DFSMS Basics: How to Create/Modify an SMS Configuration and Write ACS Routines Part 1

Steve Huber
IBM Corporation

March 3, 2011
9013

Agenda

- Intro to SMS (Configuration and ACS)
- Configuration Walk-thru
- ACS Walk-thru
- Summary

Introduction to an SMS Environment

- Base Configuration Definition
  - Default device geometry
  - Defines the Systems in the SMS plex
- 4 ACS (Automatic Class Selection) Construct Types
  - Data Class
  - Storage Class
  - Management Class
  - Storage Group
    - ACS Routines are run in this order
- Provides the default allocation values

Introduction to an SMS Environment (cont)

- What is a Data Class?
  - **RECORG or RECFM**
  - **LRECL**
  - **Space**
  - **DSNTYPE**
  - **Volume count**
  - **VSAM attributes**
  - **RETPD or EXPDT**
  - **Compaction**

Introduction to an SMS Environment (cont)

- What is a Storage Class?
  - **This IS the attribute that makes a data set SMS managed**
  - **Performance attributes**
    - **Direct & sequential millisecond response**
    - **Direct & sequential bias**
    - **Initial access response time**
  - **Availability**
  - **Accessibility**
  - **Guaranteed space**
  - **Guaranteed synchronous write**

Introduction to an SMS Environment (cont)

- What is a Management Class?
  - **Space management attributes**
    - **Expiration & retention attributes**
    - **Migration attributes**
    - **GDG management attributes**
  - **Backup attributes**
    - **Backup frequency**
    - **Backup versions**
    - **Backup retention**
  - **Class transition attributes**
  - **Aggregate backup attributes**

Introduction to an SMS Environment (cont)

- What is a Storage Group?
  - **Physical storage managed by SMS**
    - **Collection of DASD volumes**
    - **Volumes in tape libraries**
    - **Volumes in optical libraries**
    - **Virtual I/O storage**
  - **Can be in ENABLE, QUINEW, QUIALL, DISNEW, DISALL or NOTCON status**
  - **Can be set to auto migrate, auto backup and/or auto dump**

Introduction to an SMS Environment (cont)

- Translating the ACS Routines
  - **Done through ISMF**
  - **Converts (compiles) the ACS source and puts it in the SCDS**
- Validating the SMS Configuration
  - **Done through ISMF**
  - **Verifies that all classes/groups assigned in the ACS routines exist**
- Activate the new configuration
  - **Done through ISMF**
  - **Loads the SCDS into an ACDS**

Introduction to ACS Environment

- What is an ACS Routine?
  - **User written code to affect an allocation**
  - **Used to determine SMS classes and storage groups**
  - **Used for both data sets and objects**
  - **Can override specifications of SMS classes and groups on:**
    - **JCL DD statements**
    - **Dynamic allocation requests**
    - **DFSMSdssCOPY, RESTORE & CONVERTV**
    - **DFSMShsm RECALL & RECOVER**
    - **IDCAMS DEFINE, ALTER & IMPORT**
    - **OAM STORE, CHANGE & class transition**

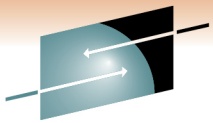
Introduction to ACS Environment (cont)

- ACS General Rules
  - **Keep them simple and straightforward**
    - **Minimize exceptions**
    - **Maximize FILTLIST usage**
  - **Keep them easy to maintain and understand**
    - **Use SELECT instead of IF when possible**
    - **EXIT the routine as soon as possible**
    - **Use OTHERWISE whenever possible**
    - **Comments, comments, comments**

Introduction to ACS Environment (cont)

- ACS Language Statements
  - **PROC - beginning of routine**
  - **FILTLIST – defines filter criteria**
  - **DO – start of statement group**
  - **SELECT – defines a set of conditional statements**
  - **IF – conditional statement**
  - **SET – assigns a read/write variable**
  - **WRITE – sends message to end user**
  - **EXIT – immediately terminates ACS routine**
  - **END – end of statement group**
- More info can be found in the pub hand-out

Introduction to ACS Environment (cont)



- ACS Read Only Variables
  - **Majority of the ACS variables**
  - **Contain data and system information**
  - **Reflect what is known at the time of the request**
  - **Can only be used for comparison**
- More info can be found in the pub handouts

Introduction to ACS Environment (cont)

- ACS Read/Write Variables
  - **Used as values in comparisons (READ)**
  - **Used to assign values (WRITE)**
  - **4 Read/Write variables**
    - **&DATACLAS**
    - **&STORCLAS**
    - **&MGMTCLAS**
    - **&STORGRP**
- The ACS PROC statement must identify which R/W variable it is setting
- More info can be found in the pub handouts

Introduction to ACS Environment (cont)

- Translating ACS Routines
  - **Done via ISMF**
  - **Checks for syntax errors**
  - **Converts (compiles) ACS source into object and stores it into the SCDS**
- Validating the SMS Configuration
  - **Also done through ISMF**
  - **Verifies that all classes/groups assigned in the ACS routines exist**
- Activating the SMS Configuration
  - **3 Methods**
    - **SETSMS SCDS(scdsname) operator command**
    - **ISMF option 8 (Control Data Set Application) then select option 5 (Activate)**
    - **Type ACTIVATE on the ISMF command line**
  - **Loads the SCDS into the ACDS**

Storage Administrator Setting

- Set yourself up as a Storage Administrator
  - ISMF
  - 0 - Profile Options
  - 0 - User Mode Selections
  - 2 – Storage Administrator
  - End/Exit 3 times

Configuration Walk-thru 1

- Create a SCDS
  - Submit SCDS job
- Complete the BCD (Base Configuration Definition)
  - Default Device Geometry
  - Trks/Cyl
  - System Name

Configuration Walk-thru 2

- Create a Storage Class named
  - Default
    - With the desired attributes

Configuration Walk-thru 3

- Create a Storage Group named
  - Default
- Put at least 1 volume in it

ACS Walk-thru 1

- Create an ACS Routine w/ PROC, FILTLIST and SET
  - Create a filter of SYSTEM which encompasses SYS1 and SYS2
- Translate the ACS routine

PROC, FILTLIST and Basic SET Examples

```
000100 PROC STORCLAS
000200 /*                    START FILTLISTS                    */
000300 FILTLIST NAME     INCLUDE('BOB','PETE')
000400 /*                    END   FILTLISTS                    */
000500   SET &STORCLAS = ''
000600 END
```

ACS Walk-thru 2

- If/Then Logic
  - Add IF/THEN logic
    - Compare the HLQ to the NAME filter and set SC to DEFAULT
- Translate the ACS routine

IF Example

```
000100 PROC STORCLAS
000200 /*                      START FILTLISTS                          */
000300 FILTLIST NAME     INCLUDE('BOB','PETE')
000400 /*                      END   FILTLISTS                          */
000410 /*****************************************************************/
000420 /*                      START SC LOGIC                           */
000430 IF &HLQ EQ &NAME THEN
000440   DO
000441     SET &STORCLAS EQ 'DEFAULT'
000450     EXIT
000460   END
000470 /*          SET NULL SC IF NOTHING ELSE ASSIGNED                 */
000500 SET &STORCLAS = ''
000600 END
```

ACS Walk-thru 3

- SELECT
  - Add a SELECT statement
    - SELECT on Read Only variable &DSN(1)
      - *2 methods to accomplish this*
      - *When it matches the NAME filter, set the SC to DEFAULT*
- Translate the ACS routine

SELECT Example(s)

```
000600 /*                        START SC LOGIC                          */
000700 IF &HLQ EQ &NAME THEN
000800    DO
000900       SET &STORCLAS EQ 'DEFAULT'
001000       EXIT
001100    END
001200 /*                        SELECT METHOD 1                         */
001300 SELECT (&DSN(1))
001400    WHEN ('BOB') SET &STORCLAS EQ 'DEFAULT'
001500    WHEN (&NAME) SET &STORCLAS EQ 'DEFAULT'
001600    END
001700    EXIT
001800 /*                        SELECT METHOD 2                         */
001900 SELECT
002000    WHEN (&HLQ EQ 'BOB') SET &STORCLAS = 'DEFAULT'
002100    WHEN (&HLQ EQ &NAME) SET &STORCLAS = 'DEFAULT'
002200    END
002300    EXIT
002400 /*              SET NULL SC IF NOTHING ELSE ASSIGNED              */
002500 SET &STORCLAS = ''
002600 END
```

ACS Walk-thru 4

- WRITE
  - Add a WRITE statement
    - WRITE ‘message’
- Translate the ACS routine

WRITE Examples

```
000600 /*                      START SC LOGIC                          */
000700 IF &HLQ EQ &NAME THEN
000800    DO
000900       SET &STORCLAS EQ 'DEFAULT'
000910       WRITE 'STORCLAS ROUTINE ASSIGNED STORCLAS: ' &STORCLAS
001000       EXIT
001100    END
001200 /*                      SELECT METHOD 1                         */
001300 SELECT (&DSN(1))
001400    WHEN ('BOB') SET &STORCLAS EQ 'DEFAULT'
001500    WHEN (&NAME) SET &STORCLAS EQ 'DEFAULT'
001600    END
001610       WRITE 'STORCLAS ROUTINE ASSIGNED STORCLAS: ' &STORCLAS
001700    EXIT
001800 /*                      SELECT METHOD 2                         */
001900 SELECT
002000    WHEN (&HLQ EQ 'BOB') SET &STORCLAS = 'DEFAULT'
002100    WHEN (&HLQ EQ &NAME) SET &STORCLAS = 'DEFAULT'
002200    END
002210       WRITE 'STORCLAS ROUTINE ASSIGNED STORCLAS: ' &STORCLAS
002300    EXIT
```

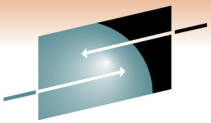
Translate and Validate

- Translate the ACS routines via ISMF 7 / 2 to the configuration (SCDS) data set

- Validate the configuration with ISMF 7 / 3 to the configuration (SCDS) data set

Summary

- Upon completion of this session, you should…
  - Have a better understanding of the SMS environment.
  - Understand how to create/modify an SMS Configuration.
  - Understand how to write a basic ACS routine.
  - Understand how to Translate an ACS routine and Validate an SMS Configuration.
  - Understand how to determine what Translate and/or Validate error(s) occurred.



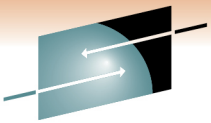
Notices & Disclaimers

Product information and data has been reviewed for accuracy as of the date of initial publication. Product information and data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or programs(s) described herein at any time without notice.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such such products, programs or services available in all countries in which IBM operates or does business. Consult your local IBM representative or IBM Business Partner for information about the product and services available in your area.

Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does notinfringe IBM's intellectually property rights, may be used instead. It is the user's responsibility to evaluate and verify the operation of any non-IBM product, program or service.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS"WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR INFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. IBM is not responsible for the performance or interoperability of any non-IBM products discussed herein.



Notices & Disclaimers

The performance data contained herein was obtained in a controlled, isolated environment. Actual results that may be obtained in other operating environments may vary significantly. While IBM has reviewed each item for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere.

The responsibility for use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's or user's ability to evaluate and integrate them into their operating environment. Customers or users attempting to adapt these techniques to their own environments do so at their own risk. IN NO EVENT SHALL IBM BE LIABLE FOR ANY DAMAGE ARISING FROM THE USE OF THIS INFORMATION, INCLUDING BUT NOT LIMITED TO, LOSS OF DATA, BUSINESS INTERRUPTION, LOSS OF PROFIT OR LOSS OF OPPORTUNITY.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not necessarily tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or anyother claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Trademarks

DFSMSdfp, DFSMSdss, DFSMShsm, DFSMSrmm, IBM, IMS, MVS, MVS/DFP, MVS/ESA, MVS/SP, MVS/XA, OS/390, SANergy, and SP are trademarks of International Business Machines Corporation in the United States, other countries, or both.

AIX, CICS, DB2, DFSMS/MVS, Parallel Sysplex, OS/390, S/390, Seascape, and z/OS are registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Domino, Lotus, Lotus Notes, Notes, and SmartSuite are trademarks or registered trademarks of Lotus Development Corporation. Tivoli, TME, Tivoli Enterprise are trademarks of Tivoli Systems Inc. in the United States and/or other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

Other company, product, and service names may be trademarks or service marks of others.