#SHAREorg

SHARE
Technology · Connections · Results

Testing and Debugging Your Installation's ACS Routines

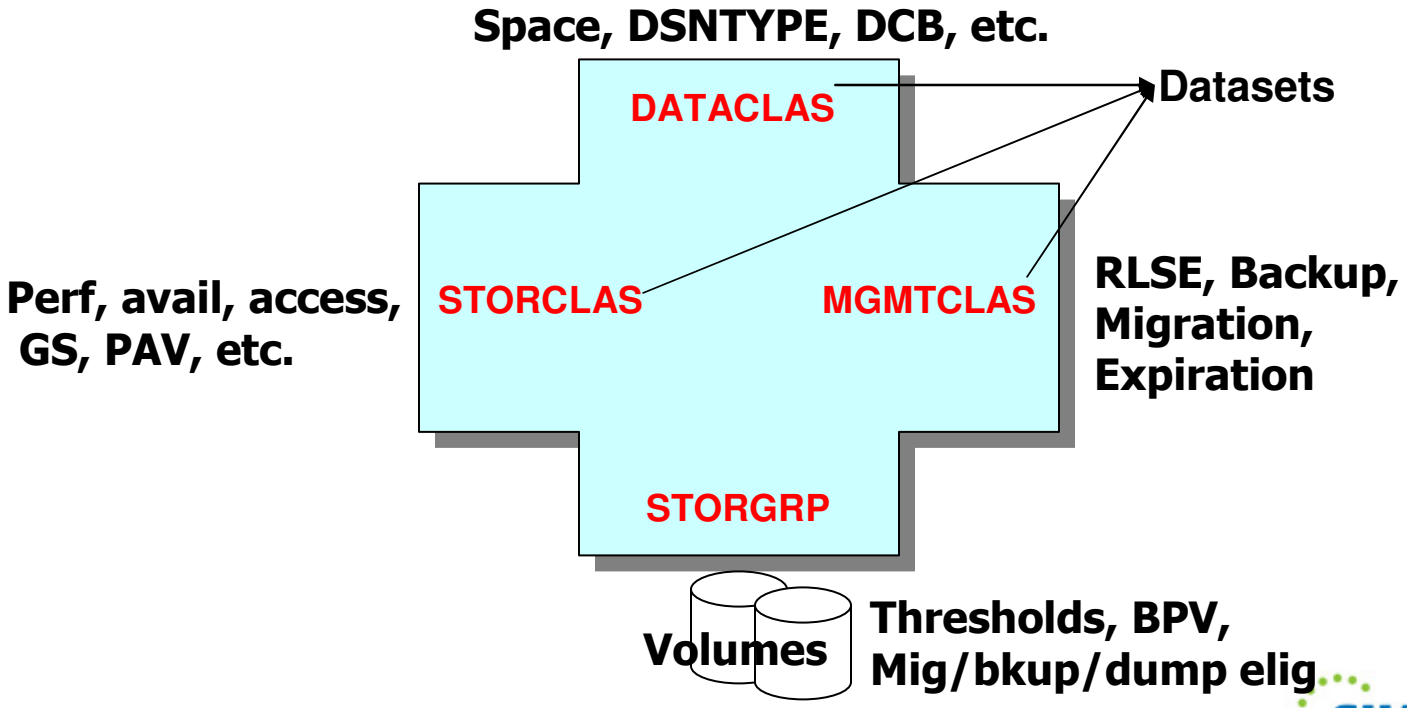
Session 10238

Steve Pryor
DTS Software, Inc.

770-922-2444

Automatic Class Selection is Important

- Implement storage management policy
- Executed for all new dataset allocations
  - SMS and non-SMS, all types of media



ACS Variables

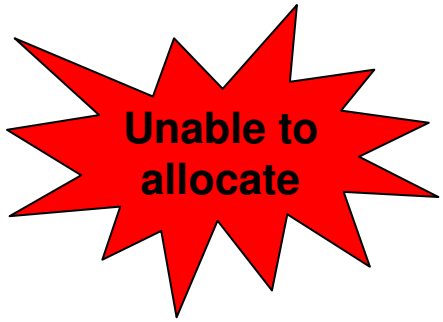
&ACCT_JOB	&DEF_STORCLAS	&JOB	&MSPARM	&SIZE
&ACCT_STEP	&DSN	&LABEL	&MSPOLICY	&SPACE_TYPE
&ACSENVIR	&DSNTYPE	&LIBNAME	&MSPOOL	&SYSNAME
&ALLVOL	&DSORG	&LLQ	&NQUAL	&SYSPLEX
&ANYVOL	&DSOWNER	&MAXSIZE	&NVOL	&UNIT
&APPLIC	&DSTYPE	&MEMHLQ	&PGM	&USER
&BLKSIZE	&EXPDT	&MEMLLQ	&RECORG	&XMODE
&DD	&FILENUM	&MEMN	&RETPD	
&DEF_DATACLAS	&GROUP	&MEMNQUAL	&SECLABL	
&DEF_MGMTCLAS	&HLQ	&MSPDEST	&SECOND_QTY	

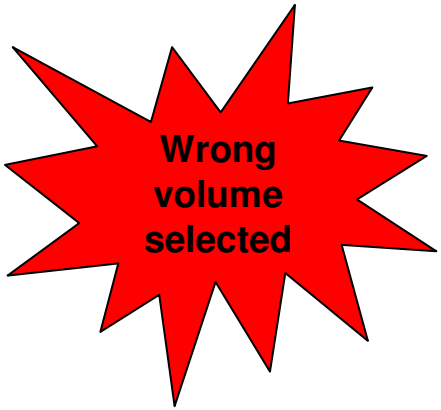
Read-write

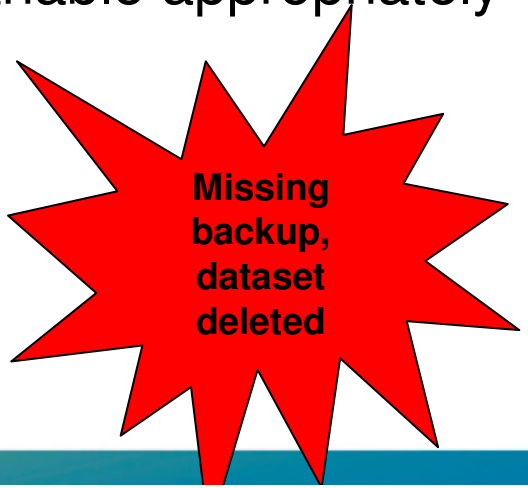
&DATACLAS	**&STORCLAS**	**&MGMTCLAS**	**&STORGRP**

ACS Routine Failures

- Coding Errors
  - Incorrect use of FILTLISTs or masks
  - Incorrect use of quotes
  - Incorrect use of GDG or special names
- Logic Errors
  - Using inappropriate R/O variables
  - Missing or early EXIT
  - Failure to SET R/W variable appropriately







Problem Variables

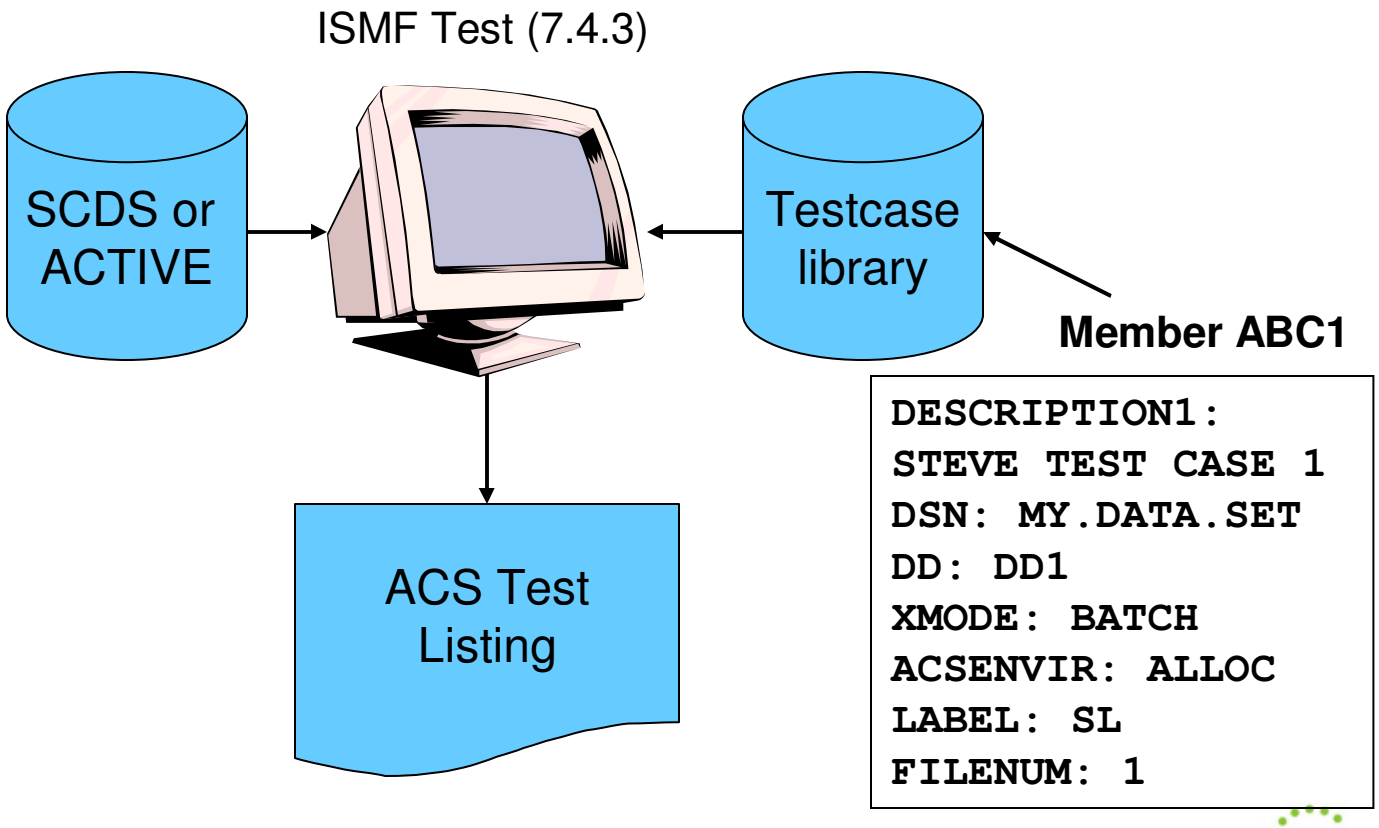
- &ALLVOL, &ANYVOL
  - Volume reference values = ‘REF=SD | ST | NS’
  - Recall or Recover and VOLCOUNT(ANY) = null
- &DSN
  - GDG = base name only, PDS = dataset name only
- Data Class assignment may change R/O variables
  - &RECORG, &NVOL, &DSNTYPE, &SIZE / MAXSIZE

Problem Variables

- &SIZE / &MAXSIZE
  - ACS test case must specify KB, ACS routine can compare KB/MB
  - z/OS 1.11 introduces &SPACE_TYPE and &SECOND_QTY
  - &SIZE / &MAXSIZE *recalculated* after DATACLAS routine runs
    - But this is not accounted for in ISMF test
  - For non-VSAM
    - &SIZE = Primary + directory
    - &MAXSIZE = Primary + directory + 15 secondaries
  - For VSAM (based on cluster, then data, then index)
    - &SIZE = Primary
    - &MAXSIZE = Primary + 122 secondaries * volcount
      - *May be different if ECR and/or Add'l Volume Amount = 'S'*

Native ISMF Test

- Test cases are flatfile PDS members

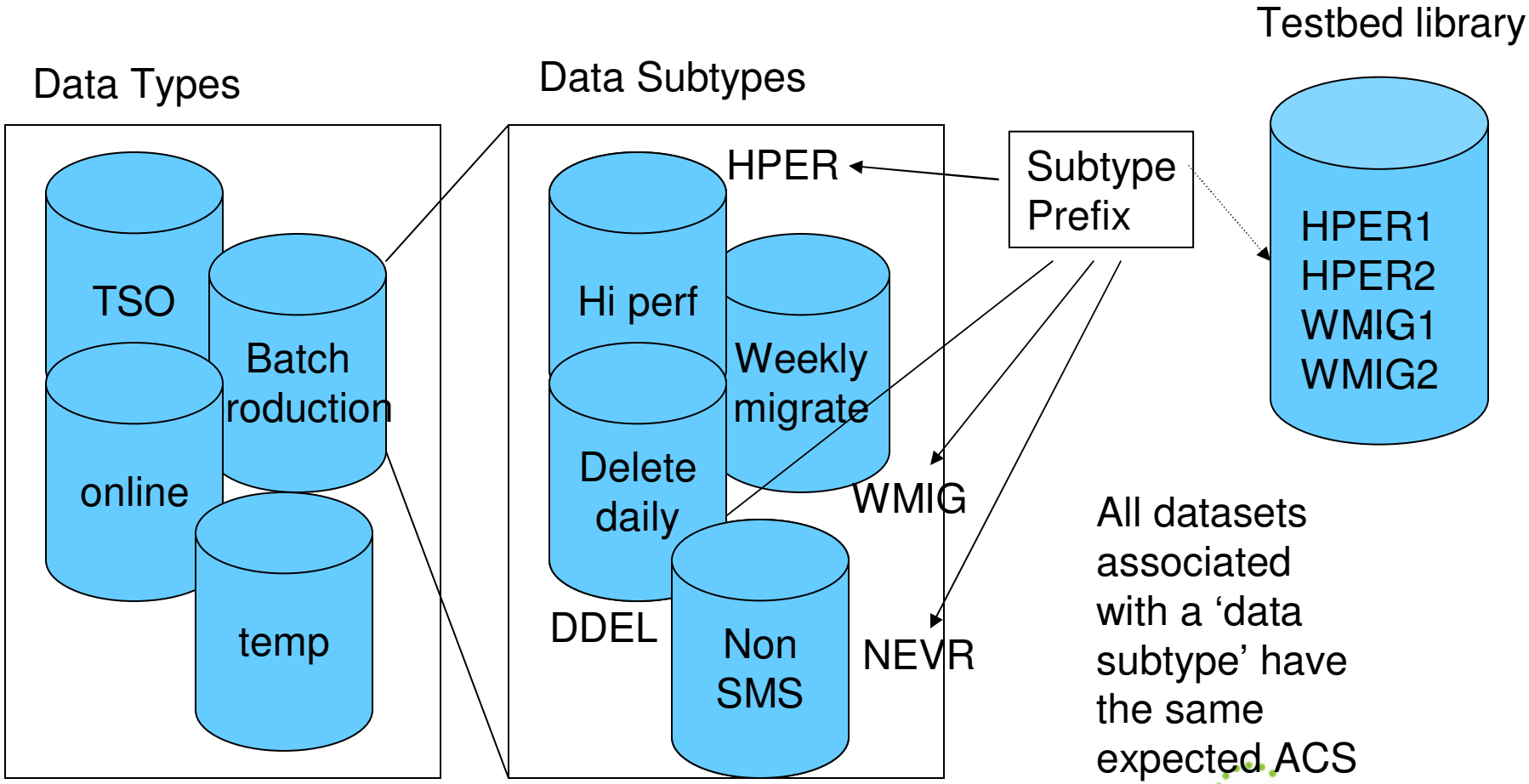


Create an ISMF Test Case

- Manually Define or Alter or Execute
  - via ISMF Test (7.4.1, 7.4.2, 7.4.3)
- Naviquest
  - Saved ISMF list
  - SMF data created by ACS routine exit
  - DCOLLECT, VMA extract

  - Data source may need to be cleaned up before testcase generation
    - SYS1.VTOC, SYS1.VVDS eliminated
    - duplicates eliminated

Naviquest Terminology
Data Classification



All datasets associated with a 'data subtype' have the same expected ACS results.

ISMF Testcase Listings

- ACS routines executed and EXIT codes
- SMS classes and SG assignments
- Messages

```
 ACS TESTING RESULTS
CDS NAME          : STEVE.SCDS
ACS ROUTINE TYPES: DC SC MC SG
ACS TEST LIBRARY : STEVE.TESTCASE.PDS

  ACS TEST
  MEMBER        EXIT CODE   RESULTS
  ---------     ----------  ---------------------------------
-
  DESCRIPTION: TEST CASE CREATED 2011/01/30 AT 08:06 BY STEVE
  EXPECTED RESULT:
  SJPA1                    0  DC = NULL VALUE ASSIGNED
                           0  SC = STC1
    MSG : SIZE GT 30M=000014C0
                           0  MC = STANDEF

ACS TESTING RC:  00
```

Saved ISMF List

- Create list via ISMF Dataset application (option 1)
  - Easy to create
- Build Test Cases via Naviquest (option 11.1)
- Easy to create
  - But does not always create valid test cases
    - Any invalid data in the testcase results unable-to-ALTER, unable-to-execute testcase, or worse
      - *Saved list uses SPACETYPE instead of SPACE_TYPE*
      - *Saved list uses 'K' or 'M' in SIZE variable (unable to ALTER but testcase executes, incorrectly)*
  - Non-dataset variables must be manually entered
    - JOB, XMODE, ACCT, DD, etc.
    - can be entered at Naviquest testcase generation time

DCOLLECT Data

- Run DCOLLECT for dataset info
  - ISMF option C or batch job
- Generate test cases
  - Naviquest (ISMF 11.2)

- Easy to create lots of test cases
  - Naviquest testcase generation filters out SYS1.VVDS, etc.
  - Non-dataset variables must be manually entered after testcase generation
    - JOB, XMODE, ACCT, DD, etc.

SMF Records from IGDACSSC

- STORCLAS exit supplied in SYS1.SACBCNTL
  - source code accesses ACERO, ACERW, ACSPM
    - must be reassembled for each new DFSMS release
  - writes ‘exit entered message’
  - write ACS R/O variables to SMF type 127
  - optionally write ACS R/O variables to job log if ACCT=DEBUG
- Program ACBTST
  - create testcases from SMF type 127 recs - one PDS member/rec
- Can create many test cases
  - includes all variables - JOB, STEP, DD, PGM, USER, etc.
  - Customize exit to select only desired jobs/applications
  - PDS member names Tnnnnnnn may need to be made meaningful

Naviquest

- Enhanced ACS Management – ISMF Option 11
  - Test Case Generation
  - Test Case Listings Comparison
  - Test Case Update

- Batch or ISPF
  - batch jobs invoke ISMF / ISPF via REXX
  - most ISPF functions can be performed in batch

Naviquest Regression Testing

- Test Case Execution
  - Execute Test Cases vs. SCDS 1 – save listing
  - Execute Test Cases vs. SCDS 2 – save listing
- Test Case Listing Comparison
  - Compare listing 1 vs. listing 2
  - Outputs:
    - Summary of exceptions listing
    - Exception Test Case PDS – test case library for test cases with different results
- Test Case Update
  - Update Test Case Library from Exception Library + Exception Listing
    - Replace DESCRIPTION2
    - DC=, SC=, MC=, SG= (limited to 80 chars)

WRITE Statements

- Use to display values of variables
  - Up to 110 characters

- Problems
  - TSO – displays only if allocation fails
  - Only first SG name displayed
  - Only first volume displayed for &ALLVOL / &ANYVOL
  - Maximum of 5 messages per ACS routine

Using 'Trigger' Values

- Testing vs. ACTIVE
  - Run test jobs, e.g., PGM=IEFBR14
  - Examine JESMSGS
- Testing vs. SCDS
  - Use ISMF Test
  - Examine Testcase Listing

- Select on special values
  - Special JOBNAME, DD, etc.

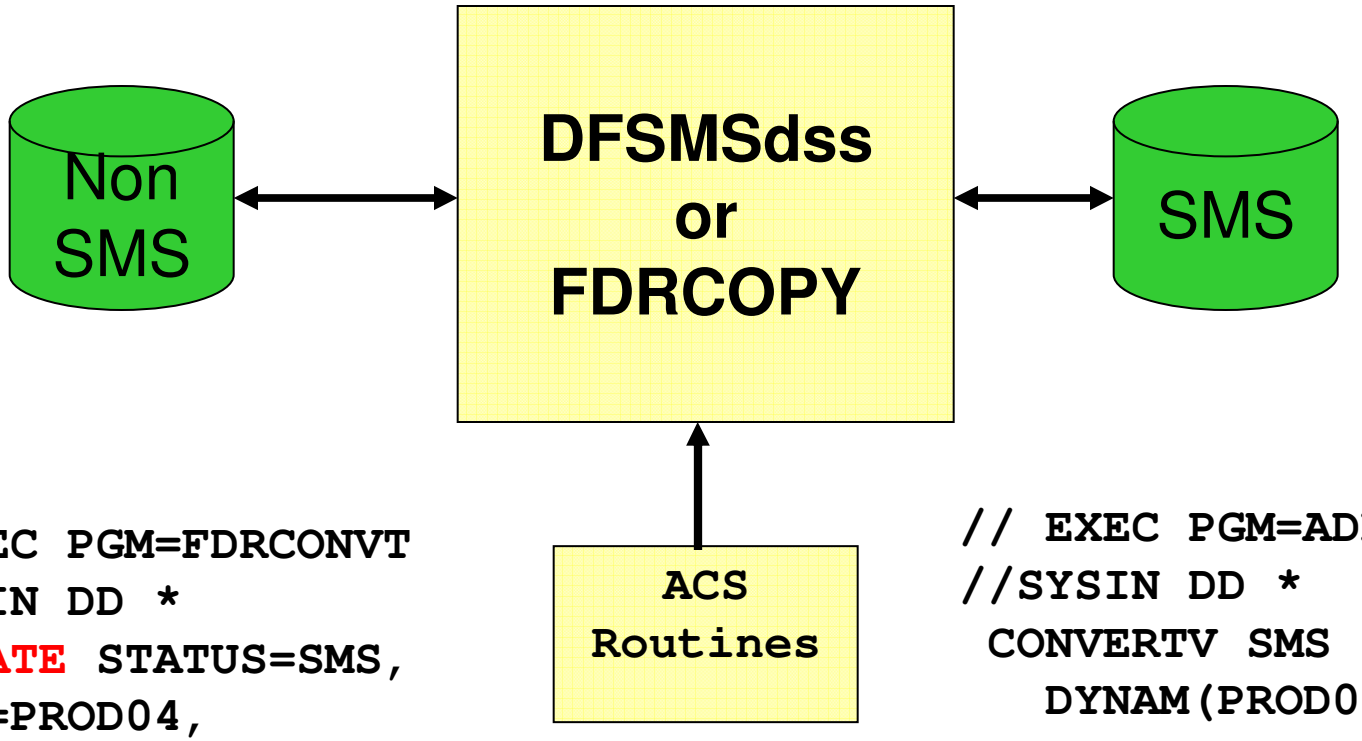
```
IF &DD = 'PI314159' THEN WRITE 'SIZE='&SIZE
IF &JOBNAME = TESTX* THEN DO
    WRITE 'SIZE='&SIZE
    IF &XMODE = TSO THEN EXIT CODE(99)
    END
```

  - Nonexistent SMS class, e.g.,

```
IF &STORCLAS = 'NULL' THEN SET &STORCLAS = ''
IF &ACSENVIR = 'ALLOCTST' (vendor use)
```

CONVERTVOL

- Use SMS Conversion-in-Place (SIM) to execute ACS routines




```
// EXEC PGM=FDRCONVT
//SYSIN DD *
SIMULATE STATUS=SMS,
  VOL=PROD04,
  REDETERMINE=YES
```

```
// EXEC PGM=ADRDSSU
//SYSIN DD *
 CONVERTV SMS TEST -
   DYNAM(PROD04)-
   RETDETERMINE
```

What ACS Testing Can't Do

- Expose ACS routine logic errors directly
- Use of LIKE= parameter in JCL
  - No &DSNTYPE, &DSORG, &SIZE/&MAXSIZE, RECORG
- Find Dataset Separation Profile errors
- Determine which volume will be selected for allocation
  - Use VOLSELMSG option in IGDSMSxx

IGDSMSxx Values

- Values affecting ACS Routines
  - ACSDEFAULTS(YES | NO)
  - BREAKPOINTVALUE(nnnnn)
  - DSNTYPE(LIBRARY | PDS | HFS)
  - REVERIFY(YES | NO)
  - USE_RESOWNER(YES | NO)

- Values affecting Allocation and Volume Selection
  - FAST_VOLSEL(ON | OFF)
  - GDS_RECLAIM(YES | NO)
  - OVRD_EXPDT(YES | NO)
  - USEEAV(YES | NO)
  - VOLSELMSG

ACS Testing and Debugging

- Keys to Success
  - Limit the size and complexity of the ACS routines
  - Log all updates to ACS and prepare a fallback
    - Use SETSMS with SAVEACDS
  - Use WRITE statements to show ACS variable values
  - Include easily-accessible ‘testing’ code in ACS routines
    - Perhaps duplicate the production ACS routines as ‘test’ segments in the test LPARs
  - Create / Maintain Testcase Library and regression testing regimen

Documentation

- *DFSMSdfp Storage Administration*
  Document Number: SC26-7402-15
  - Defining SMS Constructs
  - ACS Routine Language and Syntax
  - Naviquest
- *DFSMS Using ISMF*
  Document Number: SC26-7411-08
- *DFSMS Implementing System-Managed Storage*
  Document Number: SC26-7407-07
  - Implementing SMS management
- *DFSMS Using the New Functions*
  Document Number: SC26-7473-09
  - What's new in DFSMS (incl HSM, DFDSS, IDCAMS, et. al.)

Questions???