



SCC SMS/DEBUG

Locate and solve DFSMS class selection and allocation problems

The **SMS/DEBUG** component has been introduced to assist installations to get the most from their investment in DFSMS implementation. With SMS/DEBUG, tracing facilities are available for three key areas of DFSMS.

SMS/DEBUG provides valuable DFSMS debugging facilities:

ACS Routine Trace facility

Trace the current ACS routines and help find logic errors before they become critical.

SMS Volume Tracing

Shows all the volumes considered for SMS controlled allocations, threshold values, and which volume was selected.

DADSM Exit Tracing

Shows each volume considered for allocation and each DADSM reason code that was issued.

ACS Routine Tracing.

Have you ever tried to implement an ACS routine change and found it difficult to determine why it did not do what you intended, or tried to figure out how a certain DATACLAS was assigned to a data set?

Wouldn't it be great if you could see the logic flow of your ACS routines and the value of each variable...

With the ACS Routine Tracing facility of SMS/Debug, ACS routines become more powerful because it is easier to test and debug. Installations can trace individual allocations using additional JCL keywords to define trace parameters.

SMS Volume Tracing.

Have you ever tried to understand why DFSMS picked Volume SMS012 or maybe why it never seems to pick volume SMS212?

This can be especially difficult when the allocation occurs at 3 am...

With SMS Volume Tracing, SMS/Debug will show you all of the volumes DFSMS considered for allocation, the value of key threshold variables and storage group. The volumes placed in the Primary, Secondary and Tertiary group is also displayed.

DADSM Exit Tracing.

DFSMS messages only reflect the last DADSM return code, but there are times when it would be helpful to see all of the return codes that were posted.

Even if the allocation was successful, it could be useful to know that a volume was skipped because the VTOC was full...

SMS/Debug will display ALL the DADSM post-processing exit return codes and optionally, the DADSM preprocessing codes as well.

ACS Routine Tracing (sample)

```
1 //DONR JOB
2 // EXEC PGM=IEFB14
3 //DD1 DD DSN=DTS.TEST,UNIT=3380,SPACE=(TRK,(1)),
   SMSDEBUG = ACSA
ACC00510-* STARTING TRACE FOR SMSACS_TRACE-, DSN=DTS.TEST,VOL=, DD=DD1
ACC08100-* ..SMSACS VARIABLES BEFORE ACS ROUTINES
ACC08120-* ...DSN (TEST)
ACC08120-* ...DATACLAS ()
ACC08120-* ...STORCLAS ()
ACC08120-* ...MGMTCLAS ()
ACC08120-* ...STORGRP ()
ACC08120-* ...HLQ (DTS)
ACC08120-* ...LLQ (TEST)
ACC08120-* ...JOB (DONR)
ACC08120-* ...PGM (IEFB14)
ACC08120-* ...USER (DON)
ACC08120-* ...GROUP (DTS)
ACC08120-* ...DD (DD1)
ACC08121-* ...NVOL (1)
. . .
ACC08120-* ...*SPACE (TRK)
ACC08121-* ...*PSPACE (1)
ACC08121-* ...*SSPACE (0)
ACC08110-* ..TRACE FOR ACS ROUTINE STORCLAS
ACC08111-* ...COMPARE DSN (DTS.TEST) = LITERAL(*.*.C.C%%V*)
ACC08111-* ...COMPARE DSN (DTS.TEST) = LITERAL(*.*.D.C%%V*)
ACC08111-* ...COMPARE DSN (DTS.TEST) = LITERAL(*.*.I.C%%V*)
ACC08111-* ...COMPARE DSN (DTS.TEST) = LITERAL(*.*.O.C%%V*)
ACC08111-* ...COMPARE DSN (DTS.TEST) = LITERAL(*.*.*.C.C%%V*)
ACC08111-* ...COMPARE DSN (DTS.TEST) = LITERAL(*.*.*.D.C%%V*)
ACC08111-* ...COMPARE DSN (DTS.TEST) = LITERAL(*.*.*.I.C%%V*)
ACC08111-* ...COMPARE DSN (DTS.TEST) = LITERAL(*.*.*.O.C%%V*)
ACC08114-* ...BRANCH
ACC08111-* ...COMPARE DSN (DTS) = LITERAL (HFS)
ACC08111-* ...COMPARE DSN (DTS) = LITERAL (SYS1)
ACC08111-* ...COMPARE DSN (DTS) = LITERAL (SYS2)
ACC08111-* ...COMPARE DD (DD1) = LITERAL (STC1)
ACC08111-* ...COMPARE DD (DD1) = LITERAL (SMS)
ACC08111-* ...COMPARE DD (DD1) = LITERAL (TAPEX)
ACC08111-* ...COMPARE STORCLAS (*NULL*) = LITERAL (*NULL*)
ACC08113-* ...SET STORCLAS = *NULL*
ACC08114-* ...WRITE
ACC08114-* ...EXIT
ACC08100-* ..SMSACS VARIABLES AFTER CALL TO ACS ROUTINES
ACC08120-* ...DATACLAS ()
ACC08120-* ...STORCLAS ()
ACC08120-* ...MGMTCLAS ()
ACC08120-* ...STORGRP ()
ACC08120-* ...*STORGRP_TYPE (POOL)
```

(Note: This example only shows the STORCLAS routine but all of the ACS Routines are traced)

Volume Selection Tracing (sample)

```
// EXEC PGM=IEFB14
//DD1 DD DSN=DON.DD1,SPACE=(TRK,1),STORCLAS=STC1,SMSDEBUG=(NOACS,VOL)
//DD2 DD DSN=DON.DD2,SPACE=(TRK,1),STORCLAS=STC1,SMSDEBUG=(NOACS,VOLA)

SMSTRACE=VOLUME
ACC00510-* STARTING TRACE FOR SMS_TRACE-VOLUME, DSN=DON.DD1, VOL=, DD=DD1
ACC03750-* VOLSER STORGRP FLAGS LIST FREE CAP THRESH
ACC03751-* SMS002 STG1 EE9B9BA6 PRI 9 14 6
ACC03751-* SMS004 STG1 649B9BA2 SEC 2 14 0
ACC03751-* SMS001 STG1 649B9BA2 SEC 1 34 0
ACC03751-* SMS003 STG1 649B9BA2 SEC 0 14 0
ACC03752-* SMS SELECTED VOLUME SMS002
```

DADSM Exit Tracing (sample)

```
// EXEC PGM=IEFB14
//DD1 DD DSN=DON.DD1,SPACE=(TRK,1),SMSDEBUG=(NOACS,NOVOL,DADSM)
ACC00510-* STARTING TRACE FOR POST_DADSM-ALLOCATE, DSN=DON.DD1, VOL=SMS001, DD=DD1
ACC05201-* ALLOCATE FOR 1 TRACKS EXTENT 0 RC=12 - VTOC FULL
ACC00510-* STARTING TRACE FOR POST_DADSM-ALLOCATE, DSN=DON.DD1, VOL=SMS002, DD=DD1
ACC05201-* ALLOCATE FOR 1 TRACKS EXTENT 0 RC=0 - SUCCESSFUL
```