holder

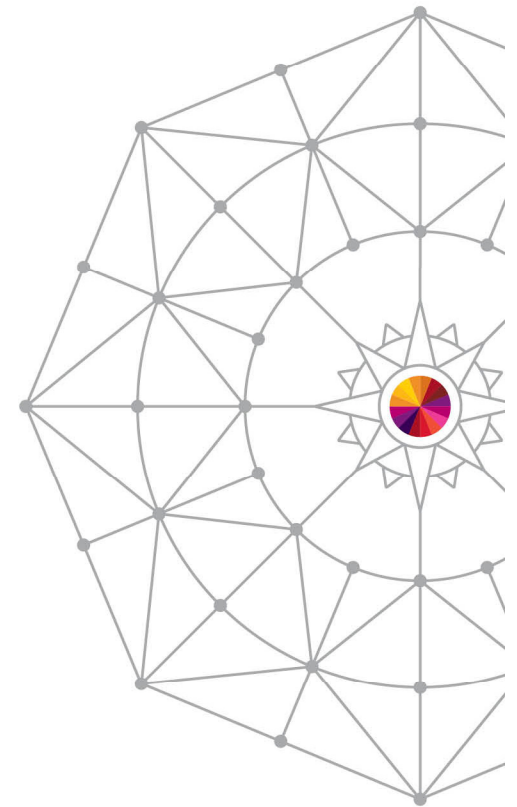
- Is it still alive?
  - Orphaned latches have a holder that's gone
  - Live holders are generally waiting
  - Check for IGWLHSUS (SSF Suspend)
- Is it in the PDSE address space?
  - This can indicate more serious issues
  - Can lead to a queue of holders obtaining and hanging on the latch
  - Call L2



## Step 7: Make sure the holder's dead

For holders NOT in the PDSE address space

- If the holder is dead: Good!
- If the holder is alive:
  - Get it out of the system
  - Cancel is the preferred option
  - Force if necessary
  - Recovery MAY resolve the latch on it's own



## Step 8: FREELATCH

V SMS,PDSE|PDSE1,FREELATCH(<latch address>,asid,tcb)

```

*****
Latch Summary
*****
  Latch Name                Holder                Waiter(s)                Latch Caller(s)
1.  IGWASRB(7F04DE00)+18    (35)0008:009F6CF0      (34)0008:009F6E88
2.  IGWHL1B(7FF62E40)+10   (35)0008:009F6CF0      (402)00E5:009AEE88
*****
Tcb Latch Summary
*****
  TCB                Held Latch(es)                AWaited Latch(es)
34.  0008:009F6E88    IGWASRB(7F04DE00)+18          IGWASRB(7F04DE00)+18
35.  0008:009F6CF0    IGWASRB(7F04DE00)+18          IGWASRB(7F04DE00)+18
35.  0008:009F6CF0    IGWASRB(7F04DE00)+18          IGWASRB(7F04DE00)+18
402. 00E5:009AEE88    IGWHL1B(7FF62E40)+10          IGWHL1B(7FF62E40)+10

```

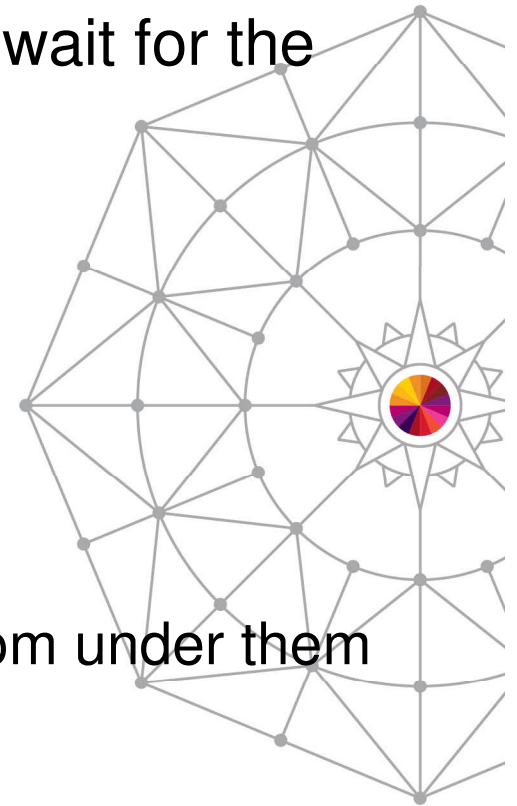
Holder's ASID : TCB

Latch Address

- Don't forget to add the latch offset!!
- WARNING: The holder MUST be out of the system

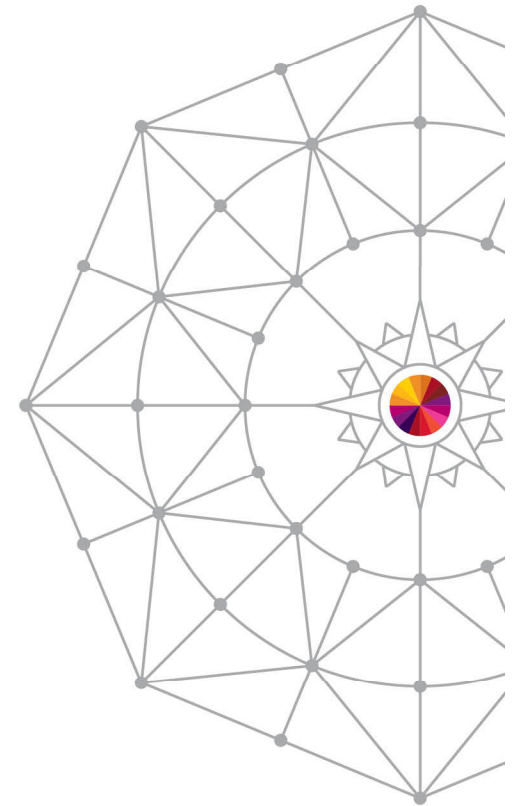
# The Aftermath

- Clearing resources may take a few minutes, wait for the next IGW038A message
- No Change:
  - Holder changed but still hung
  - PDSE1 restart (if applicable)
- Partial Clearing:
  - Get a new dump, repeat the process
  - Tasks may take 0F4's if a latch is freed out from under them
- Fully Cleared
  - PARTY!



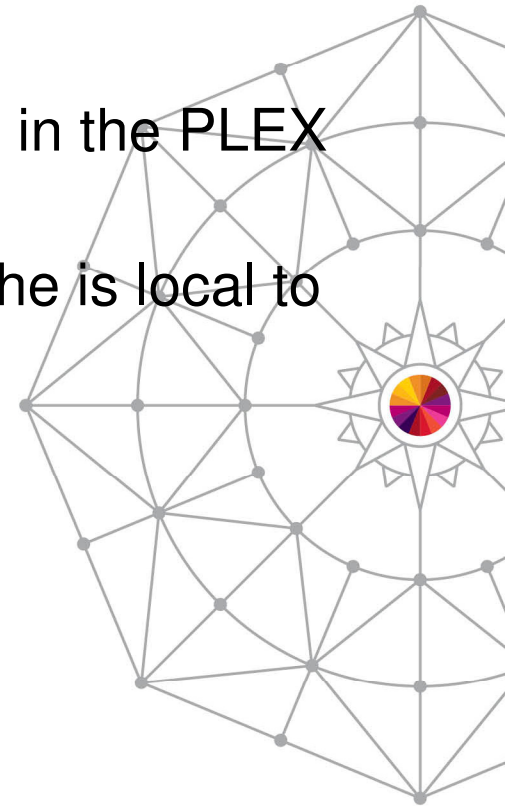
# Corruption

In-core and on DASD corruption



## In-core or on DASD?

- In-Core Corruption
  - If the PDSE is accessible from other systems in the PLEX the corruption is in-core
  - Issue is in the PDSE index cache and the cache is local to each PDSE ASID
- Corrupt on DASD
  - Will fail on all sharing systems
  - IEBPDSE will fail to run RC 0



# In-Core Corruption

- For either PDSE or PDSE1:

**V SMS,PDSE|PDSE1,REFRESH,DSN(dsname)[,VOL(volser)]**

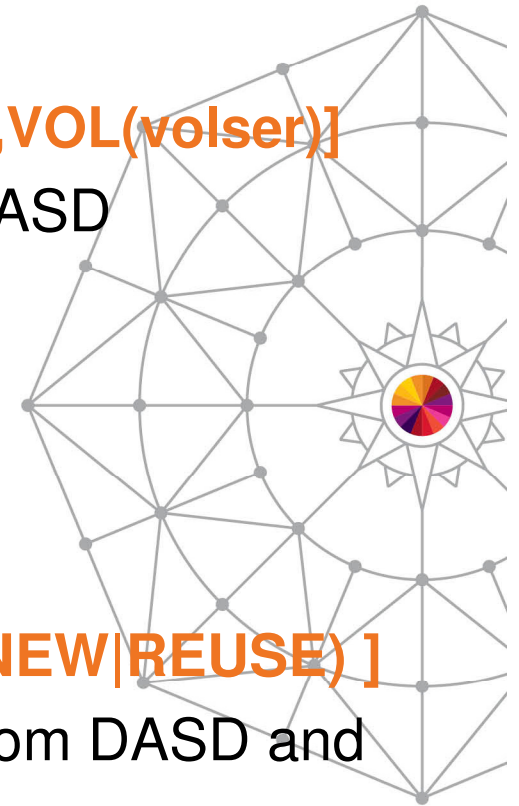
- Throws out in-core index and re-loads from DASD

- PDSE1 only

**V SMS,PDSE1,RESTART**

**[,QUIESCE(duration | 5 )[,COMMONPOOLS(NEW|REUSE) ]**

- PDSE1 restart will re-load ALL index pages from DASD and reconnect



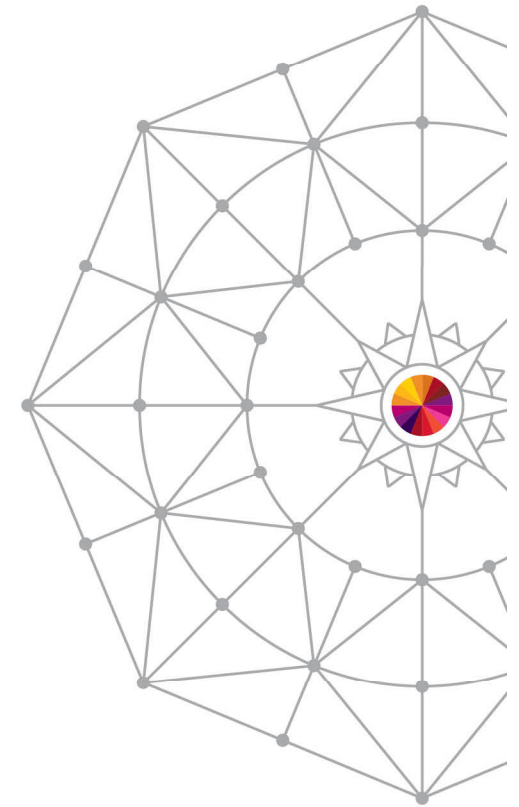


# Corrupt on DASD

- Gather a PHY Dump of the dataset
  - See appendix for JCL
  - L2 has formatting tools to help determine how the dataset broke
- An IEBCOPY may correct minor index issues
- Recovering from the last backup is often the only way to recover



# Rate this Session

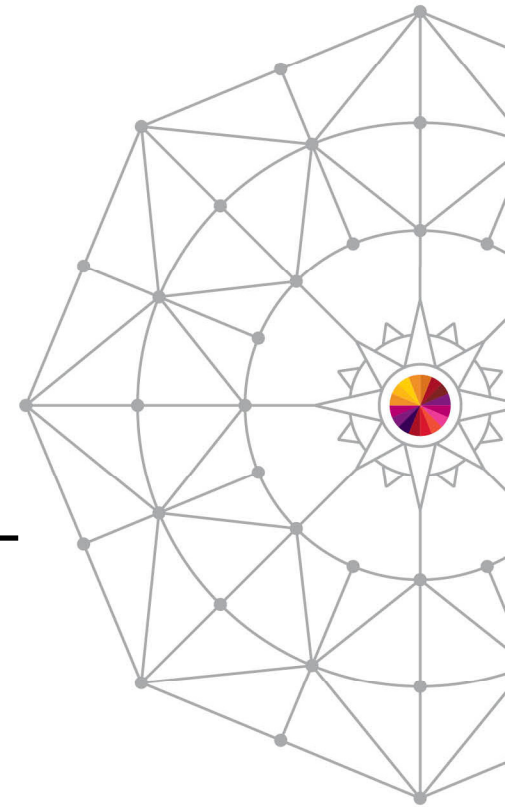


Complete your session evaluations online at [www.SHARE.org/Anaheim-Eval](http://www.SHARE.org/Anaheim-Eval)



# Appendix

## Parameters, Commands and JCL



# Appendix: Parameters, Commands and JCL

- PDSE Console Dump Parameters

```
COMM=(PDSE PROBLEM)  
JOBNAME=(*MASTER*, SMSPDSE*),  
SDATA=(PSA, CSA, SQA, GRSQ, LPA, LSQA, RGN, SUM, SWA, TRT, COUPLE  
, XESDATA), END
```

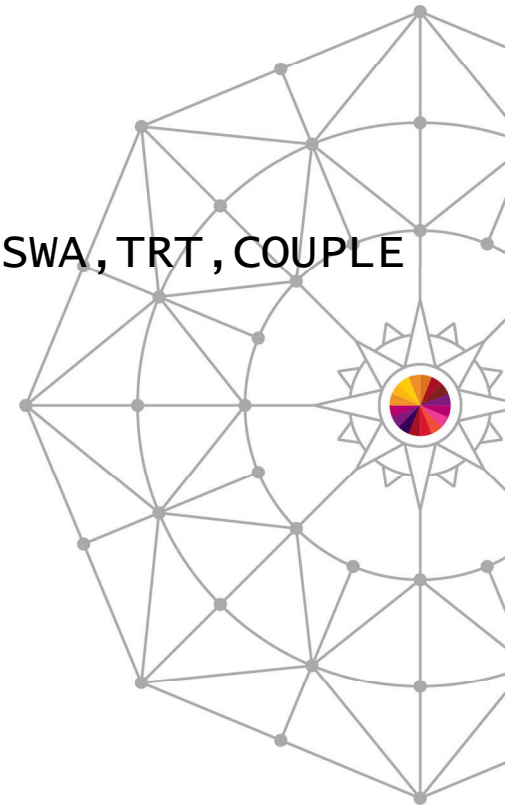
- IGDSMSxx Parameters:

- SMSPDSE1 restartable address space:

```
PDSE_RESTARTABLE_AS(NO | YES)
```

- PDSE Sharing Modes:

```
PDSESHARING(EXTENDED | NORMAL)
```



# Appendix: Parameters, Commands and JCL

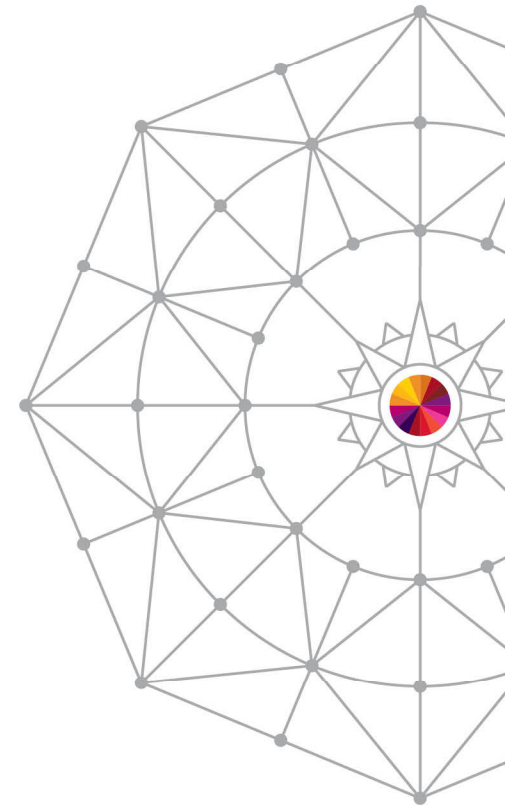
- IEBPDSE JCL (1.13 and above only)

```
//VALIDATE EXEC PGM=IEBPDSE
```

```
//SYSPRINT DD SYSOUT=*
```

```
//SYSIN DD DUMMY
```

```
//SYSLIB DD DISP=SHR,DSN=INPUT.PDSE.BAD
```



# Appendix: Parameters, Commands and JCL

- DSS PHYSICAL dump JCL

```
//DUMP      EXEC PGM=ADRDSSU
//SYSPRINT DD  SYSOUT=*
//OUT       DD  UNIT=3390,
//          VOL=SER=XXXXXX,
//          DISP=(NEW,KEEP),
//          SPACE=(CYL,(100,100)),
//          DSN=hi lev.DSSDUMP,
//          DCB=BLKSIZE=32760
//SYSIN     DD  *
DUMP  PIDY(vvvvvv) -
      OUTDD(OUT) -
      DATASET(INCLUDE(pdse.dataset.name)) -
      ALLDATA( * )
/*
```

