#SHAREorg

SHARE
Technology • Connections • Results

DFSMS Basics: SMS Configuration and ACS Routines Introduction / Demonstration

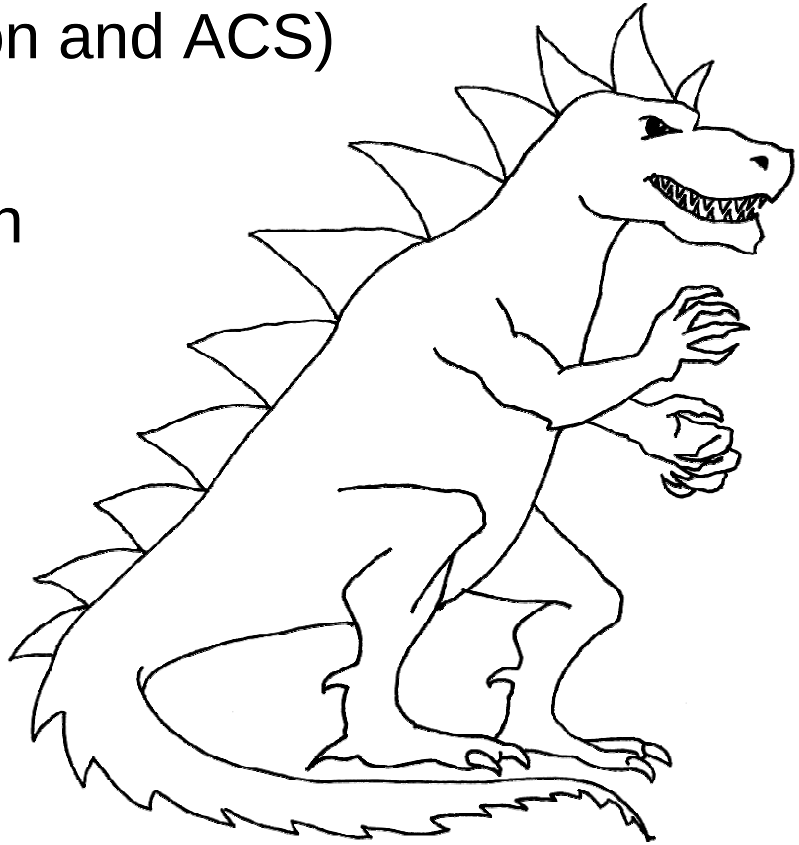
Neal Bohling and Tom Reed
IBM

February 6, 2013
Session Number 12323

SHARE in San Francisco 2013

Agenda

- Intro to SMS (Configuration and ACS)
- Configuration Walkthrough
- ACS Walkthrough
- Summary



Introduction to SMS

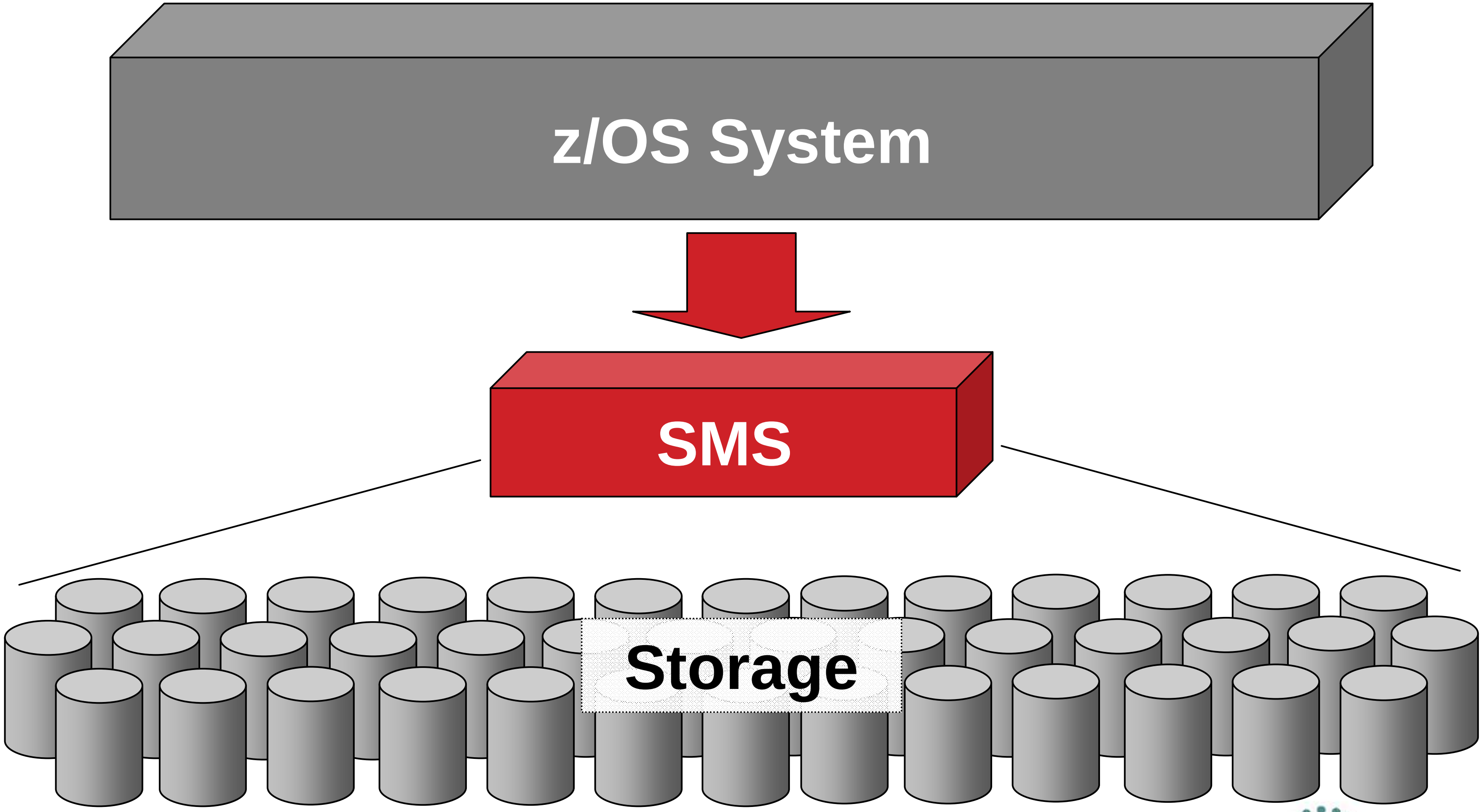
S STORAGE

M MANAGEMENT

S SUBSYSTEM

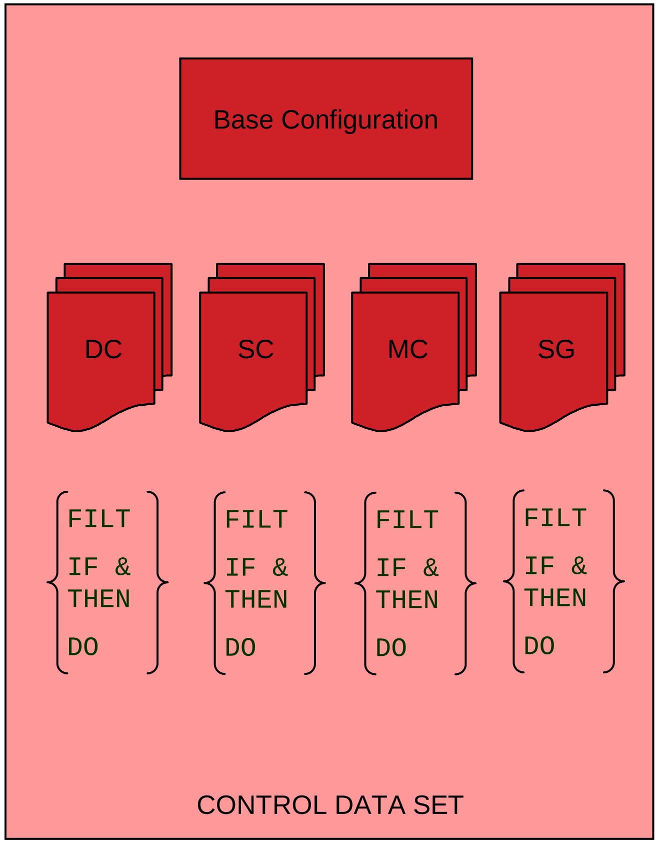
- DFSMS facility designed for automating and centralizing storage management.
- Allows you to define
  - data allocation characteristics
  - performance and availability goals,
  - backup and retention requirements
  - storage requirements
- Benefits:
  - Improves storage space use
  - Allows central control
  - Enables you to manage storage growth more efficiently

Introduction to SMS Environment



SMS Configuration Includes

- **Base Configuration**
  - Installation defaults (device geometry)
  - Systems included in the *SMS complex*
- **Constructs**
  - Data Classes – basic allocation defaults
  - Storage Classes – access attributes
  - Management Classes – migration information
  - Storage Groups – collection of volumes
- **Automating Class Selection (ACS)**
  - User-defined script
  - One per construct
  - Selects construct based on various criteria
- **Stored in the Control Data Sets (CDS)**
  - Active CDS – ACDS
  - Source CDS – SCDS
  - Communication CDS - COMMDS



Introduction to an SMS Environment

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

- Basically, everything you'd need to define a data set

DATACLAS

defines

Default JCL parameters

Introduction to an SMS Environment

- **What is a Storage Class?**
  - **Defines performance and availability**
  - **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
  - RLS Cache / Lock information

Introduction to an SMS Environment

- **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

  - It is not required to define these, but they are helpful when you do.

HSM references the MGMTCLAS to know how to handle each data set

Introduction to an SMS Environment

- **What is a Storage Group?**
  - A storage group is a logical group of at least one volume
  - 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 – Structure Summary

- Base Configuration – installation defaults

- DATACLAS – JCL defaults
- STORCLAS* – access requirements / required for SMS
- MGMTCLAS – migration / backup attributes
- STORGRP* – groups of volumes

- * at minimum, you need a storage class and storage group

Introduction to ACS Environment

- **What is an ACS Routine?**
  - User written code
  - Determines which SMS classes and storage groups are assigned
  - Used for both data sets and objects
  - One per type of construct
  - They run at ALLOCATION time
  - Process in this order:
    - DATACLAS
    - STORCLAS
    - MGMTCLAS
    - STORGRP

Introduction to ACS Environment (cont)

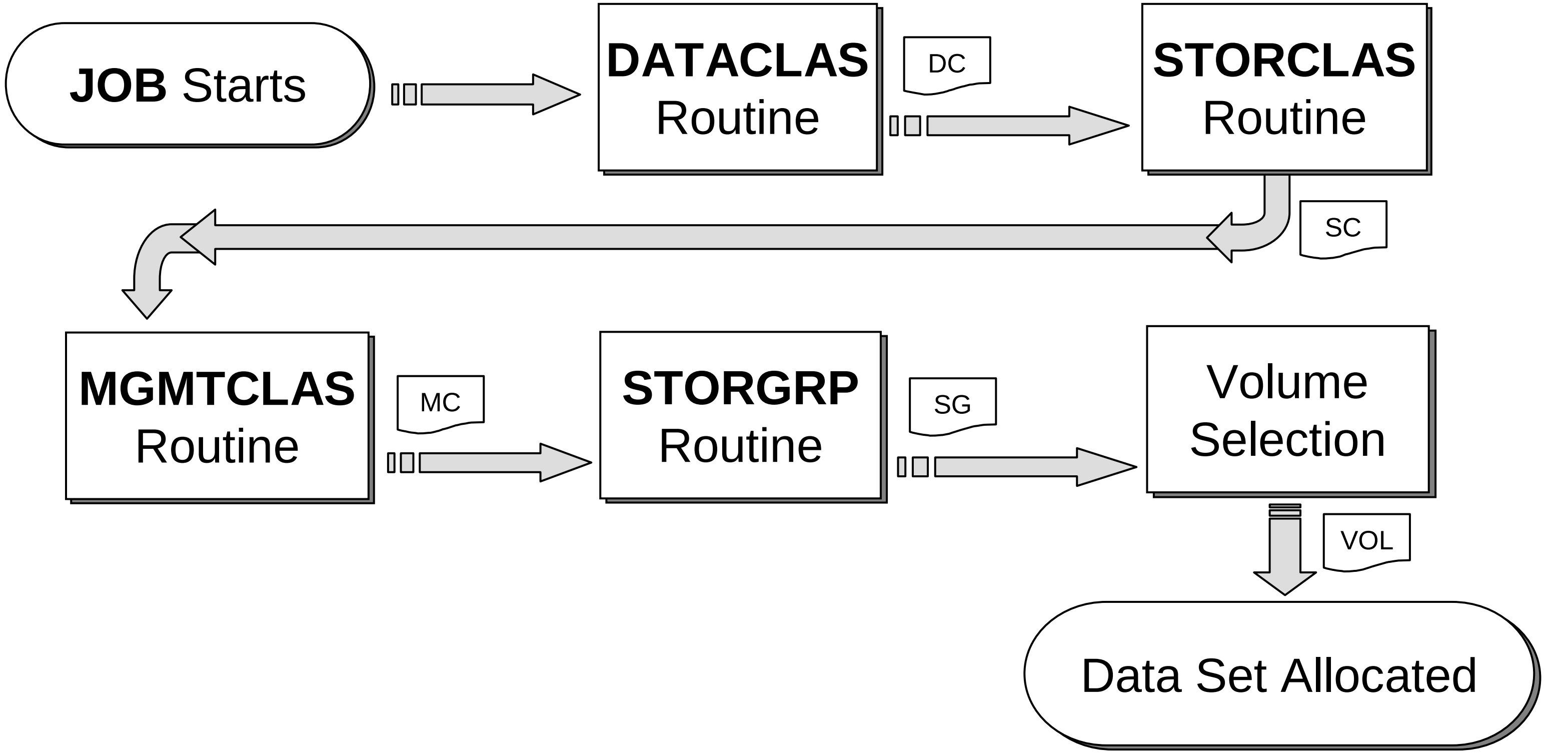
- **Write Your Routines**
  - Edit via your favorite editor

- **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**
  - Loads the SCDS into the ACDS
  - 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

ACS Routine Process Flow



Introduction to ACS Environment (cont)

- **ACS General Rules**
  - ***Know your logic before you code***
  - **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

KISS

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
  - **/* COMMENT */** - comments a line

Read Variables

- **READ ONLY Variables**
  - 47 different 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
- Examples:
  - `&DSORG`
  - `&DSNTYPE`
  - `&SIZE`
  - `&HLQ`

- **READ/WRITE Variables**
  - Used as values in comparisons (READ)
  - Used to assign values (WRITE)
  - 4 Read/Write variables
    - `&DATACLAS`
    - `&STORCLAS`
    - `&MGMTCLAS`
    - `&STORGRP`

- The PROC statement must identify which R/W variable it is setting

A Few “Gotchas”

- **Numeric constants are easy: just numbers**
  - *&NQUAL = 5*

- **Suffixes : sizes require KB or MB suffix**
  - *&MAXSIZE = 100MB*

- **String literals are in single quotes**
  - *&HLQ = ‘TEST’*

- **Masks are in NOT in quotes**
  - &DSN = SYS1.*LIB

- **&& is AND, | is OR**

- **DO / END mismatches**

- **Watch for fall-through logic in your IF and SELECT**

Introduction to ACS Environment (cont)

- **Write the ACS Routines**
    - Saved in a text format

- **Translate ACS Routines**
    - Converts to object code and inserts into the SCDS

- **Validate the SMS Configuration**
    - Verifies your construct allocation (do they all exist?)

- **Activate the SMS Configuration**

- ***Note: translate / validate from the highest z/OS level in your PLEX***

DEMO

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 to completely exit ISMF

Configuration Prep Walkthrough 1

- Create a SCDS
  - Submit DEFSCDS job
  - Verify via ISPF 3.4

- Complete the BCD (Base Configuration Definition)
  - ISMF 8 / 2
  - Default Device Geometry
  - Trks/Cyl - 56664
  - System Name – SYSTEM1

Configuration Prep Walkthrough 2

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

Configuration Prep Walkthrough 3

- Create a Storage Group named
  - Default

- Put at least 1 volume in it

- BLANK SLIDE

ACS Walkthrough

- **Always understand your logic BEFORE coding**
- Logic for this demo:
  - Make data set SMS-managed (assign SC) IF:
    - The Data Set is large (> 1000 MB)
    - Or the HLQ matches one of a few USERIDs
    - Or if the second level qualifier matches a FILTLIST
  - Otherwise, leave it non-managed (no SC)

ACS Walkthrough 1

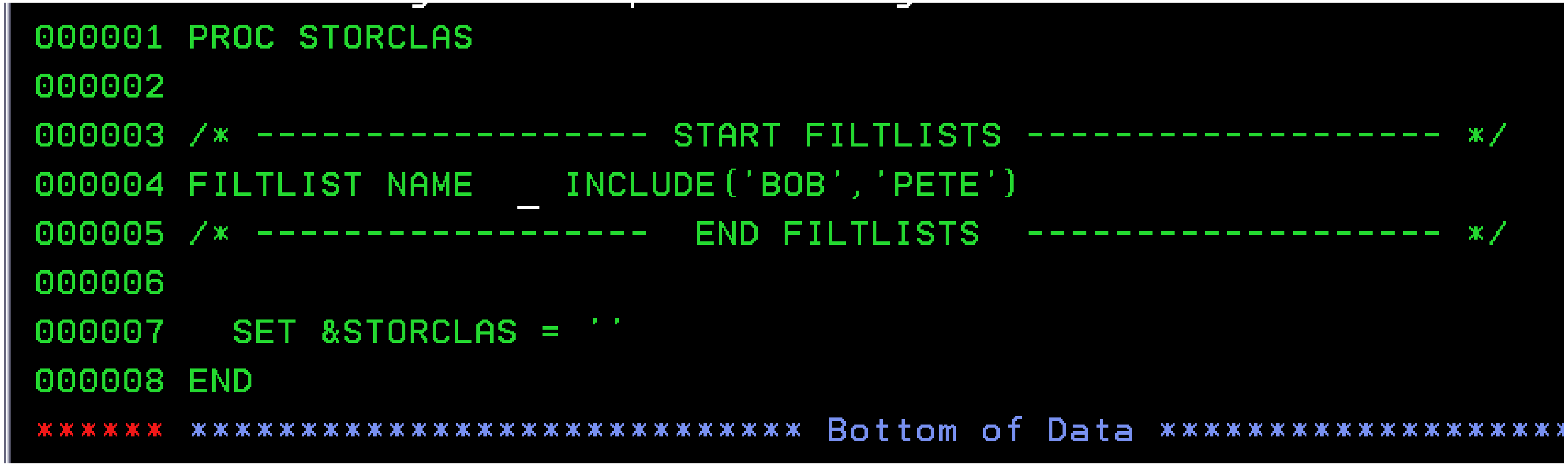
- **NOTE: These ACS exercises are an exercise in syntax, NOT logic.**

- Create an ACS Routine w/ PROC, FILTLIST and SET
    - Create a filter of NAME which encompasses BOB and PETE

- Translate the ACS routine

PROC, FILTLIST and Basic SET Examples

- Basic Framework

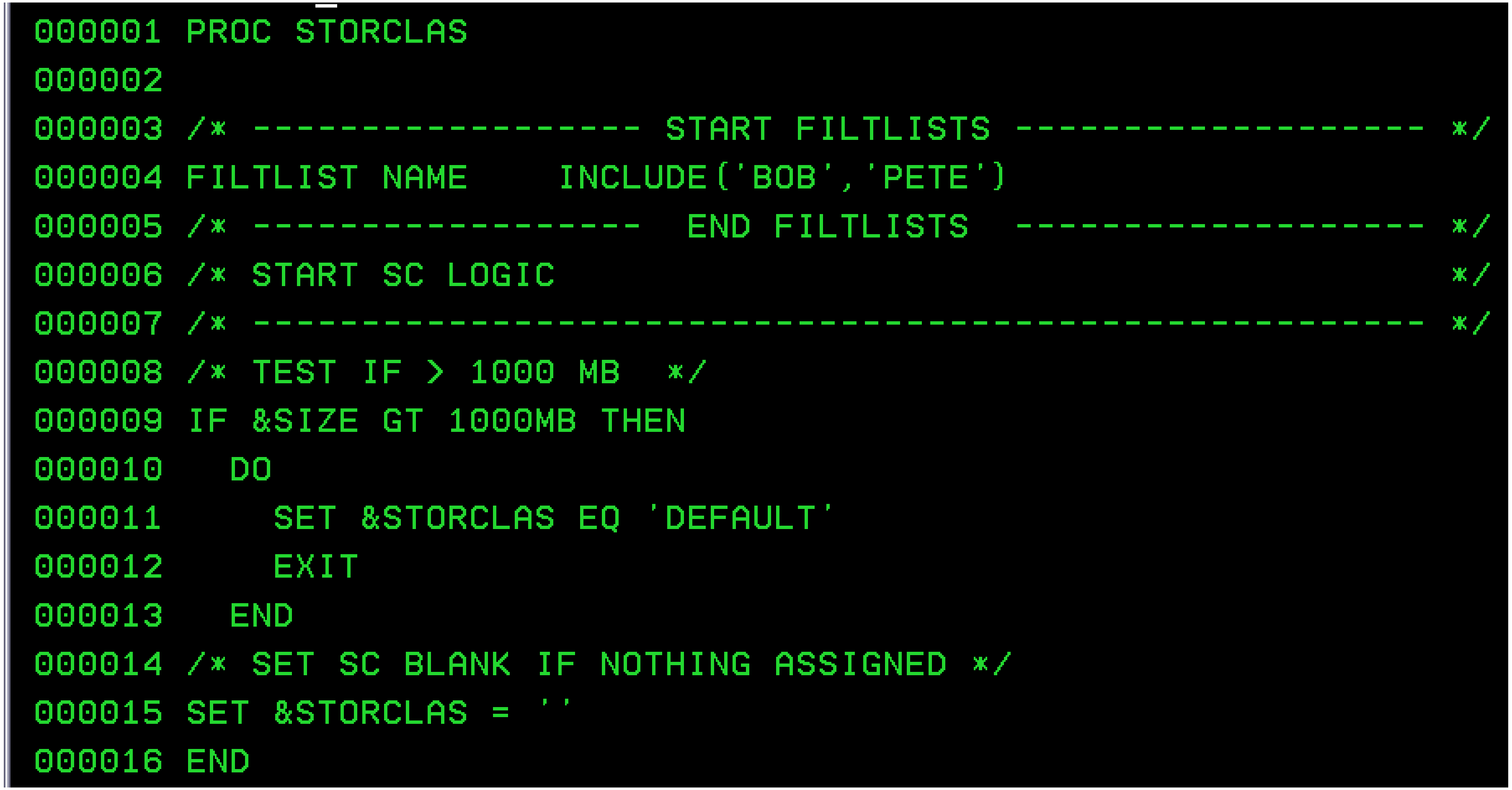


```
PROC STORCLAS

/* ------------------ START FILTLISTS ------------------- */
FILTLIST NAME   _ INCLUDE('BOB','PETE')
/* ------------------  END FILTLISTS  ------------------- */

  SET &STORCLAS = ''
END
****** ***************************** Bottom of Data *******************
```

ACS Walkthrough 2

- If/Then Logic
  - ADD Rule #1 – IF data set is greater than 1000 MB
    - Compare the SIZE to 1000MB
    - If SIZE is larger, set SC to **DEFAULT**
- Translate the ACS routine

IF Example

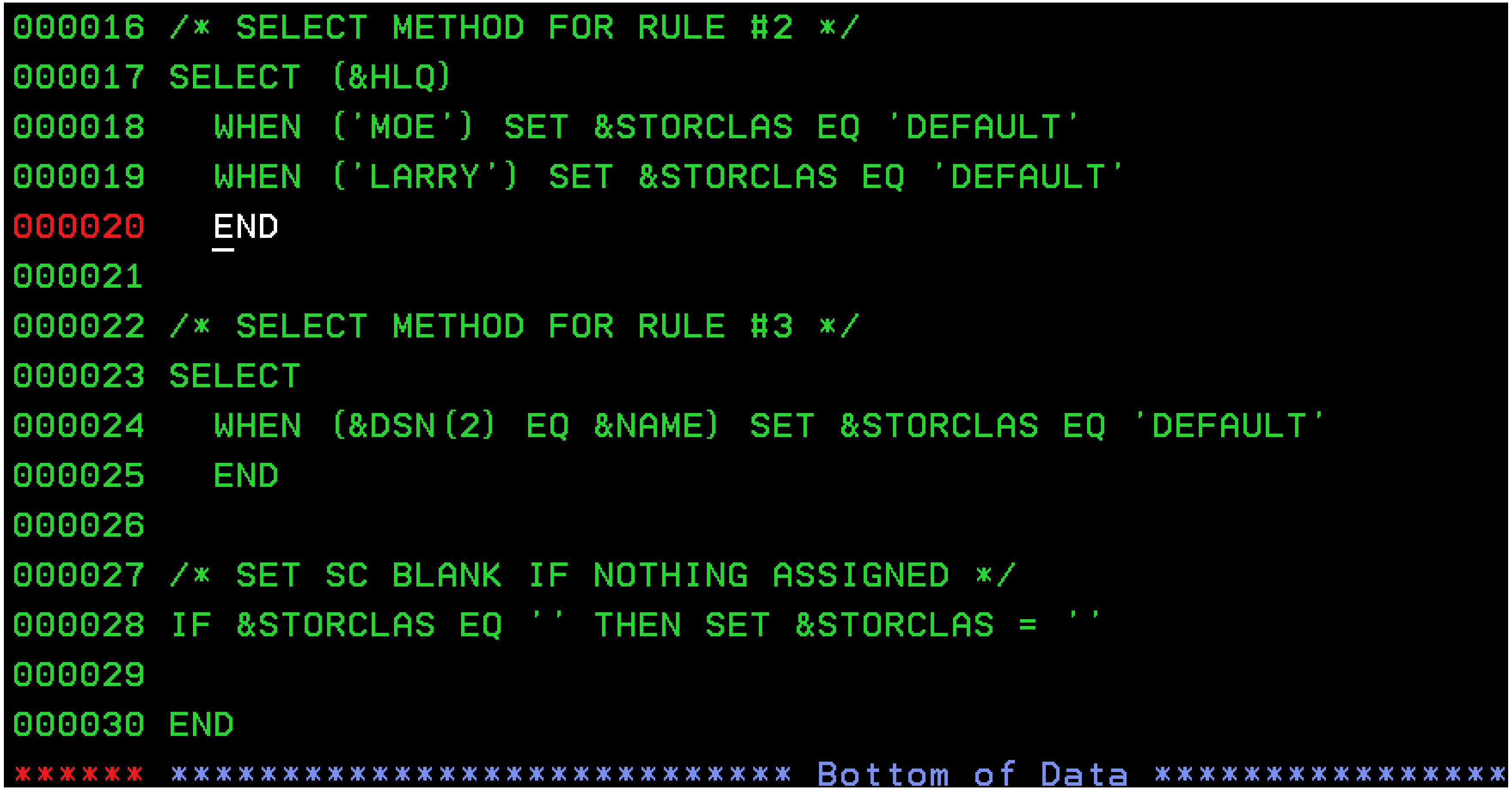


```
PROC STORCLAS

/* ------------------ START FILTLISTS ------------------- */
FILTLIST NAME    INCLUDE('BOB','PETE')
/* ------------------  END FILTLISTS  ------------------- */
/* START SC LOGIC                                          */
/* ------------------------------------------------------- */
/* TEST IF > 1000 MB  */
IF &SIZE GT 1000MB THEN
  DO
    SET &STORCLAS EQ 'DEFAULT'
    EXIT
  END
/* SET SC BLANK IF NOTHING ASSIGNED */
SET &STORCLAS = ''
END
```

ACS Walkthrough 3

- SELECT
  - Add a SELECT statements to implement rules #2 and #3
    - #2 - SELECT on Read Only variable &HLQ
      - *Test for ‘LARRY’ or ‘MOE’*
      - *If it matches, set SC to DEFAULT*
    - #3 - SELECT on Read Only variable &DSN(2)
      - *Test for NAME filter*
      - *If it matches, set the SC to DEFAULT*
- Translate the ACS routine

SELECT Example(s)

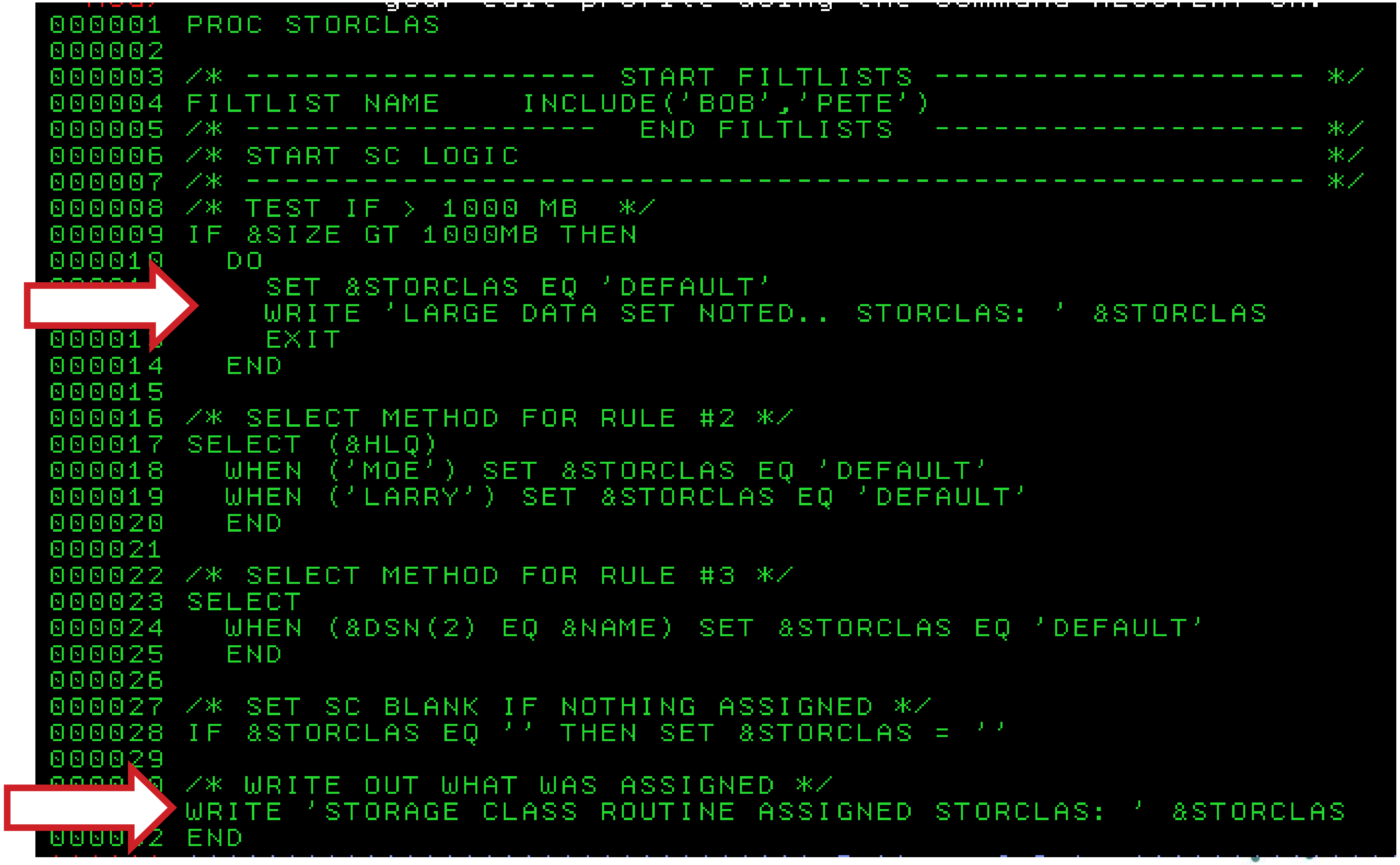


```
/* SELECT METHOD FOR RULE #2 */
SELECT (&HLQ)
  WHEN ('MOE') SET &STORCLAS EQ 'DEFAULT'
  WHEN ('LARRY') SET &STORCLAS EQ 'DEFAULT'
  END

/* SELECT METHOD FOR RULE #3 */
SELECT
  WHEN (&DSN(2) EQ &NAME) SET &STORCLAS EQ 'DEFAULT'
  END

/* SET SC BLANK IF NOTHING ASSIGNED */
IF &STORCLAS EQ '' THEN SET &STORCLAS = ''

END
****** **************************** Bottom of Data ****************
```

ACS Walkthrough 4

- **WRITE**
  - Add a WRITE statement to show which SC assigned
    - WRITE ‘message’
  - **TIP**: try to have a write near every EXIT point
- Translate the ACS routine

WRITE Examples



```
PROC STORCLAS

/* ------------------  START FILTLISTS  ------------------- */
FILTLIST NAME    INCLUDE('BOB','PETE')
/* ------------------   END FILTLISTS   ------------------- */
/* START SC LOGIC                                          */
/* ------------------------------------------------------- */
/* TEST IF > 1000 MB  */
IF &SIZE GT 1000MB THEN
  DO
    SET &STORCLAS EQ 'DEFAULT'
    WRITE 'LARGE DATA SET NOTED.. STORCLAS: ' &STORCLAS
    EXIT
  END

/* SELECT METHOD FOR RULE #2 */
SELECT (&HLQ)
  WHEN ('MOE') SET &STORCLAS EQ 'DEFAULT'
  WHEN ('LARRY') SET &STORCLAS EQ 'DEFAULT'
  END

/* SELECT METHOD FOR RULE #3 */
SELECT
  WHEN (&DSN(2) EQ &NAME) SET &STORCLAS EQ 'DEFAULT'
  END

/* SET SC BLANK IF NOTHING ASSIGNED */
IF &STORCLAS EQ '' THEN SET &STORCLAS = ''

/* WRITE OUT WHAT WAS ASSIGNED */
WRITE 'STORAGE CLASS ROUTINE ASSIGNED STORCLAS: ' &STORCLAS
END
```

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.

Resources

Books

- *DFSMS Storage Administration Reference* (SC26-7402)
- *DFSMS Implementing System-Managed Storage* (SC26-7407)
- *DFSMS Using the Interactive Storage Management Facility* (SC26-7411)

Labs:

- *Next session, 12320 is the LAB. Come try your ACS!*
- *1:30 - 2:30 PM, Union Square 23-24, Fourth Floor*

System z Social Media Channels

- Top Facebook pages related to System z:
  - IBM System z
  - IBM Academic Initiative System z
  - IBM Master the Mainframe Contest
  - IBM Destination z
  - Millennial Mainframer
  - IBM Smarter Computing

- Top LinkedIn groups related to System z:
  - System z Advocates
  - SAP on System z
  - IBM Mainframe- Unofficial Group
  - IBM System z Events
  - Mainframe Experts Network
  - System z Linux
  - Enterprise Systems
  - Mainframe Security Gurus

- Twitter profiles related to System z:
  - IBM System z
  - IBM System z Events
  - IBM DB2 on System z
  - Millennial Mainframer
  - Destination z
  - IBM Smarter Computing

- YouTube accounts related to System z:
  - IBM System z
  - Destination z
  - IBM Smarter Computing

- Top System z blogs to check out:
  - Mainframe Insights
  - Smarter Computing
  - Millennial Mainframer
  - Mainframe & Hybrid Computing
  - The Mainframe Blog
  - Mainframe Watch Belgium
  - Mainframe Update
  - Enterprise Systems Media Blog
  - Dancing Dinosaur
  - DB2 for z/OS
  - IBM Destination z
  - DB2utor

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 products and/or programs 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 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 not infringe 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 another 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.

#SHAREorg

SHARE
Technology · Connections · Results

DFSMS Basics: SMS Configuration and ACS Basics Introduction / Demonstration

Neal Bohling and Tom Reed
IBM

February 6, 2013
Session Number 12323

SHARE in San Francisco 2013

Introduction to ACS Environment

- Specifications in assigned SMS Constructs are applied to the allocation

- You can override specifications of SMS classes and groups on:
  - JCL DD statements
  - Dynamic allocation requests
  - DFSMSdss COPY, RESTORE & CONVERTV
  - DFSMShsm RECALL & RECOVER
  - IDCAMS DEFINE, ALTER & IMPORT
  - OAM STORE, CHANGE & class transition