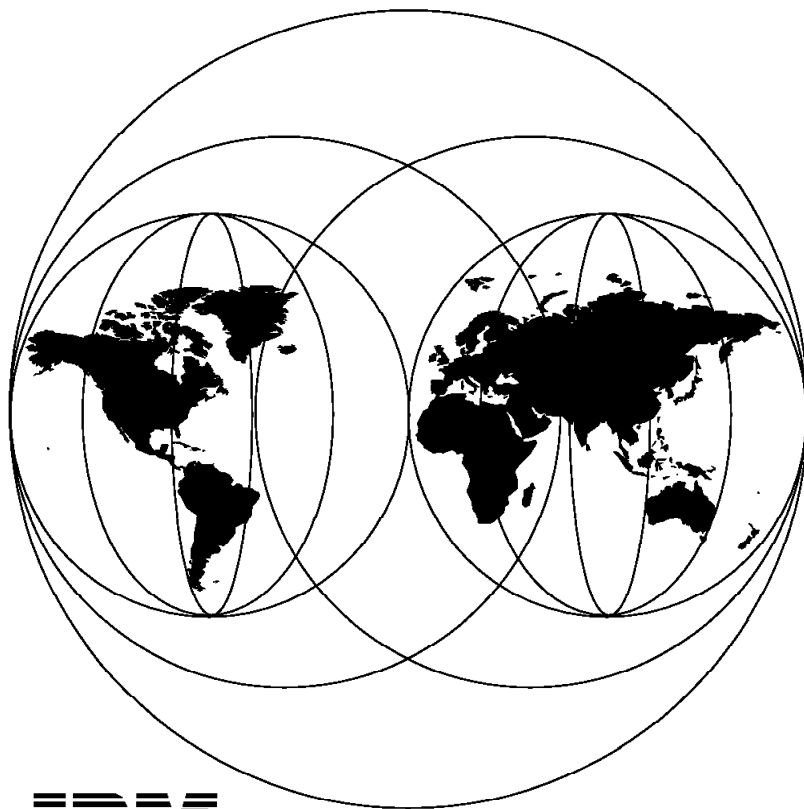


**DFSMS FIT:
Fast Implementation Techniques Forms and Foils**

October 1995



**International Technical Support Organization
San Jose Center**



International Technical Support Organization

SG24-2570-00

**DFSMS FIT:
Fast Implementation Techniques Forms and Foils**

October 1995

Take Note!

Before using this information and the product it supports, be sure to read the general information under "Special Notices" on page vii.

First Edition (October 1995)

This edition applies to Version 1, Release Number 2 of DFSMS/MVS, Program Number 5695-DF1 for use with the MVS/ESA Operating System.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

An ITSO Technical Bulletin Evaluation Form for reader's feedback appears facing Chapter 1. If the form has been removed, comments may be addressed to:

IBM Corporation, International Technical Support Organization
Dept. 471 Building 070B
5600 Cottle Road
San Jose, California 95193-0001

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1995. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Abstract

This document provides the foils and forms used in the DFSMS FIT process to implement DFSMS in two to three weeks. It is intended to be used by an experienced DFSMS implementation person while they lead a new DFSMS customer through implementation using the DFSMS FIT process to produce a simple and good DFSMS implementation.

It is assumed that the customer is positioned to begin DFSMS implementation before starting the DFSMS FIT process. A description of the DFSMS FIT process and the activities that should be completed before the start of DFSMS FIT are provided in SG24-2568, *Getting DFSMS FIT: Fast Implementation Techniques*.

Examples of nine customers that used the DFSMS FIT approach for their initial DFSMS implementation can be found in *DFSMS FIT: Fast Implementation Techniques Installation Examples*, SG24-2569. This document contains documentation of the key output steps of the DFSMS FIT process for each of these customers.

DFSMS FIT: Fast Implementation Techniques Process Guide, SG24-4478, contains text on how to use each of the foils and forms in this document.

Contents

Abstract	iii
Special Notices	vii
Preface	ix
How This Document Is Organized	ix
Related Publications	ix
International Technical Support Organization Publications	ix
Acknowledgments	xi
Chapter 1. DFSMS FIT Parts	1
Introduction	1
Data Classification	1
Storage Class Design	1
Storage Group Design	1
ACS Structure and Coding	1
Storage Class ACS Coding	2
Storage Group ACS Coding	2
Management Class Design	3
Management Class ACS Coding	3
Data Class Design	3
Data Class ACS Coding	3
Phased Implementation	4
Testing	4
Staged Data Conversion	4
Chapter 2. Foils and Forms	5

Special Notices

This document provides the foils and forms used in the DFSMS FIT process to implement DFSMS in two to three weeks. It is intended to be used by an experienced DFSMS implementation person while they lead a new DFSMS customer through implementation using the DFSMS FIT process to produce a simple and good DFSMS implementation.

It is assumed that the customer is positioned to begin DFSMS implementation before starting the DFSMS FIT process. A description of the DFSMS FIT process and the activities that should be completed before the start of DFSMS FIT are provided in SG24-2568, *Getting DFSMS FIT: Fast Implementation Techniques*.

Examples of nine customers that used the DFSMS FIT approach for their initial DFSMS implementation can be found in *DFSMS FIT: Fast Implementation Techniques Installation Examples*, SG24-2569. This document contains documentation of the key output steps of the DFSMS FIT process for each of these customers.

DFSMS FIT: Fast Implementation Techniques Process Guide, SG24-4478, contains text on how to use each of the foils and forms in this document.

The information in this publication is not intended as the specification of any programming interfaces that are provided by DFSMS/MVS. See the PUBLICATIONS section of the IBM Programming Announcement for DFSMS/MVS for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

The following terms, which are denoted by an asterisk (*) in this publication, are trademarks of the International Business Machines Corporation in the United States and/or other countries:

CICS	DB2
DFSMS	DFSMS/MVS
DFSMSdfp	IBM
IMS	MVS/ESA
NaviQuest	RACF
RAMAC	

The following terms are trademarks of other companies:

Windows is a trademark of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Preface

This document provides the foils and forms used in the DFSMS FIT process to implement DFSMS in two to three weeks. It is intended to be used by an experienced DFSMS implementation person while they lead a new DFSMS customer through implementation using the DFSMS FIT process to produce a simple and good DFSMS implementation.

It is assumed that the customer is positioned to begin DFSMS implementation before starting the DFSMS FIT process. A description of the DFSMS FIT process and the activities that should be completed before the start of DFSMS FIT are provided in SG24-2568, *Getting DFSMS FIT: Fast Implementation Techniques*.

Examples of nine customers that used the DFSMS FIT approach for their initial DFSMS implementation can be found in *DFSMS FIT: Fast Implementation Techniques Installation Examples*, SG24-2569. This document contains documentation of the key output steps of the DFSMS FIT process for each of these customers.

DFSMS FIT: Fast Implementation Techniques Process Guide, SG24-4478, contains text on how to use each of the foils and forms in this document.

How This Document Is Organized

The document is organized as follows:

- Chapter 1, "DFSMS FIT Parts" provides an introduction to foils and forms to support each part of the DFSMS FIT process.
- Chapter 2, "Foils and Forms" contains full sized foils and forms for all parts of the DFSMS process.

Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this document.

- *DFSMS/MVS Version 1 Release 2 DFSMSdfp Storage Administration Reference*, SC26-4920
- *MVS/ESA Storage Management Library Implementing System-Managed Storage*, SC26-3123

International Technical Support Organization Publications

- *Getting DFSMS FIT: Fast Implementation Techniques*, SG24-2568 (available at a later date)
- *DFSMS FIT: Fast Implementation Techniques Installation Examples*, SG24-2569 (available at a later date)
- *DFSMS FIT: Fast Implementation Techniques Process Guide*, SG24-4478 (available at a later date)
- *DFSMS Implementation Primer Series: Writing ACS Routines*, GG24-3403-01

A complete list of International Technical Support Organization publications, with a brief description of each, may be found in:

Bibliography of International Technical Support Organization Technical Bulletins, GG24-3070.

To get a catalog of ITSO technical publications (known as “redbooks”), VNET users may type:

TOOLS SENDTO WTSCPOK TOOLS REDBOOKS GET REDBOOKS CATALOG

How to Order ITSO Technical Publications

IBM employees in the USA may order ITSO books and CD-ROMs using PUBORDER. Customers in the USA may order by calling 1-800-879-2755 or by faxing 1-800-284-4721. Visa and Master Cards are accepted. Outside the USA, customers should contact their local IBM office.

Customers may order hardcopy ITSO books individually or in customized sets, called GBOFs, which relate to specific functions of interest. IBM employees and customers may also order ITSO books in online format on CD-ROM collections, which contain books on a variety of products.

Acknowledgments

This project was designed and managed by:

Dale Freeman
International Technical Support Organization, San Jose Center

The author of this document is:

Dale Freeman
IBM International Technical Support Organization, San Jose

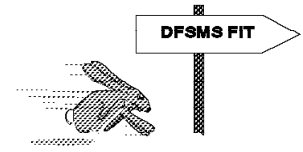
Thanks to the following people for the invaluable advice and guidance provided in the production of this document:

Dave Canan, IBM Western Area
Maggie Cutler, Technical Editor
Ron Ratcliffe, IBM San Jose Storage Services

International Technical Support Organization, San Jose Center

October 1995

Chapter 1. DFSMS FIT Parts



This chapter provides you with an overview of the parts of the DFSMS FIT process. Support for each part of the process is provided in the *DFSMS FIT: Fast Implementation Techniques Process Guide*.

Introduction

This part provides you with an overview of the DFSMS FIT process.

Use the five foils in this part to introduce the DFSMS FIT process. You should make sure that all people involved in DFSMS FIT understand the purpose of each step in the process.

Data Classification

This part covers the initial DFSMS FIT implementation step, data classification. Data classification is the process of determining the data types that your installation wants to have SMS-managed. It also is the process of identifying different detailed data subtypes that require unique data services from DFSMS.

Storage Class Design

This part covers the storage class design step of the DFSMS FIT process. During this step you determine how many storage classes are required to support the data subtypes defined during data classification. After you have identified the number of storage classes required, you evaluate each storage class and specify its parameters.

Storage Group Design

This part covers the storage group design process. This step of the DFSMS FIT approach is similar to the storage class design step. The number of storage groups needed is determined through a set of questions for each box in the data classification pictures. After you determine the number of storage groups, you define their parameters and update the data classification pictures.

ACS Structure and Coding

This part covers general ACS coding. It is an introduction to the subsequent parts where ACS routines are coded. This part provides the ACS coding techniques that you should use for all ACS routines.

Use the foils in this part to ensure that your installation understands good coding and documentation techniques. Cover this part quickly, particularly if the installation's storage administrators are experienced in good coding techniques.

Storage Class ACS Coding

This part covers storage class ACS coding. The storage class ACS code for your installation is built from code fragments supplied as part of the DFSMS FIT process.

You tailor each code fragment according to the data classification pictures, storage class design results, and the installation's data set naming standards. You should already have developed most of the key information for ACS routines during the updates to the data classification pictures after the storage class design step.

The code fragments are divided into two types: mandatory and optional. You add the optional code fragments based on the data types and functions you plan to use in your initial implementation.

You should move the FILTLISTs in the code fragments to the top of the ACS routines before you use them. The FILTLISTs are currently placed at the location where they are referenced so that you can easily understand the code fragments.

Storage Group ACS Coding

This part covers storage group ACS coding. The storage group ACS code for your installation is built from code fragments supplied as part of the DFSMS FIT process.

You tailor each code fragment according to the data classification pictures, storage group design results, and the installation's data set naming standards. You should have developed most of the key information for ACS routines during the updates to the data classification pictures after the storage group design step.

The code fragments are divided into two types: mandatory and optional. You add the optional code fragments on the basis of the data types and functions you plan to use in your initial implementation.

You should move the FILTLISTs in the code fragments to the top of the ACS routines before you use them. The FILTLISTs are currently placed at the location where they are referenced so that you can more easily understand the code fragments.

Many of the FILTLISTs used in the storage group ACS routine may be exact duplicates of FILTLISTs needed in the storage class ACS routine. Define and maintain in one place the FILTLISTs that are used in multiple ACS routines. Build and maintain the FILTLISTs in a PDS member. Before translating an ACS routine, copy the entire set of FILTLISTs into the routine.

If you use the NaviQuest tool as part of the DFSMS FIT process, you will find the COPYFILT function useful for controlling the use of a common set of FILTLISTs.

Management Class Design

This part covers the management class design step of the DFSMS FIT process. This design step is similar to the storage class and storage group design steps. A question set is used to determine the number of management classes required to support the design. Next the management classes are named and their parameters defined. Finally the data classification pictures are updated with the management class design information.

Management Class ACS Coding

This part covers management class ACS coding. The management class ACS code for your installation is built from code fragments supplied as part of the DFSMS FIT process.

You tailor each code fragment according to the data classification pictures, management class design results, and the installation's data set naming standards. You should have developed most of the key information for ACS routines during the updates to the data classification pictures after the management class design step.

The code fragments are divided into two types: mandatory and optional. You add the optional code fragments on the basis of the data types and functions you plan to use in your initial implementation.

You should move the FILTLISTs in the code fragments to the top of the ACS routines before you use them. The FILTLISTs are currently placed at the location where they are referenced so that you can easily understand the code fragments.

Many of the FILTLISTs used in the management class ACS routine may be exact duplicates of FILTLISTs needed in the other ACS routines. Define and maintain the FILTLISTs that are used in multiple ACS routines in one place. Build and maintain the FILTLISTs in a PDS member. Before translating an ACS routine, copy the entire set of FILTLISTs into the routine.

If you use the NaviQuest tool as part of the DFSMS FIT process, you will find the COPYFILT function useful to control the use of a common set of FILTLISTs.

Data Class Design

This part covers the data class design step of the DFSMS FIT process. During this step you determine the number of data classes required to support the data subtypes defined during data classification. After you have identified the number of data classes required, you evaluate each data class and specify its parameters.

Data Class ACS Coding

This part covers data class ACS coding. The data class ACS code for your installation is built from code fragments supplied as part of the DFSMS FIT process.

You tailor each code fragment according to the data classification pictures, data class design results, and the installation's data set naming standards. You should have developed most of the key information for ACS routines during the updates to the data classification pictures after the data class design step.

The code fragments are divided into two types: mandatory and optional. You add the optional code fragments on the basis of the data types and functions you plan to use in your initial implementation.

You should move the FILTLISTs in the code fragments to the top of the ACS routines before you use them. The FILTLISTs are currently placed at the location where they are referenced so that you can easily understand the code fragments.

Many of the FILTLISTs used in the data class ACS routine may be exact duplicates of FILTLISTs needed in the other ACS routine. Define and maintain the FILTLISTs that are used in multiple ACS routines in one place. Build and maintain the FILTLISTs in a PDS member. Before translating an ACS routine, copy the entire set of FILTLISTs into the routine.

If you use the NaviQuest tool as part of the DFSMS FIT process, you will find the COPYFILT function useful to control the use of a common set of FILTLISTs.

Phased Implementation

This part covers the phased implementation design step of the DFSMS FIT process. During this step you determine how many phased implementation steps, and their order, that are required to support the data subtypes defined during data classification.

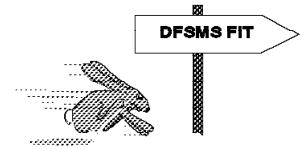
Testing

This part covers the testing step of the DFSMS FIT process. During this step you test the assignment logic in the ACS routines against actual installation data sets. The DFSMS FIT process assumes that you use the NaviQuest product to automate DFSMS testing. If you elect to not use NaviQuest you will need to modify the DFSMS FIT approach to perform each of the NaviQuest functions in a non-automated fashion using the standard DFSMS testing capability in ISMF.

Staged Data Conversion

This part covers the staged data conversion step of the DFSMS FIT process. During this step you move data to SMS-management in the defined phases.

Chapter 2. Foils and Forms



This chapter contains all the full sized foils and forms to use during the DFSMS FIT process. Each foil or form is described in the *DFSMS FIT: Fast Implementation Techniques Process Guide*.

Introduction

Overview of DFSMS FIT

- Implements DFSMS in two to three weeks
- Is repeatable and successful
- Allows customers to tailor DFSMS to their needs, data, and naming standards
- Uses a methodology that supports new functions and validates your design

General DFSMS FIT Process

- Driven by question sets
 - Facilitates decisions and resolves problems quickly
 - Limits alternatives and keeps design simple and valid
 - Documents design in tables and pictures
- Driven by sample installations
 - Uses case study installations in each step
 - Contains nine additional installation examples
- Includes sample jobs, code samples, and procedures
- Uses NaviQuest for testing
 - Should be installed during initial week of DFSMS FIT

DFSMS FIT Process

- Introduction
- Data classification
- Storage class design
- Storage group design
- Storage class ACS coding
- Storage group ACS coding
- Management class design
- Management class ACS coding
- Data class design
- Data class ACS coding
- Phased implementation design
- Testing
- Staged data conversion

Prerequisite Activities

- Identify the DFSMS implementation leader
- Identify other DFSMS implementation project members
- Schedule facilities
- Obtain seed DASD
- Verify system maintenance levels
- Protect the SMS environment
- Define DFSMS startup parameters
- Create and activate the minimal configuration
- Identify unsupported data sets
- Collect current storage management documentation

Support Materials

- Documentation

- *DFSMS FIT: Fast Implementation Techniques Process Guide*, SG24-4478
- *Getting DFSMS FIT: Fast Implementation Techniques*, SG24-2568
- *DFSMS FIT: Fast Implementation Techniques Installation Examples*, SG24-2569
- *DFSMS FIT: Fast Implementation Techniques Forms and Foils*, SG24-2570

- Diskette

Data Classification

Data Classification

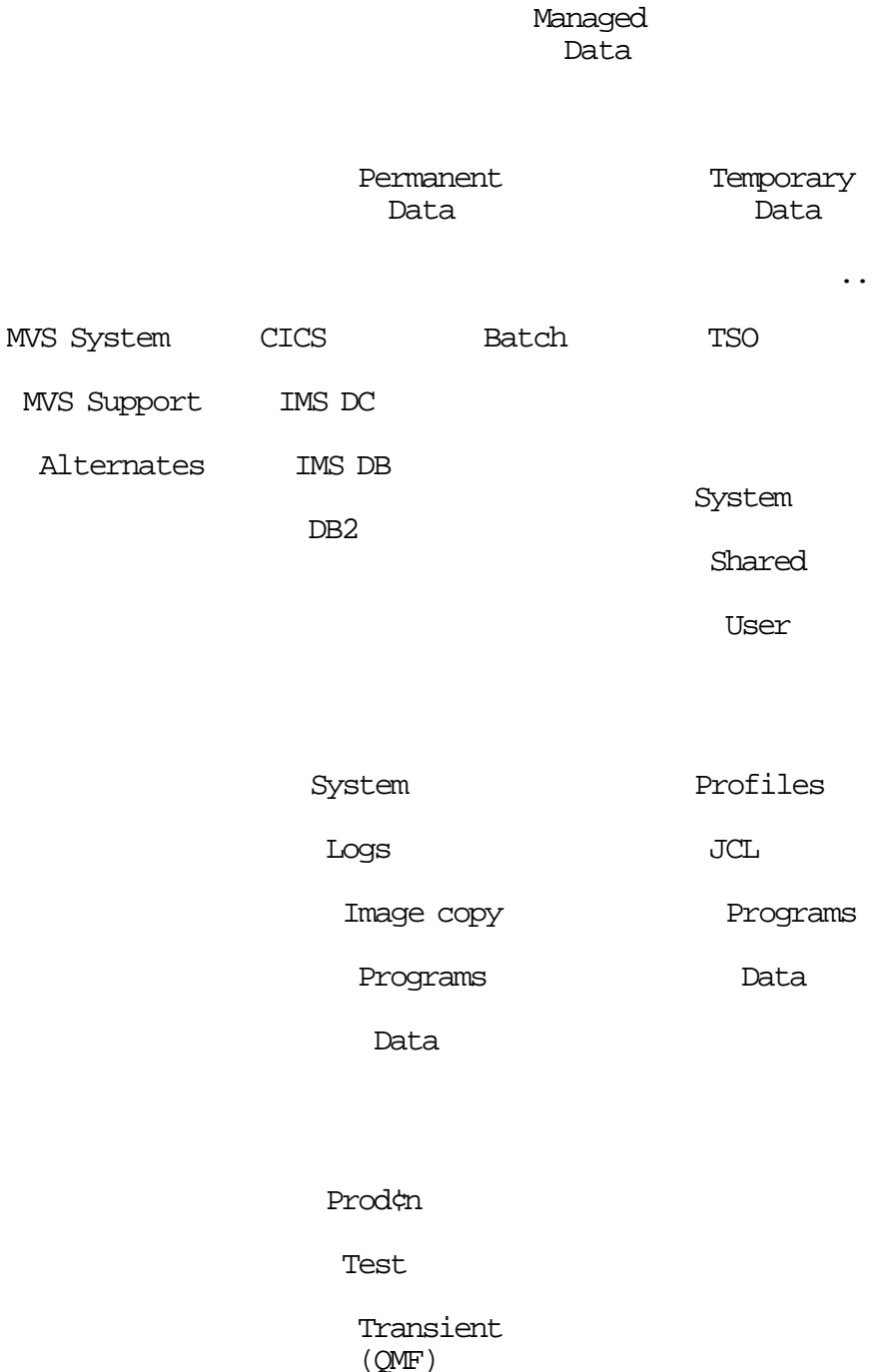
Topics

- Classification approach
- Data classification questions
- Sample installation data classification

Data Classification Value

- Highest value design activity
- Enables you to
 - Understand data
 - Determine exception handling
 - Validate storage and management class services
 - Communicate service assignment to storage users
 - Validate and document ACS logic
 - Drive structured testing

General Data Classification Approach



Alternative Data Classification Approach

- Start DFSMS implementation with a single data type
 - Quickest implementation
 - Build DFSMS experience
 - Minimal risk and limited user visibility
- Start DFSMS implementation with a single function or DFSMS tower
 - Single or limited number of applications
 - High return function - sequential striping, concurrent copy, data compression
 - Quick implementation of highest value function

Data Classification Process

- Driven by question set
 - Based on data types common for all installations
 - Question set for each common data type
 - Questions drive tailored data types and classification
- Driven by other installation examples
 - One example of an installation data classification shown
 - Many additional installation data classification samples in *DFSMS FIT Installation Examples, SG24-2569*

Data Classification Questions - Work

1. Do you want to consider using VIO to reduce overhead for selected temporary data sets?
2. Do you include short duration permanent data sets with system temporary data sets?
3. What do you call this data type: Work, Temp, Scratch?

Work Data Classification

DSTYPE=TEMP
DD =SORTWK*

DSTYPE=TEMP
DD=SORTWK*

System
Temporary

Sortwork
System Temp.

Permanent
Short Duration

Data Classification Questions - TSO

1. Do you want to consider TSO data separately or include it with test data?
2. If separate TSO, how is it identified?
3. Do you want to handle library data differently?
4. Do you want to handle large data sets differently?

TSO Data Classification

TSO

DSN=list

DSN=list

Data Used
at LOGON

LIST, LOG
and Other
Temporary

Other
TSO Data

Data Classification Questions - Test

1. Do you want to consider TSO data separately or include it with test data?
2. Do you want to consider test data separately or include it with production data?
3. Do you want to handle batch and online test data differently?
4. Do you want to handle online test data (VSAM, DB2, IMS) differently?
5. Do you want to handle test libraries differently?
6. Do you want to handle test GDGs differently?
7. Are there work or short-duration test data sets and do you want to handle them differently?
8. Do you want to handle large data sets differently?
9. Do you have quality assurance or acceptance test data and do you want to separate it or include it with test data?

Test Data Classification

Test

Acceptance Test Data Classification

Accept. Test
or Quality
Assurance

Data Classification Questions - Production

1. Do you want to consider test data separately or include it with production data?
2. Do you want to consider online production data separately or include it with batch production data?
3. Do you want to handle online production data (VSAM, DB2, IMS) differently?
4. Do you want to handle production libraries differently?
5. Do you want to handle production GDGs differently?
6. Are there work or short-duration production data sets and do you want to handle them differently?
7. Do you want to handle large data sets differently?

Batch Production Data Classification

Production
Batch

Online Production Data Classification

Production
Online

Data Classification Questions - Other

1. Do you have non-MVS-collected data that you want to SMS-manage and consider separately?
2. Do you have report package data that you want to SMS-manage and consider separately?
3. Do you have tape data that you want to SMS-manage?
4. Do you have optical data that you want to SMS-manage?
5. Do you have an application that you want to use sequential data striping?
6. Do you have an application that you want to use concurrent copy?
7. Do you have an application that you want to use data compression?
8. Do you have an application that you want to use other DFSMS function towers?

Other Data Classification

Data Classification Questions - Exceptions

1. Do you need JCL volume placement of selected data sets by a controlled number of users?
2. Do you need a JCL override for cache unfriendly data instead of dynamic cache management enhanced?
3. Do you need a JCL override for critical performance data instead of dynamic cache management enhanced?
4. Do you need a JCL override for management class deletion?
5. Do you need a JCL override for management class migration?
6. Do you need a JCL override for management class backup?
7. Do you need a JCL override for management class GDG handling?

Exception Data Classification

Exceptions

Non SMS

Data Classification Questions - Non-SMS

1. Is work data to be nonmanaged?
2. Is TSO data to be nonmanaged?
3. Is test data to be nonmanaged?
4. Is batch production data to be nonmanaged?
5. Is online production data to be nonmanaged?
6. Is system data to be nonmanaged?
7. Are there other DASD data types to be nonmanaged?
8. Is tape data to be nonmanaged?
9. Is optical data to be nonmanaged?

Non-SMS Data Classification

Nonmanaged

HLQ=SYS%

System

Sample - SMS-Managed Data Types

SMS Data Types

Work

TSO

Test

Batch
Production

Sample - Work Data Questions

1. Do you want to consider using VIO to reduce overhead for selected temporary data sets?

NO

2. Do you include short-duration permanent data sets with system temporary data sets?

NO

3. What do you call this data type: Work, Temp, Scratch?

WORK

Sample - Work Data

WKL

DSTYPE=TEMP

Work

Sample - TSO Data Questions

1. Do you want to consider TSO data separately or include it with test data?

SEPARATE

2. If separate TSO, how is it identified?

HLQ PATTERN

3. Do you want to handle library data differently?

NO

4. Do you want to handle large data sets differently?

NO

Sample - TSO Data

HLQ=%%T%%

TSO

OTHERWISE

DSN=list

DSN=list

TS2

Normal
Data

TS3

LIST, LOG,
ISPF*TEMP.CNLT
type data

TS4

Used at
LOGON

Sample - Test Data Questions

1. Do you want to consider TSO data separately or include it with test data? SEPARATE
2. Do you want to consider test data separately or include it with production data? SEPARATE
3. Do you want to handle batch and online test data differently?

NOT AT THIS TIME

4. Do you want to handle online test data (VSAM, DB2, IMS) differently? NO
5. Do you want to handle test libraries differently? NO
6. Do you want to handle test GDGs differently? YES
7. Are there work or short-duration test data sets and do you want to handle them differently? YES
8. Do you want to handle large data sets differently? NO
9. Do you have quality assurance or acceptance test data and do you want to separate it or include it with test data? NO

Sample - Test Data

```
Test          HLQ=application list
              2LQ=T*

OTHERWISE     2LQ=T%%TEMP*      DSTYPE=GDS

T5            T6            T7
Normal       One Day       GDGs
```

Sample - Production Data Questions

1. Do you want to consider test data separately or include it with production data?

SEPARATE

2. Do you want to consider online production data separately or include it with batch production data?

SEPARATE

3. Do you want to handle online production data (VSAM, DB2, IMS) differently?

N/A

4. Do you want to handle production libraries differently?

NO

5. Do you want to handle production GDGs differently?

YES

6. Are there work or short-duration production data sets and do you want to handle them differently?

YES

7. Do you want to handle large data sets differently?

NO

Sample - Batch Production Data

Batch Production HLQ=application list
2LQ=P*

2LQ=P%%MF* OTHERWISE DSN=list

BP8 Working Files BP9 Master Files BP10 Critical Performance Files

BP11 GDGs DSTYPE=GDS

Sample - Other Data Questions

1. Do you have non-MVS collected data that you want to SMS-manage and consider separately? NOT AT THIS TIME
2. Do you have report package data that you want to SMS-manage and consider separately? NOT AT THIS TIME
3. Do you have tape data that you want to SMS-manage? NOT AT THIS TIME
4. Do you have optical data that you want to SMS-manage? NOT AT THIS TIME
5. Do you have an application that you want to use sequential data striping? NOT AT THIS TIME
6. Do you have an application that you want to use concurrent copy? NOT AT THIS TIME
7. Do you have an application that you want to use data compression? NOT AT THIS TIME
8. Do you have an application that you want to use other DFSMS function towers? NOT AT THIS TIME

Sample - Exception Data Questions

1. Do you need JCL volume placement of selected data sets by a controlled number of users? YES
2. Do you need a JCL override for cache unfriendly data instead of dynamic cache management enhanced? NO
3. Do you need a JCL override for critical performance data instead of dynamic cache management enhanced? YES
4. Do you need a JCL override for management class deletion? NO
5. Do you need a JCL override for management class migration? NO
6. Do you need a JCL override for management class backup? NO
7. Do you need a JCL override for management class GDG handling? NO

Sample - Exceptions

Exceptions

JCL	JCL	JCL
EX12 Normal Performance Hand Placed	EX13 Critical Performance Hand Placed	EX14 Non SMS

Sample - Non-SMS Data Questions

1. Is work data to be nonmanaged? NO
2. Is TSO data to be nonmanaged? NO
3. Is test data to be nonmanaged? NO
4. Is batch production data to be nonmanaged? NO
5. Is online production data to be nonmanaged? YES
6. Is system data to be nonmanaged? YES
7. Are there other DASD data types to be nonmanaged? NO
8. Is tape data to be nonmanaged? YES
9. Is optical data to be nonmanaged? YES

Sample - Non-SMS Data

Nonmanaged

HLQ=SYS%

System

HLQ=application list

2LQ=P*

3LQ=IMS*

Online
Production
Database

Storage Class Design

Storage Class Design

Topics

- Design considerations
- Storage class questions
- Definition of storage classes
- Update of data classification

Storage Class Benefits

- Selection of data to be SMS-managed
- Logical and physical separation
- Automatic policy-driven data placement
- Automatic performance criteria
 - MSR and BIAS for direct and sequential access
 - Data set placement and dynamic cache management
 - PDSE use of memory and 3990-3 functions
 - Number of stripes for sequential data striping
- Automatic availability criteria
 - Use of dual copy or RAMAC RAID 5
 - Criteria to select volumes to support concurrent copy
- Direct allocations to system-managed volumes

Storage Class Considerations

- MSR of 999 = never cache
- MSR of 5, bias of W = always cache
- MSR of 25 = maybe cache
- Effect of ACCESSIBILITY with DFSMS/MVS
 - Default of PREFERRED eliminates volumes
 - Use of STANDARD eliminates volumes
 - NOPREF is best choice if concurrent copy is not used
- Method of space allocation for multivolume data sets

Storage Class Considerations ...

- Default class and group structure - two approaches
 - STORCLAS and MGMTCLAS by service type - SCFAST
 - STORCLAS and MGMTCLAS by data type - SCPROD
- Conflicts in IBM documentation and recommendations
- Flexibility and maintenance trade-off
- Installations moving from data type to service type
- Think it through and make a conscious choice

Storage Class Considerations ...

- Service type approach - allows one service and ACS routine to change independently of other services and ACS routine logic

```
FILTLIST &TEST_DATA_DSN INCLUDE (T*)
WHEN &DSN(2) EQ &TEST_DATA_DSN
DO
    SET &STORCLAS = ¢SCNORM¢
    EXIT
END
```

```
FILTLIST &TEST_DATA_DSN INCLUDE (T*)
WHEN &DSN(2) EQ &TEST_DATA_DSN
DO
    SET &STORGRP = ¢SGTEST90¢
    EXIT
END
```

- Service type approach - requires duplication of FILTLISTs and assignment logic

Storage Class Considerations ...

- Data type approach - used to make STORGRP assignment easy

```
WHEN &STORCLAS EQ ꞑSCTESTꞑ
DO
    SET &STORGRP = ꞑSGTEST90ꞑ
    EXIT
END
```

- Data type approach - no DFSMS logical and physical independence

Exception Handling

- Three methods of exception handling
 - ACS logic and FILTLISTS
 - JCL
 - RACF's DFP fields

- Two methods of control
 - ACS logic and FILTLISTS
 - RACF's DFP fields

- DFSMS FIT does not use RACF

Storage Class Recommendations

- Minimize the number of storage classes
- Restrict use of GUARANTEED SPACE
 - Allow access by authorized users
 - Recall restricted to specific volumes
- Use SUSTAINED DATA RATE for sequential striping
- Use CF DIRECT WEIGHT, CF SEQUENTIAL WEIGHT, and CACHE SET NAME for support of VSAM RLS
- Use NOPREF before concurrent copy
- Protect storage classes that provide special services
- Do not manage DFHSM-owned volumes
- Provide an override to SMS management

Storage Class Design Questions

1. What level of performance do you want?

- Maybe cache (DCME)
- Never cache
- Always cache
- Sequential data striping, single stripe
- Sequential data striping, multiple stripes

2. What level of availability do you want?

- Standard
- Dual copy or RAMAC RAID 5
- Concurrent copy

3. If concurrent copy is used, what type of controllers?

- Mixed (relative to extended platform) controllers

4. What are the required exception controls?

- GUARANTEED SPACE
- Override to make data set non-SMS-managed
- VIO and sort work data sets

Storage Class Design Table

Data Type									
Classification box									
Performance									
Never cache									
Maybe cache									
Always cache									
Single stripe									
Multiple stripes									
Availability									
Dual copy/RAID 5									
Concurrent copy									
Mixed controllers extended platform									
Exception Controls									
Guaranteed space									
Non-SMS									
Non-VIO									
Storage Class ID									

Sample Storage Class Design

Data Type	WK	TS	TS	TS	T	T	T		
Classification box	1	2	3	4	5	6	7		
Performance									
Never cache									
Maybe cache		X	X	X	X	X	X		
Always cache	X								
Single stripe									
Multiple stripes									
Availability									
Dual copy/RAID 5									
Concurrent copy									
Mixed controllers extended platform									
Exception Controls									
Guaranteed space									
Non-SMS									
Non-VIO									
Storage Class ID	1	2	2	2	2	2	2		

Sample Storage Class Design ...

Data Type	B P	B P	B P	B P	EX	EX	EX		
Classification box	8	9	10	11	12	13	14		
Performance									
Never cache									
Maybe cache	X	X		X	X				
Always cache			X			X			
Single stripe									
Multiple stripes									
Availability									
Dual copy/RAID 5									
Concurrent copy									
Mixed controllers extended platform									
Exception Controls									
Guaranteed space					X	X			
Non-SMS							X		
Non-VIO									
Storage Class ID	2	2	1	2	3	4	5		

Storage Class Naming

1.

2.

3.

4.

5.

6.

7.

8.

9.

Sample Storage Class Naming

1. SCFAST
2. SCNORMAL
3. SCSPECN
4. SCSPECF
5. SCNOSMS

Storage Class Protection Design

Storage Class	JCL Use	Protected	By What

Sample Storage Class Protection

Storage Class	JCL Use	Protected	By What
SCFAST	No		
SCNORMAL	No		
SCSPECN	Yes	Yes	ACS
SCSPECF	Yes	Yes	ACS
SCNOSMS	Yes	Yes	ACS

Storage Class Parameters

STORCLAS NAME	DIR RESP MSEC	DIR BIAS	SEQ RESP MSEC	SEQ BIAS	AVAIL- ABILITY	GUARANTEED SPACE

Storage Class Parameters ...

STORCLAS NAME	...	GUARANTEED SYNC WRITE	INIT ACC RESPONSE	ACCESS- IBILITY	SUSTAINED DATA RATE

Storage Class Parameters ...

STORCLAS NAME	CF CACHE SET NAME	CF SEQUENTIAL WEIGHT	CF DIRECT WEIGHT

Sample Storage Class Parameters

STORCLAS NAME	DIR RESP MSEC	DIR BIAS	SEQ RESP MSEC	SEQ BIAS	AVAIL- ABILITY	GUARANTEED SPACE
SCFAST	5	W	5	W	STANDARD	N
SCNORMAL	25		25		STANDARD	N
SCSPECN	25		25		STANDARD	Y
SCSPECF	5	W	5	W	STANDARD	Y
SCNOSMS						

Sample Storage Class Parameters ...

STORCLAS NAME	...	GUARANTEED SYNC WRITE	INIT ACC RESPONSE	ACCESS- IBILITY	SUSTAINED DATA RATE
SCFAST				NOPREF	
SCNORMAL				NOPREF	
SCSPECN				NOPREF	
SCSPECF				NOPREF	
SCNOSMS					

Sample - Work Data

WK1

DSTYPE=TEMP

Work

SC=SCFAST

Sample - TSO Data

HLQ=%%T%%

TSO

OTHERWISE

DSN=list

DSN=list

TS2

Normal
Data

TS3

LIST, LOG,
ISPF*TEMP.CNTL
type data

TS4

Used at
LOGON

SC=SCNORMAL

SC=SCNORMAL

SC=SCNORMAL

Sample - Test Data

```
Test                HLQ=application list
                   2LQ=T*

                   OTHERWISE                2LQ=T%%TEMP*                DSTYPE=GDS

T5                  T6                  T7
Normal              One Day              GDGs

SC=SCNORMAL        SC=SCNORMAL        SC=SCNORMAL
```

Sample - Batch Production Data

Batch HLQ=application list
Production 2LQ=P*

2LQ=P%%MF* OTHERWISE DSN=list

BP8 Working Files	BP9 Master Files	BP10 Critical Performance Files
-------------------------	------------------------	--

SC=SCNORMAL SC=SCNORMAL SC=SCFAST

BP11
GDGs DSTYPE=GDS

SC=SCNORMAL

Sample - Exceptions

Exceptions

	DSN=list and JCL	DSN=list and JCL	JCL STORCLAS=SCNOSMS
EX12	Normal Performance Hand Placed	EX13 Critical Performance Hand Placed	EX14 Non SMS
	SC=SCSPECN	SC=SCSPECF	SC=☐☐

Storage Group Design

Storage Group Design

Topics

- Design considerations
- Storage group questions
- Definition of storage groups
- Update of data classification

Pool Storage Group Benefits

- Formalized implementation of pooling
 - Volume space thresholds
 - Automatic dump, backup, and migration
- Isolation
 - Business reasons
 - Physical security
 - Performance or availability
- Allow for both logical and physical aspects of pooling
 - Add and use volumes without disruption
 - Facilitate device installation and migration
 - Reblocking facilities

VIO and Dummy Storage Group Benefits

VIO

- MAXSIZE controls which data sets can use VIO

Dummy

- JCL conversion aid for data sets allocated using specific volume serial numbers
- JCL volume serial numbers that you see will no longer be what you get

Storage Group Recommendations

- Evaluate physical and logical DASD connectivity
- Consolidate pools as appropriate
 - A single pool is a theoretical ideal
 - It is unachievable in practice; availability, performance, isolation, large data sets
- May want to group 3390-9s or RAMACs
- Use AUTO MIGRATE = I for storage groups with TMM data
- Not all ACS variables are available to storage group ACS routine

Storage Group Design Questions

1. Are volumes shared across systems?

2. Which device types are you using for data storage?
 - 3380s
 - 3390s
 - 3390-9s
 - RAMACs

3. Which level of data isolation is required?
 - Isolate data type
 - Shared with other data types

4. Is VIO to be used?

5. Do you want to separate data sets by size?

Storage Group Design Table

Data Type									
Classification box									
Volume Sharing									
Shared									
Dedicated system									
Device Geometry									
3380s									
3390s									
3390-9s									
RAMACs									
Isolated Pool									
VIO Mapping									
VIO pool									
Data Set Size									
Large pool									
Storage Group ID									

Sample Storage Group Answers

Data Type	WK	TS	TS	TS	T	T	T	
Classification box	1	2	3	4	5	6	7	
Volume Sharing								
Shared	X	X	X	X	X	X	X	
Dedicated system								
Device Geometry								
3380s								
3390s	X	X	X	X	X	X	X	
3390-9s								
RAMACs								
Isolated Pool								
WORK	X							
TSO		X	X	X	X	X	X	
VIO Mapping								
VIO pool								
Data Set Size								
Large pool								
Storage Group ID	1	2	2	2	2	2	2	

Sample Storage Group Answers ...

Data Type	B P	B P	B P	B P	EX	EX	EX	
Classification box	8	9	10	11	12	13	14	
Volume Sharing								
Shared	X	X	X	X	X	X	X	
Dedicated system								
Device Geometry								
3380s								
3390s	X	X	X	X	X	X	X	
3390-9s								
RAMACs								
Isolated Pool								
WORK					N	N	N	
TSO					Y	Y	Y	
PRIME	X	X	X	X	Y	Y	Y	
VIO Mapping								
VIO pool								
Dataset Size								
Large pool								
Storage Group ID	3	3	3	3	N/A	N/A	N/A	

Storage Group Naming

1.

2.

3.

4.

5.

6.

7.

8.

9.

Sample Storage Group Naming

1. SGWORK

2. SGTSO

3. SGPRIME

Storage Group Parameter Design

STORGRP NAME	SG TYPE	VIO MAXSIZE	VIO UNIT	AUTO MIGRATE	MIGRATE SYSTEM	AUTO BACK

Storage Group Parameter Design ...

STORGRP NAME	BACKUP SYSTEM	AUTO DUMP	DUMP SYSTEM	MIGR HIGH	MIGR LOW	...

Storage Group Parameter Design ...

STORGRP NAME	DUMP CLASS	DUMP CLASS	DUMP CLASS	DUMP CLASS	DUMP CLASS

Storage Group Parameter Design ...

STORGRP NAME	GUARANTEED BACKUP FREQ	QUALIFIER GROUP	CYCLE START TIME	CYCLE END TIME	LIBRARY NAME

Storage Group Parameter Design ...

STORGRP NAME	LIBRARY NAME	LIBRARY NAME	LIBRARY NAME	LIBRARY NAME	LIBRARY NAME

Storage Group Parameter Design ...

STORGRP NAME	LIBRARY NAME	LIBRARY NAME	VOLUME FULL THRESHOLD	DRIVE START THRESHOLD

Storage Group Parameter Design ...

STORGRP NAME	VOLUME FULL AT WRITE ERROR	SYSTEM STATUS

Sample Storage Group Parameters

STORGRP NAME	SG TYPE	VIO MAXSIZE	VIO UNIT	AUTO MIGRATE	MIGRATE SYSTEM	AUTO BACK
SGWORK	pool			Y		N
SGTSO	pool			Y	MVSA	Y
SGPRIME	pool			Y	MVSA	Y

Sample Storage Group Parameters ...

STORGRP NAME	BACKUP SYSTEM	AUTO DUMP	DUMP SYSTEM	MIGR HIGH	MIGR LOW	...
SGWORK		N		99	1	
SGTSO	MVSA	Y	MVSA	95	80	
SGPRIME	MVSA	Y	MVSA	90	70	

Sample - Work Data

WK1

DSTYPE=TEMP

Work

SC=SCFAST

SG=SGWORK

Sample - TSO Data

HLQ=%%T%%

TSO

OTHERWISE

DSN=list

DSN=list

TS2

Normal
Data

TS3

LIST, LOG,
ISPF*TEMP.CNTL
type data

TS4

Used at
LOGON

SC=SCNORMAL
SG=SGTISO

SC=SCNORMAL
SG=SGTISO

SC=SCNORMAL
SG=SGTISO

Sample - Test Data

			HLQ=application list 2LQ=T*
	Test		
	OTHERWISE	2LQ=T%%TEMP*	DSTYPE=GDS
T5	T6	T7	
Normal	One Day	GDGs	
SC=SCNORMAL SG=SGISO	SC=SCNORMAL SG=SGISO	SC=SCNORMAL SG=SGISO	

Sample - Batch Production Data

Batch Production HLQ=application list
2LQ=P*

2LQ=P%%MF* OTHERWISE DSN=list

BP8 Working Files BP9 Master Files BP10 Critical Performance Files

SC=SCNORMAL SG=SGPRIME SC=SCNORMAL SG=SGPRIME SC=SCFAST SG=SGPRIME

BP11 GDGs DSTYPE=GDS

SC=SCNORMAL SG=SGPRIME

Sample - Exceptions

Exceptions

	DSN=list and JCL	DSN=list and JCL	JCL STORCLAS=SCNOSMS
EX12	Normal Performance Hand Placed	EX13 Critical Performance Hand Placed	EX14 Non SMS
	SC=SCSPECN SG=as usual	SC=SCSPECF SG=as usual	SC=☐☐

ACS Structure and Coding

ACS Code Structure

- Use structured ACS logic
 - Mandatory to handle changes and expansion
 - Isolate DASD, tape, and optical logic
- Use structured SMS assignment logic
 - Use canned block of ACS code
 - Use simple concepts of program structuring
- Structuring is best suited to SELECT WHEN coding style
- Structuring always works and is simple

Structured ACS Logic

```
PROC
SELECT /* ALLOCATION TYPE */
    WHEN
        SELECT /* STANDARDS ENFORCEMENT */
            . . . . .
        END /* STANDARDS ENFORCEMENT SELECT */
    WHEN &UNIT EQ &VALID_DASD_UNIT
        SELECT /* DASD ALLOCATION */
            . . . . .
        END /* DASD ALLOCATION SELECT */
    WHEN &UNIT EQ &VALID_TAPE_UNIT
        SELECT /* TAPE ALLOCATIONS */
            . . . . .
        END /* TAPE ALLOCATION SELECT */
    WHEN &ACSENVIR EQ &VALID_OPTICAL_ACSENVIR
        SELECT /* OPTICAL ALLOCATIONS */
            . . . . .
        END /* OPTICAL ALLOCATION SELECT */
    OTHERWISE /* NONMANAGED DEVICES */
        DO
            SET &STORCLAS = ¢¢
            EXIT
        END /* DO */
    END /* SELECT */
END /* PROC */
```

Structured SMS Assignment Logic

```
WHEN &UNIT EQ &VALID_DASD_UNIT

  SELECT /* DASD ALLOCATION */

    WHEN &USER EQ &SPECIAL_USER /* FOR SPECIAL USERS */
      /* REPEAT AS NECESSARY */

      DO
        SET &STORCLAS =
          EXIT
      END /* DO */

    WHEN &DSN EQ &EXCEPTION_SERVICE /* FOR EXCEPTION SERVICE*/
      /* REPEAT AS NECESSARY */

      DO
        SET &STORCLAS =
          EXIT
      END /* DO */

    WHEN &DSTYPE EQ ¢TEMP¢

      . . . . .

    WHEN &DSN EQ & TEST_DSN

      . . . . .

    WHEN &DSN EQ & PRODUCTION_DSN

      . . . . .

    . . . . . /* ADD OTHER DATA TYPES */

  OTHERWISE /* NONMANAGED DASD DATA */

  DO
    SET &STORCLAS = ¢¢
    EXIT
  END /* DO */

END /* DASD ALLOCATION SELECT */
```

ACS Coding Recommendations

- Use *Writing ACS Routines* (GG24-3403-1)
 - Helps keep coding simple
 - Helps others read and understand logic
 - Builds on what has proven successful
 - Uses supplied "building blocks" of code
 - Techniques always work and are simple

Use of FILTLISTS

- Whenever possible to minimize change

```
FILTLIST &SMS_APPLIC INCLUDE( ACCOUNT.** ,TEST.** , ... )  
  
SELECT  
  
    WHEN ( &DSN EQ &SMS_APPLIC )
```

- To avoid complex AND/OR testing

```
SELECT  
  
    WHEN ( ( &LLQ EQ ϕCNTLϕ )  
          OR ( &LLQ EQ LINK* AND &LLQ NE ϕLINKLSTϕ )  
          OR ( &LLQ EQ SRC* ) )  
  
FILTLIST &PDSF_LLQ INCLUDE ( ϕCNTLϕ , LINK* , SRC* )  
                EXCLUDE ( ϕLINKLSTϕ )  
  
SELECT  
  
    WHEN ( &LLQ EQ &PDSF_LLQ )
```

- Sort FILTLIST into alphabetic sequence
- Sort FILTLIST INCLUDE and EXCLUDE entries into alphabetic sequence
- Select readable FILTLIST names

```
    WHEN ( &HLQ EQ &SMS_MANAGED_DATA_HLQ )  
    WHEN ( &USER EQ &AUTHORIZED_FOR_NONSMS_STORCLAS_USER )
```

Use of SELECT WHEN or IF THEN

- Choose SELECT or IF and be consistent
- Use parentheses

```
WHEN ( &LLQ EQ &PDSF_LLQ OR
      &DSORG EQ 'PO' AND &USER EQ 'MYONE' )
```

```
WHEN ( &LLQ EQ &PDSF_LLQ OR
      ( &DSORG EQ 'PO' AND &USER EQ 'MYONE' ) )
```

- Do not test the more general condition first

```
SELECT
    WHEN ( &DSN EQ P*.*.CN* )
    WHEN ( &DSN EQ P*.*.CNTL )
END
```


Use of SELECT WHEN or IF THEN ...

- Code explicit OTHERWISE in SELECT statements
- Code explicit EXITS

```
WHEN ( &GROUP EQ &DBMAINT AND &STORCLAS EQ DB* )  
    DO  
        SET &STORCLAS = ¢DBBASE¢  
        EXIT  
    END
```

- Do not be afraid of nested SELECTs

```
SELECT (&USER)  
    WHEN (P*)  
    WHEN (R*)  
        SELECT (&DSN(2))  
            WHEN (¢ABC¢)  
            OTHERWISE  
        END  
    OTHERWISE  
END
```

ACS Code Documentation

- Describe the purpose of a section of code
- Describe the specific purpose of the programming statements
- Use indentation and comments to tie END to what it ends

```
PROC
  SELECT
    WHEN
      DO          /* Do Number 1      */
        SET
        EXIT
      END        /* End of Do Number 1 */
    WHEN
      DO          /* Do Number 2      */
        SET
        EXIT
      END        /* End of Do Number 2 */
    OTHERWISE
  END          /* End of Select    */
END          /* End of Proc      */
```

- Have a change log within the program

Storage Class ACS Coding

Storage Class ACS Considerations

- Assign defaults to data types through ACS logic
- Assign exceptions through either JCL or ACS logic
- Protect JCL storage class assignment through ACS logic
- Always use SCNOSMS
- Redrive ACS logic on RECALL and RECOVERY
- HSM system data sets and other non-SMS-managed data are excluded as non-SMS-managed data types
- RACF defaults not used with DFSMS FIT

Mandatory - DASD Allocations

```

/*****
/* INCLUDE ALL DASD ALLOCATIONS IN THIS BLOCK, INCLUDE BOTH DASD      */
/* ALLOCATIONS FOR SMS AND NON SMS                                     */
*****/

FILTLIST &VALID_DASD_UNIT INCLUDE
(¢3380¢,¢3390¢,¢SYSDA¢,¢SYSALLDA¢,¢¢,
 ¢_____¢,¢_____¢,¢_____¢)

FILTLIST &VALID_OPTICAL_ACSENVIR INCLUDE
(¢STORE¢,¢CHANGE¢,¢TRANS¢)

WHEN ((&UNIT EQ &VALID_DASD_UNIT)
AND (&ACSENVIR NE &VALID_OPTICAL_ACSENVIR))

        SELECT /* DASD DATA ALLOCATION */

/*****
/* ALLOW SPECIAL USERS TO PLACE SELECTED DATA SETS OUTSIDE OF SMS    */
/* BY USING STORCLAS=SCNOSMS AT ALLOCATION                             */
*****/

FILTLIST &SPECIAL_SCNOSMS_USER INCLUDE
(¢_____¢,¢_____¢,¢_____¢)

WHEN ((&USER EQ &SPECIAL_SCNOSMS_USER)
AND (&STORCLAS EQ ¢SCNOSMS¢))

        DO
                SET &STORCLAS = ¢¢
                EXIT
        END /* DO */

```

Mandatory - Exceptions

```
/*  
/* TAILOR AND ADD JCL EXCEPTIONS HERE IF THEY ARE USED */  
*/
```

```
/*  
/* TAILOR AND ADD DATA SET LIST EXCEPTIONS HERE IF THEY ARE USED */  
/* INCLUDE SEQUENTIAL STRIPING AND CONCURRENT COPY */  
*/
```

Mandatory - Temp Data Type

```
/*  
/* TAILOR AND ADD EITHER TEMP AND VIO OR TEMP AND NO VIO HERE IF */  
/* TEMP DATA IS SMS MANAGED */  
*/
```

```
/*  
/* TAILOR AND ADD TEMP SHORT DURATION HERE IF TEMP SHORT DURATION */  
/* DATA AND IF IT IS SMS MANAGED */  
*/
```


Mandatory - TSO and Test Data Types

```
/*  
/* TAILOR AND ADD TSO LOGON AND LIST DATA SUBTYPES IF THEY HAVE */  
/* STORAGE CLASSES DIFFERENT THAN OTHER TSO DATA AND THEY ARE */  
/* SMS MANAGED */  
*/
```

```
/*  
/* TAILOR AND ADD OTHER TSO SUBTYPES HERE IF THEY ARE SMS MANAGED */  
*/
```

```
/*  
/* TAILOR AND ADD TEST SUBTYPES HERE IF THEY ARE SMS MANAGED */  
*/
```

Mandatory - Production Data Types

```
/*  
/* TAILOR AND ADD BATCH PRODUCTION SUBTYPES HERE IF THEY ARE */  
/* SMS MANAGED */  
*/
```

```
/*  
/* TAILOR AND ADD ONLINE PRODUCTION SUBTYPES HERE IF THEY ARE */  
/* SMS MANAGED */  
*/
```

```
/*  
/* TAILOR AND ADD ANY OTHER DATA SUBTYPES HERE IF THEY ARE */  
/* SMS MANAGED */  
*/
```

```
/*  
/* MAKE ALL OTHER DASD DATA SETS NONMANAGED */  
*/
```

```
OTHERWISE /* NONMANAGED DASD DATA */
```

```
DO  
    SET &STORCLAS = ¢¢  
    EXIT  
END /* DO */
```

```
END /* DASD ALLOCATION SELECT */
```

Mandatory - Tape Allocations

```

/*****
/* INCLUDE ALL TAPE ALLOCATIONS IN THIS BLOCK */
/*****

FILTLIST &VALID_TAPE_UNIT INCLUDE
(ç3490ç,ç3480ç,ç3590ç,TAPE*,çT3480ç,çT3490ç,çT3590ç,çAFF=ç,
ç_____ç,ç_____ç,ç_____ç)

WHEN (&UNIT EQ &VALID_TAPE_UNIT)

SELECT /* TAPE ALLOCATIONS */

        WHEN (&UNIT EQ &VALID_TAPE_UNIT)
        SET &STORCLAS = &STORCLAS

/*****
/* TAILOR AND ADD TAPE MOUNT MANAGEMENT SUBTYPES HERE IF THEY EXIST */
/*****

/*****
/* TAILOR AND ADD SMS MANAGED TAPE SUBTYPES HERE IF THEY EXIST */
/*****

/*****
/* ADD OTHERWISE FOR TAPE DATA HERE IF EITHER TAPE MOUNT MANAGEMENT */
/* OR SMS MANAGED TAPE EXIST */
/*****

END /* TAPE ALLOCATION SELECT */

```

Mandatory - Optical Allocations

```

/*****
/* INCLUDE ALL OPTICAL ALLOCATIONS IN THIS BLOCK, INCLUDE LOGIC FOR */
/* SUBTYPES FOR IAFC OR IMAGE PLUS HERE IF THEY EXIST */
*****/

      WHEN (&ACSENVIR EQ &VALID_OPTICAL_ACSENVIR)

          SELECT /* OPTICAL ALLOCATIONS */

              WHEN (&ACSENVIR EQ &VALID_OPTICAL_ACSENVIR)
                  SET &STORCLAS = &STORCLAS

/*****
/* TAILOR AND ADD OPTICAL STORE FUNCTION HERE IF OPTICAL DATA EXISTS */
*****/

/*****
/* TAILOR AND ADD OPTICAL TRANSITION AND CHANGE FUNCTIONS HERE IF */
/* OPTICAL DATA EXISTS */
*****/

/*****
/* ADD OTHERWISE FOR OPTICAL DATA HERE IF OPTICAL DATA EXISTS */
*****/

      END /* OPTICAL ALLOCATION SELECT */

/*****
/* THIS OTHERWISE IS FOR ALLOCATIONS OTHER THAN DASD, TAPE, OR */
/* OPTICAL */
*****/

      OTHERWISE /* NONMANAGED DEVICES */

          DO
              SET &STORCLAS = ¢¢
              EXIT
          END /* DO */

      END /* SELECT */

END /* PROC */
```

Optional - Data Set Naming Standard

```
/* ***** */
/* TAILOR TO ENFORCE DATA SET NAMING STANDARDS */
/* ***** */

FILTLIST &STANDARD_NAME_DSN INCLUDE
(_____/_____/_____,
_____/_____/_____,
_____/_____/_____)

WHEN (&DSN NE ☐☐)

SELECT /* STANDARDS ENFORCEMENT */

    WHEN (&DSN NE &STANDARD_NAME_DSN)

        DO
            WRITE ☐(☐&DSN☐) FAILS NAMING STANDARD☐
            EXIT CODE(16)
        END /* DO */

    OTHERWISE /* OTHER DASD DATA */

        DO
            END /* DO */

END /* STANDARDS ENFORCEMENT SELECT */
```

Optional - Volume Fencing Standard

```
/*
*****
/* TAILOR TO ENFORCE VOLUME FENCING STANDARDS
*****
*/

FILTLIST &OKAY_____USER INCLUDE
(ϕ_____ϕ,ϕ_____ϕ,ϕ_____ϕ)

FILTLIST &OKAY_____VOL INCLUDE
(ϕ_____ϕ,ϕ_____ϕ,ϕ_____ϕ)

WHEN (&DSN NE ϕϕ)

    SELECT /* STANDARDS ENFORCEMENT */

        WHEN ((&USER NE &OKAY_____USER)
              AND (&ALLVOL NE &OKAY_____VOL))

            DO
                WRITE ϕ(ϕ&USERϕ) CANNOT USE VOLUME (ϕ&ANYVOLϕ)ϕ
                EXIT CODE(16)
            END /* DO */

        OTHERWISE /* OTHER DASD DATA */

            DO
            END /* DO */

    END /* STANDARDS ENFORCEMENT SELECT */
```

Optional - Data Set Size Standard

```
/* *****  
/* TAILOR TO ENFORCE DATA SET SIZE STANDARDS *  
/* *****  
  
    WHEN (&DSN NE ¢¢)  
  
        SELECT /* STANDARDS ENFORCEMENT */  
  
            WHEN ((&SIZE GT ____MB)  
                OR (&MAXSIZE GT _____MB))  
  
                DO  
                    WRITE ¢(¢&DSN¢) EXCEEDS SIZE STANDARD¢  
                    EXIT CODE(16)  
                END /* DO */  
  
            OTHERWISE /* OTHER DASD DATA */  
  
                DO  
                END /* DO */  
  
        END /* STANDARDS ENFORCEMENT SELECT */
```

Optional - JCL Exceptions

```
/* ***** */
/* TAILOR AND SET STORCLAS FOR ALLOCATION EXCEPTIONS */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH SET OF SPECIAL USERS THAT ARE */
/* ALLOWED TO REQUEST SMS SERVICE THROUGH STORCLAS AT ALLOCATION */
/* ***** */

FILTLIST &SPECIAL_SC_USER INCLUDE
(  _____  ,  _____  ,  _____  )

FILTLIST &SPECIAL_STORCLAS INCLUDE
(  _____  ,  _____  ,  _____  )

WHEN ( ( &USER EQ &SPECIAL_SC_USER )
AND ( &STORCLAS EQ &SPECIAL_STORCLAS ) )

DO
    SET &STORCLAS = &STORCLAS
EXIT
END /* DO */
```


Optional - Sequential Striping Exception

```
/* *****  
/* TAILOR AND SET STORCLAS FOR SEQUENTIAL STRIPING IF USED          */  
/* *****  
  
    FILTLIST &SEQUENTIAL_STRIPING_DSN INCLUDE  
    ( _____ / _____ / _____ /  
      _____ / _____ / _____ /  
      _____ / _____ / _____ )  
  
    WHEN ( &DSN EQ &SEQUENTIAL_STRIPING_DSN )  
  
        DO  
            SET &STORCLAS =ç _____ ç  
            EXIT  
        END /* DO */
```

Optional - Concurrent Copy Exceptions

```
/* *****  
/* TAILOR AND SET STORCLAS FOR CONCURRENT COPY IF USED          */  
/* *****  
  
    FILTLIST &CONCURRENT_COPY_DSN INCLUDE  
    ( _____, _____, _____,  
      _____, _____, _____,  
      _____, _____, _____ )  
  
    WHEN ( &DSN EQ &CONCURRENT_COPY_DSN )  
  
        DO  
            SET &STORCLAS =ç_____ç  
            EXIT  
        END /* DO */
```

Optional - Data Set List Exceptions

```
/* *****  
/* TAILOR AND SET STORCLAS FOR DATA SET LIST EXCEPTIONS */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SET OF EXCEPTION STORAGE CLASSES */  
/* BASED ON DATA SET NAME LISTS */  
/* *****  
  
FILTLIST &EXCEPTION_SC_SERVICE_DSN INCLUDE  
( _____ / _____ / _____ /  
  _____ / _____ / _____ /  
  _____ / _____ / _____ )  
  
WHEN ( &DSN EQ &EXCEPTION_SC_SERVICE_DSN )  
  
  DO  
    SET &STORCLAS =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Temp Data Type and VIO

```
/* ***** */
/* TAILOR AND SET STORCLAS FOR SYSTEM TEMPORARY SUBTYPES IF THEY */
/* ARE SMS MANAGED AND VIO IS USED */
/* ***** */
/* SORT WORK DATA SETS SHOULD NOT BE IN VIO DUE TO PERFORMANCE. */
/* THE SORT WORK DATA SETS ARE ASSIGNED TO STORCLAS=SCNOVIO TO */
/* PROVIDE A FLAG TO BE USED IN THE STORAGE GROUP ACS LOGIC SO */
/* THAT THE SORT WORK DATA SETS ARE NOT ASSIGNED TO VIO */
/* ***** */
/* AFTER ALL TEMP DATA IS CONVERTED TO SMS YOU SHOULD REMOVE THE */
/* AND (&HLQ EQ &SMS_MANAGED_TEMP_DATA_USER)) LINE AND ADD A ) TO */
/* THE END OF THE PRIOR LINE */
/* ***** */
```

```
FILTLIST &SMS_MANAGED_TEMP_DATA_USER INCLUDE
(ç_____ç,ç_____ç,ç_____ç)
```

```
WHEN ((&DSTYPE EQ çTEMPç)
AND (&DD EQ SORT%K*)
AND (&HLQ EQ &SMS_MANAGED_TEMP_DATA_USER))
```

```
DO
    SET &STORCLAS = çSCNOVIOç
    EXIT
END /* DO */
```

```
WHEN ((&DSTYPE EQ çTEMPç)
AND (&HLQ EQ &SMS_MANAGED_TEMP_DATA_USER))
```

```
DO
    SET &STORCLAS = ç_____ç
    EXIT
END /* DO */
```

Optional - Temp Data Type and No VIO

```
/* *****  
/* TAILOR AND SET STORCLAS FOR SYSTEM TEMPORARY SUBTYPES IF THEY */  
/* ARE SMS MANAGED AND VIO IS NOT USED */  
/* *****  
/* AFTER ALL TEMP DATA IS CONVERTED TO SMS YOU SHOULD REMOVE THE */  
/* AND (&HLQ EQ &SMS_MANAGED_TEMP_DATA_USER)) LINE AND ADD A ) TO */  
/* THE END OF THE PRIOR LINE */  
/* *****
```

```
FILTLIST &SMS_MANAGED_TEMP_DATA_HLQ INCLUDE  
(ϕ_____ϕ,ϕ_____ϕ,ϕ_____ϕ)
```

```
WHEN ((&DSTYPE EQ ϕTEMPϕ)  
AND (&HLQ EQ &SMS_MANAGED_TEMP_DATA_HLQ))
```

```
DO  
    SET &STORCLAS = ϕ_____ϕ  
    EXIT  
END /* DO */
```

Optional - Temp Short Duration

```
/* *****  
/* TAILOR AND SET STORCLAS FOR TEMP SHORT DURATION SUBTYPES IF THEY */  
/* ARE SMS MANAGED */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SUBTYPE OF TEMP SHORT DURATION DATA */  
/* THAT HAS A DIFFERENT STORAGE CLASS */  
/* *****  
/* AFTER ALL TEMP SHORT DURATION IS CONVERTED TO SMS YOU SHOULD */  
/* REMOVE THE AND (&HLQ EQ &SMS_MANAGED_TEMPPERM_DATA_HLQ) ) LINE */  
/* AND REMOVE THE FIRST ( FROM THE PRIOR LINE */  
/* *****
```

```
FILTLIST &TEMPPERM_DATA_DSN INCLUDE  
( _____, _____, _____,  
  _____, _____, _____,  
  _____, _____, _____ )
```

```
FILTLIST &SMS_MANAGED_TEMPPERM_DATA_HLQ INCLUDE  
( ¢ _____ ¢, ¢ _____ ¢, ¢ _____ ¢ )
```

```
WHEN ( (&DSN EQ &TEMPPERM_DATA_DSN)  
AND ( &HLQ EQ &SMS_MANAGED_TEMPPERM_DATA_HLQ ) )
```

```
DO  
  SET &STORCLAS = ¢ _____ ¢  
  EXIT  
END /* DO */
```

Optional - TSO Logon and List

```
/* ***** */
/* TAILOR AND SET STORCLAS FOR TSO LOGON AND LIST SUBTYPES IF THEY */
/* ARE SMS MANAGED AND HAVE DIFFERENT STORAGE CLASSES FROM OTHER */
/* TSO SUBTYPES */
/* ***** */
/* AFTER ALL TSO DATA IS CONVERTED TO SMS YOU SHOULD REMOVE THE */
/* AND (&HLQ EQ &SMS_MANAGED_TSO_DATA_HLQ) LINE AND REMOVE THE */
/* FIRST ( FROM THE PRIOR LINES */
/* ***** */
```

```
FILTLIST &TSO_LOGON_DATA_DSN INCLUDE
(*.ISPF.PROFILE, _____, _____,
_____, _____, _____,
_____, _____, _____)
```

```
FILTLIST &TSO_LIST_DATA_DSN INCLUDE
(**.LIST, **.OUTLIST, **.LINKLIST,
_____, _____, _____,
_____, _____, _____)
```

```
FILTLIST &SMS_MANAGED_TSO_DATA_HLQ INCLUDE
(ç_____ç, ç_____ç, ç_____ç)
```

```
WHEN ((&DSN EQ &TSO_LOGON_DATA_DSN)
AND (&HLQ EQ &SMS_MANAGED_TSO_DATA_HLQ))
```

```
DO
    SET &STORCLAS =ç_____ç
    EXIT
END /* DO */
```

```
WHEN ((&DSN EQ &TSO_LIST_DATA_DSN)
AND (&HLQ EQ &SMS_MANAGED_TSO_DATA_HLQ))
```

```
DO
    SET &STORCLAS =ç_____ç
    EXIT
END /* DO */
```

Optional - TSO

```
/* *****  
/* TAILOR AND SET STORCLAS FOR TSO SUBTYPES IF THEY ARE SMS_MANAGED */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH TSO SUBTYPE THAT HAS A DIFFERENT */  
/* STORAGE CLASS AND IS OTHER THAN LOGON OR LIST SUBTYPES */  
/* *****  
/* AFTER ALL TSO DATA IS CONVERTED TO SMS YOU SHOULD REMOVE THE */  
/* AND (&HLQ EQ &SMS_MANAGED_TSO_DATA_HLQ) LINE AND REMOVE THE */  
/* FIRST ( FROM THE PRIOR LINES */  
/* *****
```

```
    FILTLIST &TSO_DATA_HLQ INCLUDE  
    ( ¢_____¢, ¢_____¢, ¢_____¢ )
```

```
    FILTLIST &SMS_MANAGED_TSO_DATA_HLQ INCLUDE  
    ( ¢_____¢, ¢_____¢, ¢_____¢ )
```

```
    WHEN ( (&HLQ EQ &TSO_DATA_HLQ)  
    AND ( &HLQ EQ &SMS_MANAGED_TSO_DATA_HLQ ) )
```

```
        DO  
            SET &STORCLAS = ¢_____¢  
            EXIT  
        END /* DO */
```


Optional - Test

```
/* *****  
/* TAILOR AND SET STORCLAS FOR TEST SUBTYPES IF THEY ARE SMS_MANAGED */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH TEST SUBTYPE THAT HAS A DIFFERENT */  
/* STORAGE CLASS */  
/* *****  
/* AFTER ALL TEST DATA IS CONVERTED TO SMS YOU SHOULD REMOVE THE */  
/* AND (&HLQ EQ &SMS_MANAGED_TEST_DATA_HLQ) LINE AND REMOVE THE */  
/* FIRST ( FROM THE PRIOR LINES */  
/* *****
```

```
FILTLIST &TEST_DATA_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )
```

```
FILTLIST &SMS_MANAGED_TEST_DATA_HLQ INCLUDE  
( ¢ _____ ¢ , ¢ _____ ¢ , ¢ _____ ¢ )
```

```
WHEN ( (&DSN EQ &TEST_DATA_DSN)  
AND ( &HLQ EQ &SMS_MANAGED_TEST_DATA_HLQ ) )
```

```
DO  
  SET &STORCLAS = ¢ _____ ¢  
  EXIT  
END /* DO */
```

Optional - Batch Production

```
/* *****  
/* TAILOR AND SET STORCLAS FOR BATCH PRODUCTION SUBTYPES IF THEY ARE */  
/* SMS_MANAGED */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH BATCH PRODUCTION SUBTYPE THAT HAS A */  
/* DIFFERENT STORAGE CLASS */  
/* *****  
/* AFTER ALL BATCH PRODUCTION DATA IS CONVERTED TO SMS YOU SHOULD */  
/* REMOVE THE AND (&HLQ EQ &SMS_MANAGED_BATCH_PROD_DATA_HLQ) ) LINE */  
/* AND REMOVE THE FIRST ( FROM THE PRIOR LINE */  
/* *****
```

```
FILTLIST &BATCH_PROD_DATA_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )
```

```
FILTLIST &SMS_MANAGED_BATCH_PROD_DATA_HLQ INCLUDE  
( ¢ _____ ¢ , ¢ _____ ¢ , ¢ _____ ¢ )
```

```
WHEN ( ( &DSN EQ &BATCH_PROD_DATA_DSN )  
AND ( &HLQ EQ &SMS_MANAGED_BATCH_PROD_DATA_HLQ ) )
```

```
DO  
  SET &STORCLAS = ¢ _____ ¢  
  EXIT  
END /* DO */
```

Optional - Online Production

```
/* *****  
/* TAILOR AND SET STORCLAS FOR ONLINE PRODUCTION SUBTYPES IF THEY */  
/* ARE SMS_MANAGED */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH ONLINE PRODUCTION SUBTYPE THAT HAS A */  
/* DIFFERENT STORAGE CLASS */  
/* *****  
/* AFTER ALL BATCH PRODUCTION DATA IS CONVERTED TO SMS YOU SHOULD */  
/* REMOVE THE AND (&HLQ EQ &SMS_MANAGED_ONLINE_PROD_DATA_HLQ) ) LINE */  
/* AND REMOVE THE FIRST ( FROM THE PRIOR LINE */  
/* *****
```

```
FILTLIST &ONLINE_PROD_DATA_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )
```

```
FILTLIST &SMS_MANAGED_ONLINE_PRODDATA_HLQ INCLUDE  
( ¢ _____ ¢ , ¢ _____ ¢ , ¢ _____ ¢ )
```

```
WHEN ( ( &DSN EQ &ONLINE_PROD_DATA_DSN )  
AND ( &HLQ EQ &SMS_MANAGED_ONLINE_PRODDATA_HLQ ) )
```

```
DO  
  SET &STORCLAS = ¢ _____ ¢  
  EXIT  
END /* DO */
```

Optional - Other

```
/* *****  
/* TAILOR AND SET STORCLAS FOR OTHER SUBTYPES IF THEY ARE SMS MANAGED*  
/* *****  
/* REPEAT THIS BLOCK FOR EACH OTHER SUBTYPE THAT HAS A DIFFERENT      */  
/* STORAGE CLASS                                                         */  
/* *****  
/* AFTER ALL OTHER DATA IS CONVERTED TO SMS YOU SHOULD REMOVE THE    */  
/* AND (&HLQ EQ &SMS_MANAGED_____DATA_HLQ) ) LINE AND REMOVE      */  
/* THE FIRST ( FROM THE PRIOR LINE                                       */  
/* *****
```

```
    FILTLIST &_____DATA_DSN INCLUDE  
    (_____,_____,_____,  
     _____,_____,_____,  
     _____,_____,_____)
```

```
    FILTLIST &SMS_MANAGED_____DATA_HLQ INCLUDE  
    (¢_____,¢_____,¢_____)
```

```
    WHEN ((&DSN EQ &_____DATA_DSN)  
    AND  (&HLQ EQ &SMS_MANAGED_____DATA_HLQ))
```

```
        DO  
            SET &STORCLAS =¢_____  
            EXIT  
        END /* DO */
```

Optional - Tape Mount Management

```
/* ***** */
/* TAILOR AND SET STORCLAS FOR TAPE MOUNT MANAGEMENT SUBTYPES IF */
/* THEY EXIST */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH TAPE MOUNT MANAGEMENT SUBTYPE THAT HAS */
/* A DIFFERENT STORAGE CLASS */
/* ***** */

FILTLIST &TAPE_MOUNT_MANAGEMENT_DSN INCLUDE
( _____ , _____ , _____ ,
  _____ , _____ , _____ ,
  _____ , _____ , _____ )

FILTLIST &TAPE_MOUNT_MANAGEMENT_DATACLAS INCLUDE
( $ _____ $ , $ _____ $ , $ _____ $ )

WHEN ( ( &DSN EQ &TAPE_MOUNT_MANAGEMENT_DSN )
OR ( &DATACLAS EQ &TAPE_MOUNT_MANAGEMENT_DATACLAS ) )

DO
    SET &STORCLAS = $ _____ $
    EXIT
END /* DO */
```

Optional - SMS-Managed Tape

```
/* *****  
/* TAILOR AND SET STORCLAS FOR SMS MANAGED TAPE SUBTYPES IF THEY */  
/* EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SMS MANAGED TAPE SUBTYPE THAT HAS A */  
/* DIFFERENT STORAGE CLASS */  
/* *****
```

```
    FILTLIST &SMS_MANAGED_TAPE_DSN INCLUDE  
    ( _____ / _____ / _____ /  
      _____ / _____ / _____ /  
      _____ / _____ / _____ )
```

```
    WHEN ((&LABEL NE &SL) )  
    OR (&LABEL NE &AL)
```

```
        DO  
            SET &STORCLAS =&C  
            EXIT  
        END /* DO */
```

```
    WHEN (&DSN EQ &SMS_MANAGED_TAPE_DSN)
```

```
        DO  
            SET &STORCLAS =&C _____ &C  
            EXIT  
        END /* DO */
```

Optional - Otherwise Tape

```
/* *****  
/* OTHERWISE STATEMENT FOR TAPE ALLOCATIONS *  
/* *****  
  
    OTHERWISE  
  
        DO  
            SET &STORCLAS = @@  
            EXIT  
        END /* DO */
```

Optional - Optical Store

```
/* *****  
/* TAILOR AND SET STORCLAS FOR STORE PROCESSING OF OPTICAL SUBTYPES */  
/* IF THEY EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH OPTICAL SUBTYPE THAT HAS A DIFFERENT */  
/* STORAGE CLASS AND STORE PROCESSING LOGIC */  
/* *****  
  
FILTLIST &OPTICAL_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )  
  
WHEN ((&DSN EQ &OPTICAL_DSN)  
AND (&ACSENVIR EQ ¢STORE¢))  
  
DO  
  SET &STORCLAS = ¢ _____ ¢  
  EXIT  
END /* DO */
```


Optional - Optical Transition and Change

```
/* *****  
/* TAILOR AND SET STORCLAS FOR CTRAN AND CHANGE PROCESSING OF */  
/* OPTICAL SUBTYPES IF THEY EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH OPTICAL SUBTYPE THAT HAS A DIFFERENT */  
/* STORAGE CLASS AND CTRAN OR CHANGE PROCESSING LOGIC */  
/* *****
```

```
    WHEN ((&DSN EQ &OPTICAL_DSN)  
          AND (&ACSENVIR EQ ¢CTRANS¢))
```

```
        DO  
            SET &STORCLAS = ¢_____¢  
            EXIT  
        END /* DO */
```

```
    WHEN ((&DSN EQ &OPTICAL_DSN)  
          AND (&ACSENVIR EQ ¢CHANGE¢))
```

```
        DO  
            SET &STORCLAS = ¢_____¢  
            EXIT  
        END /* DO */
```

Optional - Otherwise Optical

```
/* *****  
/* OTHERWISE STATEMENT FOR OPTICAL ALLOCATIONS *  
/* *****
```

```
OTHERWISE
```

```
DO  
    SET &STORCLAS = ¢¢  
    EXIT  
END /* DO */
```


Sample - Storage Class ACS Code ...

```
FILTLIST &BATCH_PROD_DATA_DSN INCLUDE
(P*.**)

FILTLIST &VALID_TAPE_UNIT INCLUDE
(ç3490ç,ç3480ç,ç3590ç,TAPE*,çT3480ç,çT3490ç,
çT3590ç,çAFF=)

FILTLIST &TAPE_MOUNT_MANAGEMENT_DSN INCLUDE
(çç)

FILTLIST &SMS_MANAGED_TAPE_DSN INCLUDE
(çç)

FILTLIST &OPTICAL_DSN INCLUDE
(çç)
```

```
/** *_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_**/
```

```
SELECT /* ALLOCATION TYPE */
```

```
/**
/* INCLUDE ALL STANDARDS ENFORCEMENT LOGIC IN THIS BLOCK IN */
/* ONE AND ONLY ONE OF EITHER THE DATA CLASS OR STORAGE ACS */
/* ROUTINE, LEAVE BLOCK IN OTHER ACS ROUTINES FOR CONSISTENCY */
**/
```

```
/**
/* START STANDARDS ENFORCEMENT LOGIC BLOCK */
/**
/*
**/
/* END STANDARDS ENFORCEMENT LOGIC BLOCK */
**/
```

Sample - Storage Class ACS Code ...

```
/* ***** */
/* INCLUDE ALL DASD ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* BOTH DASD ALLOCATIONS TO SMS AND TO NON SMS */
/* ***** */

      WHEN ((&UNIT EQ &VALID_DASD_UNIT)
            AND (&ACSENVIR NE &VALID_OPTICAL_ACSENVIR))

          SELECT /* DASD DATA ALLOCATION */

/* ***** */
/* DO NOT MANAGE DATA FOR SPECIAL USERS THAT HAVE BEEN */
/* GIVEN AUTHORITY TO USE STORCLAS=SCNOSMS ON THEIR JCL */
/* THIS ALLOWS SELECTIVE OVERRIDE OF THE STORAGE CLASS */
/* ACS LOGIC DURING AN EMERGENCY SITUATION SO DATA THAT */
/* WOULD NORMALLY BE MANAGED BY SMS CAN BE ALLOCATED OUTSIDE OF SMS */
/* ***** */

          WHEN ((&USER EQ &SPECIAL_NONSMS_USER)
                AND (&STORCLAS EQ ¢SCNOSMS¢))

              DO
                  SET &STORCLAS = ¢¢
                  EXIT
              END /* DO */

/* ***** */
/* DUPLICATE THIS BLOCK FOR EACH SET OF SPECIAL */
/* USERS THAT ARE ALLOWED TO REQUEST SMS SERVICE */
/* THROUGH JCL STORCLAS AND MGMTCLAS PARAMETERS */
/* THIS MAY INCLUDE ALL USERS FOR THOSE STORCLAS */
/* EXCEPTIONS THAT YOU ARE WILLING TO LET ALL USERS */
/* USE WITHOUT ANY PROTECTION */
/* ***** */

          WHEN ((&USER EQ &SPECIAL_SC_USER)
                AND (&STORCLAS EQ &SPECIAL_STORCLAS))

              DO
                  SET &STORCLAS = &STORCLAS
                  EXIT
              END /* DO */
```

Sample - Storage Class ACS Code ...

```
/* *****  
/* DUPLICATE THIS BLOCK FOR EACH SET OF EXCEPTION *  
/* SERVICES, BASED ON DATA CLASSIFICATION, THAT *  
/* ARE BASED ON CONTROL THROUGH DATA SET NAME LISTS *  
/* *****
```

```
WHEN (&DSN EQ &CRITICAL_PERFORMANCE_DSN)
```

```
DO  
    SET &STORCLAS =ϕSCFASTϕ  
    EXIT  
END /* DO */
```

```
/* *****  
/* INCLUDE SYSTEM TEMPORARY DATA ALLOCATIONS INTO *  
/* SMS IN THIS BLOCK, CONVERSION TO SMS IS CONTROLLED *  
/* BY HLQ AND THE TEST FOR HLQ CAN BE REMOVED AFTER *  
/* FULL SYSTEM TEMPORARY DATA CONVERSION TO SMS *  
/* *****  
/* SET SMS MANAGED SYSTEM TEMPORARY DATA SETS TO THEIR *  
/* DEFAULT STORCLAS DETERMINED BY DATA CLASSIFICATION *  
/* *****
```

```
WHEN (&DSTYPE EQ ϕTEMPϕ)
```

```
DO  
    SET &STORCLAS = ϕSCFASTϕ  
    EXIT  
END /* DO */
```

Sample - Storage Class ACS Code ...

```
/* ***** */
/* INCLUDE TSO DATA (IDENTIFIED BY HLQ) FOR SMS */
/* IN THIS BLOCK, CONVERSION TO SMS IS CONTROLLED */
/* BY HLQ AND THE TEST FOR HLQ CAN BE REMOVED AFTER */
/* FULL TSO DATA CONVERSION TO SMS */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH DATA CLASSIFICATION */
/* TYPE OF TSO DATA, USING THE FILTLIST AND LOGIC */
/* TEST IDENTIFIED FOR THE DATA CLASSIFICATION */
/* DIAGRAM AND ASSIGN THE DEFAULT STORCLAS DEFINED */
/* DURING THE STORAGE CLASS DESIGN */
/* ***** */
```

WHEN (&HLQ EQ &TSO_DATA_HLQ)

```
DO
    SET &STORCLAS = $\phi$ SCNORMAL $\phi$ 
    EXIT
END /* DO */
```

```
/* ***** */
/* INCLUDE TEST DATA (IDENTIFIED BY DSN) FOR SMS */
/* IN THIS BLOCK, CONVERSION TO SMS IS CONTROLLED */
/* BY HLQ AND THE TEST FOR HLQ CAN BE REMOVED AFTER */
/* FULL TEST DATA CONVERSION TO SMS */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH DATA CLASSIFICATION */
/* TYPE OF TEST DATA, USING THE FILTLIST AND LOGIC */
/* TEST IDENTIFIED FOR THE DATA CLASSIFICATION */
/* DIAGRAM AND ASSIGN THE DEFAULT STORCLAS DEFINED */
/* DURING THE STORAGE CLASS DESIGN */
/* ***** */
```

WHEN (&DSN EQ &TEST_DATA_DSN)

```
DO
    SET &STORCLAS = $\phi$ SCNORMAL $\phi$ 
    EXIT
END /* DO */
```

Sample - Storage Class ACS Code ...

```
/* ***** */
/* INCLUDE BATCH PRODUCTION DATA ( IDENTIFIED          */
/* BY DSN) FOR SMS IN THIS BLOCK                          */
/* CONVERSION TO SMS IS CONTROLLED                       */
/* BY HLQ AND THE TEST FOR HLQ CAN BE REMOVED AFTER     */
/* FULL BATCH PRODUCTION DATA CONVERSION TO SMS       */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH DATA CLASSIFICATION      */
/* TYPE OF BATCH PRODUCTION DATA, USING THE FILTLIST   */
/* AND LOGIC TEST IDENTIFIED FOR THE DATA CLASSIFICATION */
/* DIAGRAM AND ASSIGN THE DEFAULT STORCLAS DEFINED     */
/* DURING THE STORAGE CLASS DESIGN                     */
/* ***** */

        WHEN (&DSN EQ &BATCH_PROD_DATA_DSN)

                DO
                        SET &STORCLAS =çSCNORMALç
                        EXIT
                END /* DO */

/* ***** */
/* REPEAT THE ABOVE BLOCKS FOR ANY OTHER DATA          */
/* TYPES DEFINED DURING THE DATA CLASSIFICATION        */
/* PROCESS AND INCLUDE EACH DATA CLASSIFICATION ASSIGNMENT BOX */
/* ***** */
/* FOR ANY OTHER DATA TYPES NOT IDENTIFIED FOR        */
/* SMS MANAGEMENT, PLACE THEM IN NON SMS STORAGE       */
/* ***** */

        OTHERWISE /* NONMANAGED DASD DATA */

                DO
                        SET &STORCLAS = çç
                        EXIT
                END /* DO */

        END /* DASD ALLOCATION SELECT */
```


Sample - Storage Class ACS Code ...

```
/* ***** */
/* INCLUDE ALL TAPE ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* BOTH TAPE ALLOCATIONS TO SMS AND TO NON SMS */
/* PRIOR TO SMS MANAGED TAPE OR TAPE MOUNT MANAGEMENT */
/* ALL TAPE ALLOCATION WOULD BE TO NON SMS */
/* ***** */

WHEN (&UNIT EQ &VALID_TAPE_UNIT)

    SELECT /* TAPE ALLOCATIONS */

        WHEN (&DSN EQ &TAPE_MOUNT_MANAGEMENT_DSN)

            DO
                SET &STORCLAS =çç
                EXIT
            END /* DO */

        WHEN (&DSN EQ &SMS_MANAGED_TAPE_DSN)

            DO
                SET &STORCLAS =çç
                EXIT
            END /* DO */

        OTHERWISE

            DO
                SET &STORCLAS = çç
                EXIT
            END /* DO */

    END /* TAPE ALLOCATION SELECT */
```

Sample - Storage Class ACS Code ...

```

/*****
/* INCLUDE ALL OPTICAL ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* LOGIC FROM PRODUCTS LIKE IAFD OR IMAGE PLUS HERE */
*****/

/*****
/* START OPTICAL ALLOCATION LOGIC BLOCK */
*****/
/*
*****/
/* END OPTICAL ALLOCATION LOGIC BLOCK */
*****/

        OTHERWISE /* NONMANAGED DEVICES */

                DO
                        SET &STORCLAS = ¢¢
                        EXIT
                END /* DO */

        END /* SELECT */

END /* PROC */
```

Storage Group ACS Coding

Storage Group ACS Considerations

- Review variables not available to storage group ACS routine
- Data set assignment to storage group based on service type
 - &SIZE
 - &DSTYPE or generic FILTLIST of data set name
- Data set assignment to storage group based on data type
 - STORAGE CLASS
 - MANAGEMENT CLASS
- Do not assign SORTWK data sets to VIO

Mandatory - Exceptions and Temp Data Type

```
/*  
/* TAILOR AND ADD GUARANTEED SPACE LOGIC HERE IF GUARANTEED SPACE IS *  
/* USED *  
*/
```

```
/*  
/* TAILOR AND ADD DATA SET LIST EXCEPTIONS HERE IF THEY ARE USED *  
/* INCLUDE SEQUENTIAL STRIPING AND CONCURRENT COPY *  
*/
```

```
/*  
/* TAILOR AND ADD EITHER TEMP AND VIO OR TEMP AND NO VIO HERE IF *  
/* TEMPORARY DATA IS SMS MANAGED *  
*/
```

```
/*  
/* TAILOR AND ADD TEMP SHORT DURATION HERE IF THERE IS TEMP SHORT *  
/* DURATION DATA AND IF IT IS SMS MANAGED *  
*/
```

Mandatory - TSO and Test Data Types

```
/* **** */
/* * TAILOR AND ADD TSO LOGON AND LIST SUBTYPE IF THEY HAVE STORAGE */
/* * GROUPS DIFFERENT THAN OTHER TSO SUBTYPES AND THEY ARE SMS MANAGED */
/* **** */
```

```
/* **** */
/* * TAILOR AND ADD OTHER TSO SUBTYPES HERE IF THEY ARE SMS MANAGED */
/* **** */
```

```
/* **** */
/* * TAILOR AND ADD TEST SUBTYPES HERE IF THEY ARE SMS MANAGED */
/* **** */
```

Mandatory - Production Data Types

```
/*  
/* TAILOR AND ADD BATCH PRODUCTION SUBTYPES HERE IF THEY ARE */  
/* SMS MANAGED */  
*/
```

```
/*  
/* TAILOR AND ADD ONLINE PRODUCTION SUBTYPES HERE IF THEY ARE */  
/* SMS MANAGED */  
*/
```

```
/*  
/* TAILOR AND ADD OTHER SUBTYPES HERE IF THEY ARE SMS MANAGED */  
*/
```

```
/*  
/* TAILOR AND SET STORGRP FOR ALL OTHER SMS MANAGED SUBTYPES */  
*/
```

```
OTHERWISE /* UNEXPECTED DASD DATA */
```

```
DO  
    SET &STORGRP = ¢ _____ ¢  
    EXIT  
END /* DO */
```

```
END /* DASD ALLOCATION SELECT */
```


Mandatory - Tape Allocations

```

/*****
/* INCLUDE ALL TAPE ALLOCATIONS IN THIS BLOCK                               */
/*****

FILTLIST &VALID_TAPE_UNIT INCLUDE
(ç3490ç,ç3480ç,ç3590ç,TAPE*,çT3480ç,çT3490ç,çT3590ç,çAFF=ç,
 ç_____ç,ç_____ç,ç_____ç)

WHEN (&UNIT EQ &VALID_TAPE_UNIT)

SELECT /* TAPE ALLOCATIONS */

        WHEN (&UNIT EQ &VALID_TAPE_UNIT)
        SET &STORGRP = &STORGRP

/*****
/* TAILOR AND ADD TAPE MOUNT MANAGEMENT SUBTYPES HERE IF THEY EXIST      */
/*****

/*****
/* TAILOR AND ADD SMS MANAGED TAPE SUBTYPES HERE IF THEY EXIST           */
/*****

/*****
/* ADD OTHERWISE FOR TAPE DATA HERE IF EITHER TAPE MOUNT MANAGEMENT     */
/* OR SMS MANAGED TAPE EXIST                                             */
/*****

END /* TAPE ALLOCATION SELECT */

```

Mandatory - Optical Allocations

```

/*****
/* INCLUDE ALL OPTICAL ALLOCATIONS IN THIS BLOCK          */
*****/

      WHEN (&ACSENVIR EQ &VALID_OPTICAL_ACSENVIR)

          SELECT /* OPTICAL ALLOCATIONS */

              WHEN (&ACSENVIR EQ &VALID_OPTICAL_ACSENVIR)
                  SET &STORGRP = &STORGRP

/*****
/* TAILOR AND ADD OPTICAL STORE FUNCTION HERE IF OPTICAL DATA EXISTS */
*****/

/*****
/* TAILOR AND ADD OPTICAL TRANSITION FUNCTION HERE IF OPTICAL DATA
/* EXISTS
*****/

/*****
/* ADD OTHERWISE FOR OPTICAL DATA HERE IF OPTICAL DATA EXISTS
*****/

      END /* OPTICAL ALLOCATION SELECT */

/*****
/* THIS OTHERWISE IS FOR ALLOCATIONS OTHER THAN DASD, TAPE, OR
/* OPTICAL
*****/

      OTHERWISE /* UNEXPECTED DEVICES */

          DO
              SET &STORGRP = ¢_____¢
              EXIT
          END /* DO */

      END /* SELECT */

END /* PROC */
```

Optional - Guaranteed Space

```
/* ***** */
/* SET ALL STORAGE GROUPS WHEN A STORAGE CLASS WITH GUARANTEED      */
/* SPACE IS ASSIGNED                                                */
/* ***** */

/* DEFINE FILTLIST FOR ALL STORAGE CLASSES WITH */
/* GUARANTEED SPACE AND THEN ASSIGN ANY DATA SET */
/* WITH A GUARANTEED SPACE STORAGE CLASS TO ALL */
/* STORAGE GROUPS AND THUS ALL VOLUMES */

FILTLIST &GUARANTEED_SPACE_STORCLAS INCLUDE
(¢_____¢,¢_____¢,¢_____¢)

WHEN (&STORCLAS EQ &GUARANTEED_SPACE_STORCLAS)

DO
    SET &STORGRP = ¢_____¢,¢_____¢,¢_____¢
    EXIT
END /* END DO */
```

Optional - Sequential Striping Exception

```
/* *****  
/* TAILOR AND SET STORGRP FOR SEQUENTIAL STRIPING IF USED          */  
/* *****  
  
    FILTLIST &SEQUENTIAL_STRIPING_DSN INCLUDE  
    ( _____ / _____ / _____ /  
      _____ / _____ / _____ /  
      _____ / _____ / _____ / )  
  
    WHEN ( &DSN EQ &SEQUENTIAL_STRIPING_DSN )  
  
        DO  
            SET &STORGRP =ç _____ ç  
            EXIT  
        END /* DO */
```

Optional - Concurrent Copy Exceptions

```
/* *****  
/* TAILOR AND SET STORGRP FOR CONCURRENT COPY IF USED *  
/* *****
```

```
FILELIST &CONCURRENT_COPY_DSN INCLUDE  
( _____ / _____ / _____ /  
  _____ / _____ / _____ /  
  _____ / _____ / _____ )
```

```
WHEN ( &DSN EQ &CONCURRENT_COPY_DSN)
```

```
DO  
  SET &STORGRP =¢ _____ ¢  
  EXIT  
END /* DO */
```

Optional - Data Set List Exceptions

```
/* *****  
/* TAILOR AND SET STORGRP FOR DATA SET LIST EXCEPTIONS */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SET OF EXCEPTION STORAGE GROUPS */  
/* BASED ON DATA SET NAME LISTS */  
/* *****  
  
FILTLIST &EXCEPTION_SG_SERVICE_DSN INCLUDE  
( _____ / _____ / _____ /  
  _____ / _____ / _____ /  
  _____ / _____ / _____ )  
  
WHEN ( &DSN EQ &EXCEPTION_SG_SERVICE_DSN )  
  
  DO  
    SET &STORGRP =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Temp Data Type and VIO

```
/* *****  
/* TAILOR AND SET STORGRP FOR SYSTEM TEMPORARY SUBTYPES AND VIO */  
/* *****  
/* USE THE SWITCH OF STORCLAS=SCNOVIO SET IN THE STORAGE CLASS ACS */  
/* ROUTINE TO IDENTIFY SORTWORK DATA SETS THAT SHOULD NOT BE PLACED */  
/* IN VIO */  
/* *****
```

```
    WHEN ( &STORCLAS EQ ¢SCNOVIO¢ )
```

```
        DO  
            SET &STORGRP = ¢_____¢  
            EXIT  
        END /* END DO */
```

```
/* *****  
/* TAILOR AND SET STORGRP FOR SYSTEM TEMPORARY SUBTYPES AND VIO */  
/* FOR ALL NON SORTWORK TEMPORARY DATA SETS */  
/* *****
```

```
    WHEN ( &DSTYPE EQ ¢TEMP¢ )
```

```
        DO  
            SET &STORGRP = ¢SGVIO¢, ¢_____¢  
            EXIT  
        END /* END DO */
```

Optional - Temp Data Type and No VIO

```
/*  
*****  
/* TAILOR AND SET STORGRP FOR SYSTEM TEMPORARY SUBTYPES AND NO VIO */  
*****  
*/  
  
    WHEN ( &DSTYPE EQ 'TEMP' )  
  
        DO  
            SET &STORGRP = '_____'  
            EXIT  
        END /* END DO */
```


Optional - Data Set Size

```
/* *****  
/* TAILOR AND SET STORGRP WHEN YOU WANT TO SEPARATE LARGE DATA SETS */  
/* FROM AVERAGE SIZED DATA SETS */  
/* *****  
/* YOU CAN ADD THIS LOGIC BLOCK TO ANY SET STORAGE GROUP SUBTYPES */  
/* *****  
  
    WHEN ((&SIZE GT ____MB)  
        OR (&MAXSIZE GT ____MB))  
  
        DO  
            SET &STORGRP = ¢_____¢  
            EXIT  
        END /* END DO */
```

Optional - Temp Short Duration

```
/* **** */
/* TAILOR AND SET STORGRP FOR TEMP SHORT DURATION SUBTYPES */
/* **** */
/* REPEAT THIS BLOCK FOR EACH SUBTYPE OF TEMP SHORT DURATION THAT */
/* HAS A DIFFERENT STORAGE GROUP */
/* **** */

FILTLIST &TEMPPERM_DATA_DSN INCLUDE
(
  _____, _____, _____,
  _____, _____, _____,
  _____, _____, _____)

WHEN (&DSN EQ &TEMPPERM_DATA_DSN)

DO
  SET &STORGRP =ç_____ç
  EXIT
END /* DO */
```

Optional - TSO Logon and List

```
/* ***** */
/* TAILOR AND SET STORGRP FOR TSO LOGON AND LIST SUBTYPES IF THEY */
/* HAVE DIFFERENT STORAGE GROUPS FROM OTHER TSO SUBTYPES */
/* ***** */

FILTLIST &TSO_LOGON_DATA_DSN INCLUDE
( *.ISPF.PROFILE, _____, _____,
  _____, _____, _____ )

FILTLIST &TSO_LIST_DATA_DSN INCLUDE
( **.LIST, **.OUTLIST, **.LINKLIST,
  _____, _____, _____ )

WHEN ( &DSN EQ &TSO_LOGON_DATA_DSN )

DO
  SET &STORGRP =ç_____ç
  EXIT
END /* DO */

WHEN ( &DSN EQ &TSO_LIST_DATA_DSN )

DO
  SET &STORGRP =ç_____ç
  EXIT
END /* DO */
```

Optional - TSO

```
/* ***** */
/* TAILOR AND SET STORGRP FOR TSO SUBTYPES */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH SUBTYPE OF TSO DATA THAT HAS A */
/* DIFFERENT STORAGE GROUP AND IS OTHER THAN LOGON OR LIST SUBTYPES */
/* ***** */

FILTLIST &TSO_DATA_HLQ INCLUDE
(¢_____¢,¢_____¢,¢_____¢)

WHEN (&HLQ EQ &TSO_DATA_HLQ)

DO
    SET &STORGRP =¢_____¢
    EXIT
END /* DO */
```

Optional - Test

```
/* ***** */
/* TAILOR AND SET STORGRP FOR TEST SUBTYPES */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH SUBTYPE OF TEST DATA THAT HAS A */
/* DIFFERENT STORAGE GROUP */
/* ***** */

FILTLIST &TEST_DATA_DSN INCLUDE
( _____, _____, _____,
  _____, _____, _____,
  _____, _____, _____ )

WHEN ( &DSN EQ &TEST_DATA_DSN )

DO
    SET &STORGRP =ç_____ç
    EXIT
END /* DO */
```

Optional - Batch Production

```
/* *****  
/* TAILOR AND SET STORGRP FOR BATCH PRODUCTION SUBTYPES */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SUBTYPE OF BATCH PRODUCTION DATA THAT */  
/* HAS A DIFFERENT STORAGE GROUP */  
/* *****  
  
FILTLIST &BATCH_PROD_DATA_DSN INCLUDE  
( _____ / _____ / _____ /  
  _____ / _____ / _____ /  
  _____ / _____ / _____ )  
  
WHEN ( &DSN EQ &BATCH_PROD_DATA_DSN)  
  
  DO  
    SET &STORGRP =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Online Production

```
/* *****  
/* TAILOR AND SET STORGRP FOR ONLINE PRODUCTION SUBTYPES */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SUBTYPE OF ONLINE PRODUCTION DATA THAT */  
/* HAS A DIFFERENT STORAGE GROUP */  
/* *****  
  
FILTLIST &ONLINE_PROD_DATA_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )  
  
WHEN ( &DSN EQ &ONLINE_PROD_DATA_DSN )  
  
  DO  
    SET &STORGRP =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Other

```
/* ***** */
/* TAILOR AND SET STORGRP FOR OTHER SUBTYPES */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH SUBTYPE OF OTHER DATA THAT HAS A */
/* DIFFERENT STORAGE GROUP */
/* ***** */

FILTLIST &_____DATA_DSN INCLUDE
(_____,_____,_____
_____,_____,_____
_____,_____,_____
)

WHEN (&DSN EQ &_____DATA_DSN)

DO
    SET &STORGRP =ç_____ç
    EXIT
END /* DO */
```


Optional - Tape Mount Management

```
/* *****  
/* TAILOR AND SET STORGRP FOR TAPE MOUNT MANAGEMENT SUBTYPES */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH TAPE MOUNT MANAGEMENT SUBTYPE THAT HAS */  
/* A DIFFERENT STORAGE GROUP */  
/* *****  
  
FILTLIST &TAPE_MOUNT_MANAGEMENT_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )  
  
FILTLIST &TAPE_MOUNT_MANAGEMENT_DATACLAS INCLUDE  
( $ _____ $ , $ _____ $ , $ _____ $ )  
  
WHEN ( ( &DSN EQ &TAPE_MOUNT_MANAGEMENT_DSN )  
OR ( &DATACLAS EQ &TAPE_MOUNT_MANAGEMENT_DATACLAS ) )  
  
DO  
  SET &STORGRP = $ _____ $  
  EXIT  
END /* DO */
```

Optional - SMS-Managed Tape

```
/* ***** */
/* TAILOR AND SET STORGRP FOR SMS MANAGED TAPE SUBTYPES */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH SMS MANAGED TAPE SUBTYPES THAT HAS A */
/* DIFFERENT STORAGE GROUP */
/* ***** */

FILTLIST &SMS_MANAGED_TAPE_DSN INCLUDE
(
_____/_____/_____/
_____/_____/_____/
_____/_____/_____/
)

WHEN ( &DSN EQ &SMS_MANAGED_TAPE_DSN )

DO
    SET &STORGRP =ç_____ç
    EXIT
END /* DO */
```

Optional - Otherwise Tape

```
/* *****  
/* OTHERWISE STATEMENT FOR TAPE ALLOCATIONS *  
/* *****
```

```
OTHERWISE
```

```
DO  
    SET &STORGRP = $_____ $  
    EXIT  
END /* DO */
```

Optional - Optical Store

```
/* *****  
/* TAILOR AND SET STORGRP FOR STORE PROCESSING OF OPTICAL SUBTYPES */  
/* *****  
/* CTRAN AND CHANGE PROCESSING DOES NOT OCCUR FOR STORAGE GROUP */  
/* *****
```

```
FILTLIST &OPTICAL_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )
```

```
WHEN ((&DSN EQ &OPTICAL_DSN)  
AND (&ACSENVIR EQ ¢STORE¢))
```

```
DO  
  SET &STORGRP = ¢ _____ ¢  
  EXIT  
END /* DO */
```

Optional - Otherwise Optical

```
/* *****  
/* OTHERWISE STATEMENT FOR OPTICAL ALLOCATIONS *  
/* *****
```

```
OTHERWISE
```

```
DO  
    SET &STORGRP = ¢_____¢  
    EXIT  
END /* DO */
```


Sample - Storage Group ACS Code ...

```
SELECT /* ALLOCATION TYPE */
```

```
/* ***** */
/* INCLUDE ALL STANDARDS ENFORCEMENT LOGIC IN THIS BLOCK IN */
/* ONE AND ONLY ONE OF EITHER THE DATA CLASS OR STORAGE ACS */
/* ROUTINE, LEAVE BLOCK IN OTHER ACS ROUTINES FOR CONSISTENCY */
/* ***** */

/* ***** */
/* START STANDARDS ENFORCEMENT LOGIC BLOCK */
/* ***** */
/* */
/* ***** */
/* END STANDARDS ENFORCEMENT LOGIC BLOCK */
/* ***** */
```

Sample - Storage Group ACS Code ...

```
/* ***** */
/* INCLUDE ALL DASD ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* BOTH DASD ALLOCATIONS TO SMS AND TO NON SMS */
/* ***** */

      WHEN ((&UNIT EQ &VALID_DASD_UNIT)
            AND (&ACSENVIR NE &VALID_OPTICAL_ACSENVIR))

            SELECT /* DASD DATA ALLOCATION */

/* ***** */
/* DEFINE FILTLIST FOR ALL STORAGE CLASSES WITH */
/* GUARANTEED SPACE AND THEN ASSIGN ANY DATA SET */
/* WITH A GUARANTEED SPACE STORAGE CLASS TO ALL */
/* STORAGE GROUPS AND THUS ALL VOLUMES TO ALLOW */
/* STORAGE ADMINISTRATOR TO ALLOCATE ON EVERY VOLUME */
/* ***** */

            WHEN (&STORCLAS EQ &GUARANTEED_SPACE_STORCLAS)

                    DO
                            SET &STORGRP =çSGWORKç,çSGISOç,çSGPRIMEç
                            EXIT
                    END /* END DO */

/* ***** */
/* ASSIGN OTHER TEMPORARY DATA SETS TO VIO OR A */
/* NORMAL STORAGE GROUP BASED ON THE SIZE PARAMETER */
/* OF THE VIO STORAGE GROUP DEFINITION */
/* DFSMS DOES THE SIZE TESTING */
/* ***** */

            WHEN ( &DSTYPE EQ çTEMPç )

                    DO
                            SET &STORGRP = çSGWORKç
                            EXIT
                    END /* END DO */
```


Sample - Storage Group ACS Code ...

```
/* ***** */
/* INCLUDE TSO DATA (IDENTIFIED BY HLQ) FOR SMS */
/* IN THIS BLOCK, CONVERSION TO SMS IS CONTROLLED */
/* BY HLQ AND THE TEST FOR HLQ CAN BE REMOVED AFTER */
/* FULL TSO DATA CONVERSION TO SMS */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH DATA CLASSIFICATION */
/* TYPE OF TSO DATA, USING THE FILTLIST AND LOGIC */
/* TEST IDENTIFIED FOR THE DATA CLASSIFICATION */
/* DIAGRAM AND ASSIGN THE DEFAULT STORGRP DEFINED */
/* DURING THE STORAGE GROUP DESIGN */
/* ***** */
```

WHEN (&HLQ EQ &TSO_DATA_HLQ)

```
DO
    SET &STORGRP =çSGTISOç
    EXIT
END /* DO */
```

```
/* ***** */
/* INCLUDE TEST DATA (IDENTIFIED BY DSN) FOR SMS */
/* IN THIS BLOCK, CONVERSION TO SMS IS CONTROLLED */
/* BY HLQ AND THE TEST FOR HLQ CAN BE REMOVED AFTER */
/* FULL TEST DATA CONVERSION TO SMS */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH DATA CLASSIFICATION */
/* TYPE OF TEST DATA, USING THE FILTLIST AND LOGIC */
/* TEST IDENTIFIED FOR THE DATA CLASSIFICATION */
/* DIAGRAM AND ASSIGN THE DEFAULT STORGRP DEFINED */
/* DURING THE STORAGE GROUP DESIGN */
/* ***** */
```

WHEN (&DSN EQ &TEST_DATA_DSN)

```
DO
    SET &STORGRP =çSGTISOç
    EXIT
END /* DO */
```

Sample - Storage Group ACS Code ...

```

/*****/
/* INCLUDE BATCH PRODUCTION DATA ( IDENTIFIED                */
/* BY DSN) FOR SMS IN THIS BLOCK                               */
/* CONVERSION TO SMS IS CONTROLLED                             */
/* BY HLQ AND THE TEST FOR HLQ CAN BE REMOVED AFTER           */
/* FULL BATCH PRODUCTION DATA CONVERSION TO SMS              */
/*****/
/* REPEAT THIS BLOCK FOR EACH DATA CLASSIFICATION            */
/* TYPE OF BATCH PRODUCTION DATA, USING THE FILTLIST         */
/* AND LOGIC TEST IDENTIFIED FOR THE DATA CLASSIFICATION     */
/* DIAGRAM AND ASSIGN THE DEFAULT STORGRP DEFINED             */
/* DURING THE STORAGE GROUP DESIGN                            */
/*****/

        WHEN (&DSN EQ &BATCH_PROD_DATA_DSN)

                DO
                        SET &STORGRP =çSGPRIMEç
                        EXIT
                END /* DO */

/*****/
/* REPEAT THE ABOVE BLOCKS FOR ANY OTHER DATA                */
/* TYPES DEFINED DURING THE DATA CLASSIFICATION              */
/* PROCESS AND INCLUDE EACH DATA CLASSIFICATION ASSIGNMENT   */
/* BOX                                                          */
/*****/
/* FOR ANY OTHER DATA TYPES NOT IDENTIFIED FOR              */
/* SMS MANAGEMENT, PLACE THEM IN NON SMS STORAGE             */
/*****/

        OTHERWISE /* NONMANAGED DASD DATA */

                DO
                        SET &STORGRP = çSGISOç
                        EXIT
                END /* DO */

        END /* DASD ALLOCATION SELECT */
```

Sample - Storage Group ACS Code ...

```
/* ***** */
/* INCLUDE ALL TAPE ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* BOTH TAPE ALLOCATIONS TO SMS AND TO NON SMS */
/* PRIOR TO SMS MANAGED TAPE OR TAPE MOUNT MANAGEMENT */
/* ALL TAPE ALLOCATION WOULD BE TO NON SMS */
/* ***** */

/* ***** */
/* START TAPE ALLOCATION LOGIC BLOCK */
/* ***** */
/*
/* ***** */
/* END TAPE ALLOCATION LOGIC BLOCK */
/* ***** */
```

Sample - Storage Group ACS Code ...

```

/*****
/* INCLUDE ALL OPTICAL ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* LOGIC FROM PRODUCTS LIKE IAFB OR IMAGE PLUS HERE */
/*****
/* FILTLIST &VALID_OPTICAL_ACSENVIR INCLUDE STORE LOGIC */
/*****

/*****
/* START OPTICAL ALLOCATION LOGIC BLOCK */
/*****
/*
/*
/*****
/* END OPTICAL ALLOCATION LOGIC BLOCK */
/*****

        OTHERWISE /* NONMANAGED DEVICES */

                DO
                        SET &STORGRP =%SGTISO%
                        EXIT
                END /* DO */

        END /* SELECT */

END /* PROC */
```

Management Class Design

Management Class Design

Topics

- Design considerations
- Management class questions
- Definition of management classes
- Update of data classification

Management Class Benefits

- Manages the data set's life cycle
 - Release overallocated space from sequential and partitioned data sets with secondary allocation
 - Control data set migration
 - Allow early GDS migration and expire rolled-off generations
 - Control frequency of automatic backup, number of backup versions maintained, use of concurrent copy, and automatic deletion of backups
 - Control automatic deletion
- DFHSM functions at a data set level

Management Class Considerations

- Involve application owners to understand business needs
- Guaranteed space restricts the recall of migrated data sets
- Storage group can override automatic migration and backup

Management Class Recommendations

- Control the number of management classes
- Migrate directly to tape where appropriate
- Take incremental backups to reduce volume dump frequency
- Protect management classes that provide special services (such as no migration or extra backups)
- Test automatic deletion before use
- Consider ABARS for disaster recovery support
- Use `COMMAND OR AUTO MIGRATE = BOTH`, `PRIMARY DAYS = 0 or 1`, and `LEVEL 1 DAYS = 0` for TMM data

Management Class Design Questions

1. How long should data be kept when unused?

- Early deletion (3-10 days)
- Medium deletion (35 to 200 days)
- Late deletion (400 or more days)
- Never delete

2. When and to where should data be moved in the storage hierarchy when unused?

- Direct to tape
- Early migration (3-10 days)
- Medium migration (20 to 100 days)
- Late migration (100 or more days)
- Never migrate

Management Class Design Questions ...

3. When and how much data backup is required?

- Never backup
- Backup on change
- Keep 1 backup copy
- Keep a few backup generations
- Keep many backup generations
- Use concurrent copy

4. How should space release be handled?

- No partial release
- Partial release of overallocated space if secondary allocation
 - at close
- Partial release of overallocated space if secondary allocation
 - during overnight automatic processing

5. What GDG action is required?

- Keep 1 generation of GDG on primary
- Keep multiple generations of GDG on primary
- Expire GDS at roll off

Management Class Design Table

Data Type									
Classification box									
Data Deletion									
Early delete									
Medium delete									
Late delete									
Never delete									
Data Migration									
Direct to tape									
Early migrate									
Medium migrate									
Late migrate									
Never migrate									
Backup of Data									
Never backup									
One version									
Few versions									
Many versions									
Concurrent copy									
Partial Release									
Conditional/close									
Conditional/daily									
GDG Action									
1 GDS on primary									
Many GDSs on primary									
Roll off expire									
Management Class ID									

Sample Management Class Answers

Data Type	WK	TS	TS	TS	T	T	T	
Classification box	1	2	3	4	5	6	7	
Data Deletion								
Early delete			X			X		
Medium delete								
Late delete								
Never delete		X		X	X		X	
Data Migration								
Direct to tape								
Early migrate		X						
Medium migrate					X		X	
Late migrate								
Never migrate				X				
Backup of Data								
Never backup			X			X	X	
One version								
Few versions		X			X			
Many versions				X				
Concurrent copy								
Partial Release								
Conditional/close								
Conditional/daily		X	X		X	X	X	
GDG Action								
1 GDS on primary							X	
Many GDSs on primary								
Roll off expire							Y	
Management Class ID	N/A	1	2	3	4	2	5	

Sample Management Class Answers ...

Data Type	B P	B P	B P	B P	EX	EX	EX	
Classification box	8	9	10	11	12	13	14	
Data Deletion								
Early delete								
Medium delete								
Late delete								
Never delete	X	X	X	X				
Data Migration								
Direct to tape								
Early migrate	X			X				
Medium migrate								
Late migrate		X						
Never migrate			X					
Backup of Data								
Never backup								
One version								
Few versions	X			X				
Many versions		X	X					
Concurrent copy								
Partial Release								
Conditional/close								
Conditional/daily	X			X				
GDG Action								
1 GDS on primary				X				
Many GDSs on primary								
Roll off expire				Y				
Management Class ID	1	6	3	1	N/A	N/A	N/A	

Management Class Naming

1.

2.

3.

4.

5.

6.

7.

8.

9.

Sample Management Class Naming

1. MCEARLY

2. MCINTERM

3. MCNOMIG

4. MCSTAND

5. MCNOBACK

6. MCLATE

Management Class Protection Design

Mgmt. Class	JCL Use	Protected	By What

Sample Management Class Protection

Mgmt. Class	JCL Use	Protected	By What
MCEARLY	No		
MCINTERM	No		
MCNOMIG	No		
MCSTAND	No		
MCNOBACK	No		
MCLATE	No		

Management Class Parameter Design

MGMTCLAS NAME	EXPIRE NON-USAGE	EXPIRE DATE/DAYS	RET LIMIT	PARTIAL RELEASE	PRIMARY DAYS

Management Class Parameter Design ...

MGMTCLAS NAME	LEVEL 1 DAYS	CMD/AUTO MIGRATE	# GDG ON PRIMARY	ROLLED-OFF GDS ACTION	BACKUP FREQ.

Management Class Parameter Design ...

MGMTCLAS NAME	# BACKUPS DS EXISTS	# BACKUPS DS DELETED	RETAIN DAYS ONLY BACKUP	RETAIN DAYS EXTRA BACKUPS

Management Class Parameter Design ...

MGMTCLAS NAME	ADM/USER BACKUP	AUTO BACKUP	...	CREATION YEARS	CREATION MONTHS

Management Class Parameter Design ...

MGMTCLAS NAME	CREATION DAYS	LAST USE YEARS	LAST USE MONTHS	LAST USE DAYS

Management Class Parameter Design ...

MGMTCLAS NAME	MONTHLY ON DAY	QUARTERLY ON DAY	QUARTERLY IN MONTH	YEARLY ON DAY	YEARLY IN MONTH

Management Class Parameter Design ...

MGMTCLAS NAME	# VERSIONS	RETAIN ONLY VERSION	RETAIN ONLY UNIT	RETAIN EXTRA VERSIONS	RETAIN EXTRA UNIT

Management Class Parameter Design ...

MGMTCLAS NAME	COPY SERIAL	BACKUP COPY TECHNIQUE	ABACKUP COPY TECHNIQUE

Sample Management Class Parameters

MGMTCLAS NAME	EXPIRE NON-USAGE	EXPIRE DATE/DAYS	RET LIMIT	PARTIAL RELEASE	PRIMARY DAYS
MCEARLY	NOLIMIT	NOLIMIT		C	3
MCINTERM		1		C	
MCNOMIG	NOLIMIT	NOLIMIT		N	9999
MCSTAND	NOLIMIT	NOLIMIT		C	10
MCNOBACK	NOLIMIT	NOLIMIT		C	10
MCLATE	NOLIMIT	NOLIMIT		N	17

Sample Management Class Parameters ...

MGMTCLAS NAME	LEVEL 1 DAYS	CMD/AUTO MIGRATE	# GDG ON PRIMARY	ROLLED-OFF GDS ACTION	BACKUP FREQ.
MCEARLY	7	BOTH	1	EXPIRE	1
MCINTERM		NONE			
MCNOMIG		NONE			1
MCSTAND	7	BOTH			1
MCNOBACK	7	BOTH	1	EXPIRE	
MCLATE	45	BOTH			1

Sample Management Class Parameters ...

MGMTCLAS NAME	# BACKUPS DS EXISTS	# BACKUPS DS DELETED	RETAIN DAYS ONLY BACKUP	RETAIN DAYS EXTRA BACKUPS
MCEARLY	2	1	NOLIMIT	70
MCINTERM				
MCNOMIG	8	1	NOLIMIT	70
MCSTAND	2	1	100	35
MCNOBACK				
MCLATE	8	1	NOLIMIT	70

Sample Management Class Parameters ...

MGMTCLAS NAME	ADM/USER BACKUP	AUTO BACKUP	...	CREATION YEARS	CREATION MONTHS
MCEARLY	BOTH	Y			
MCINTERM	NONE	N			
MCNOMIG	BOTH	Y			
MCSTAND	BOTH	Y			
MCNOBACK	NONE	N			
MCLATE	BOTH	Y			

Sample Management Class Parameters ...

MGMTCLAS NAME	COPY SERIAL	BACKUP COPY TECHNIQUE	ABACKUP COPY TECHNIQUE
MCEARLY		STANDARD	
MCINTERM			
MCNOMIG		STANDARD	
MCSTAND		STANDARD	
MCNOBACK			
MCLATE		STANDARD	

Sample - Work Data

WKL

DSTYPE=TEMP

Work

SC=SCFAST

SG=SGWORK

MC=N/A

Sample - TSO Data

HLQ=%%T%%

TSO

OTHERWISE

DSN=list

DSN=list

TS2

Normal
Data

TS3

LIST, LOG,
ISPF*TEMP.CNTL
type data

TS4

Used at
LOGON

SC=SCNORMAL
SG=SGTSO
MC=MCEARLY

SC=SCNORMAL
SG=SGTSO
MC=MCINTERM

SC=SCNORMAL
SG=SGTSO
MC=MCNOMIG

Sample - Test Data

```
Test          HLQ=application list
              2LQ=T*

              OTHERWISE          2LQ=T%%TEMP*          DSTYPE=GDS

T5            T6            T7
Normal       One Day       GDGs

SC=SCNORMAL  SC=SCNORMAL  SC=SCNORMAL
SG=SGTISO    SG=SGTISO    SG=SGTISO
MC=MCSTAND   MC=MCINTERM   MC=MCNOBACK
```

Sample - Batch Production Data

Batch HLQ=application list
Production 2LQ=P*

2LQ=P%%MF* OTHERWISE DSN=list

BP8	BP9	BP10
Working	Master	Critical
Files	Files	Performance
		Files

SC=SCNORMAL	SC=SCNORMAL	SC=SCFAST
SG=SGPRIME	SG=SGPRIME	SG=SGPRIME
MC=MCEARLY	MC=MCLATE	MC=MCNOMIG

BP11 DSTYPE=GDS
GDGs

SC=SCNORMAL
SG=SGPRIME
MC=MCEARLY

Sample - Exceptions

Exceptions

	DSN=list and JCL	DSN=list and JCL	JCL STORCLAS=SCNOSMS
EX12	Normal Performance Hand Placed	EX13 Critical Performance Hand Placed	EX14 Non SMS
	SC=SCSPECN SG=as usual MC=as usual	SC=SCSPECF SG=as usual MC=as usual	SC=☐☐

Management Class ACS Coding

Management Class ACS Considerations

- Assign defaults to data types through ACS logic
- Assign exceptions through either JCL or ACS logic
- Protect JCL management class assignment through ACS logic
- Redrive ACS logic on RECALL and RECOVERY
- Redrive ACS logic on RENAME
- RACF defaults not used with DFSMS FIT

Mandatory - Exceptions

```
/*  
/* TAILOR AND ADD JCL EXCEPTIONS HERE IF THEY ARE USED */  
*/
```

```
/*  
/* TAILOR AND ADD DATA SET LIST EXCEPTIONS HERE IF THEY ARE USED */  
/* INCLUDE CONCURRENT COPY */  
*/
```


Mandatory - Temp, TSO, and Test Data Types

```
/*  
/* SYSTEM TEMPORARY DATA SETS DO NOT HAVE MANAGEMENT CLASSES */  
/*  
/*  
/*  
/*
```

```
/*  
/* TAILOR AND ADD TEMP SHORT DURATION HERE IF IT EXISTS */  
/*  
/*  
/*  
/*
```

```
/*  
/* TAILOR AND ADD TSO LOGON AND LIST SUBTYPES IF THEY HAVE */  
/* MANAGEMENT CLASSES DIFFERENT THAN OTHER TSO DATA */  
/*  
/*  
/*
```

```
/*  
/* TAILOR AND ADD OTHER TSO SUBTYPES HERE IF THEY EXIST */  
/*  
/*  
/*  
/*
```

```
/*  
/* TAILOR AND ADD TEST SUBTYPES HERE IF THEY EXIST */  
/*  
/*  
/*  
/*
```

Mandatory - Production Data Types

```
/*  
/* TAILOR AND ADD BATCH PRODUCTION SUBTYPES HERE IF THEY EXIST */  
*/
```

```
/*  
/* TAILOR AND ADD ONLINE PRODUCTION SUBTYPES HERE IF THEY EXIST */  
*/
```

```
/*  
/* TAILOR AND ADD OTHER SUBTYPES HERE IF THEY EXIST */  
*/
```

```
/*  
/* TAILOR AND SET MGMTCLAS FOR ALL OTHER SMS MANAGED SUBTYPES */  
*/
```

```
OTHERWISE /* UNEXPECTED DASD DATA */
```

```
DO  
    SET &MGMTCLAS = c_____c  
    EXIT  
END /* DO */
```

```
END /* DASD ALLOCATION SELECT */
```

Mandatory - Tape Allocations

```

/*****
/* INCLUDE ALL TAPE ALLOCATIONS IN THIS BLOCK */
/*****

FILTLIST &VALID_TAPE_UNIT INCLUDE
(ç3490ç,ç3480ç,ç3590ç,TAPE*,çT3480ç,çT3490ç,çT3590ç,çAFF=ç,
ç_____ç,ç_____ç,ç_____ç)

WHEN (&UNIT EQ &VALID_TAPE_UNIT)

SELECT /* TAPE ALLOCATIONS */

        WHEN (&UNIT EQ &VALID_TAPE_UNIT)
        SET &MGMTCLAS = &MGMTCLAS

/*****
/* TAILOR AND ADD TAPE MOUNT MANAGEMENT SUBTYPES HERE IF THEY EXIST */
/*****

/*****
/* TAILOR AND ADD SMS MANAGED TAPE SUBTYPES HERE IF THEY EXIST */
/*****

/*****
/* ADD OTHERWISE FOR TAPE DATA HERE IF EITHER TAPE MOUNT MANAGEMENT */
/* OR SMS MANAGED TAPE EXIST */
/*****

END /* TAPE ALLOCATION SELECT */

```

Mandatory - Optical Allocations

```

/*****
/* INCLUDE ALL OPTICAL ALLOCATIONS IN THIS BLOCK */
*****/

      WHEN (&ACSENVIR EQ &VALID_OPTICAL_ACSENVIR)

          SELECT /* OPTICAL ALLOCATIONS */

              WHEN (&ACSENVIR EQ &VALID_OPTICAL_ACSENVIR)
                  SET &MGMTCLAS = &MGMTCLAS

/*****
/* TAILOR AND ADD OPTICAL STORE FUNCTION HERE IF OPTICAL DATA EXISTS */
*****/

/*****
/* TAILOR AND ADD OPTICAL TRANSITION AND CHANGE FUNCTIONS HERE IF
/* OPTICAL DATA EXISTS */
*****/

/*****
/* ADD OTHERWISE FOR OPTICAL DATA HERE IF OPTICAL DATA EXISTS */
*****/

      END /* OPTICAL ALLOCATION SELECT */

/*****
/* THIS OTHERWISE IS FOR ALLOCATIONS OTHER THAN DASD, TAPE, OR
/* OPTICAL */
*****/

      OTHERWISE /* UNEXPECTED DEVICES */

          DO
              SET &MGMTCLAS = ¢_____¢
              EXIT
          END /* DO */

      END /* SELECT */

END /* PROC */
```

Optional - JCL Exceptions

```
/* ***** */
/* TAILOR AND SET MGMTCLAS FOR ALLOCATION EXCEPTIONS */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH SET OF SPECIAL USERS THAT ARE */
/* ALLOWED TO REQUEST SMS SERVICE THROUGH MGMTCLAS AT ALLOCATION */
/* ***** */

FILTLIST &SPECIAL_MC_USER INCLUDE
(  _____  ,  _____  ,  _____  )

FILTLIST &SPECIAL_MGMTCLAS INCLUDE
(  _____  ,  _____  ,  _____  )

WHEN ( ( &USER EQ &SPECIAL_MC_USER )
AND ( &MGMTCLAS EQ &SPECIAL_MGMTCLAS ) )

DO
    SET &MGMTCLAS = &MGMTCLAS
    EXIT
END /* DO */
```

Optional - Concurrent Copy Exceptions

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR CONCURRENT COPY IF USED          */  
/* *****  
  
FILTLIST &CONCURRENT_COPY_DSN INCLUDE  
( _____, _____, _____,  
  _____, _____, _____,  
  _____, _____, _____ )  
  
WHEN ( &DSN EQ &CONCURRENT_COPY_DSN )  
  
  DO  
    SET &MGMTCLAS =ç_____ç  
    EXIT  
  END /* DO */
```

Optional - Data Set List Exceptions

```
/* ***** */
/* TAILOR AND SET MGMTCLAS FOR DATA SET LIST EXCEPTIONS          */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH SET OF EXCEPTION MANAGEMENT CLASSES */
/* BASED ON DATA SET NAME LISTS                                  */
/* ***** */

      FILTLIST &EXCEPTION_MC_SERVICE_DSN INCLUDE
      ( _____ / _____ / _____ /
        _____ / _____ / _____ /
        _____ / _____ / _____ / )

      WHEN ( &DSN EQ &EXCEPTION_MC_SERVICE_DSN )

          DO
              SET &MGMTCLAS =ç _____ ç
              EXIT
          END /* DO */
```

Optional - Data Set Size

```
/* *****  
/* TAILOR AND SET MGMTCLAS WHEN YOU WANT TO GIVE DIFFERENT SERVICE */  
/* TO LARGE DATA SETS VERSUS AVERAGE SIZED DATA SETS */  
/* *****  
/* YOU CAN ADD THIS LOGIC BLOCK TO ANY SET MANAGEMENT CLASS SUBTYPES */  
/* *****  
  
    WHEN ((&SIZE GT ___MB)  
        OR (&MAXSIZE GT ___MB))  
  
        DO  
            SET &MGMTCLAS = ¢_____¢  
            EXIT  
        END /* END DO */
```


Optional - Temp Short Duration

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR TEMP SHORT DURATION SUBTYPES IF THEY */  
/* EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SUBTYPE OF TEMP SHORT DURATION THAT */  
/* HAS A DIFFERENT MANAGEMENT CLASS */  
/* *****  
  
    FILTLIST &TEMPPERM_DATA_DSN INCLUDE  
    ( _____, _____, _____,  
      _____, _____, _____,  
      _____, _____, _____ )  
  
    WHEN ( &DSN EQ &TEMPPERM_DATA_DSN )  
  
        DO  
            SET &MGMTCLAS =ç_____ç  
            EXIT  
        END /* DO */
```

Optional - TSO Logon and List

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR TSO LOGON AND LIST SUBTYPES IF THEY */  
/* HAVE DIFFERENT MANAGEMENT CLASSES FROM OTHER TSO SUBTYPES */  
/* *****  
  
FILTLIST &TSO_LOGON_DATA_DSN INCLUDE  
(*.ISPF.PROFILE, _____, _____,  
_____, _____, _____)  
  
FILTLIST &TSO_LIST_DATA_DSN INCLUDE  
(**.LIST, **.OUTLIST, **.LINKLIST,  
_____, _____, _____)  
  
WHEN (&DSN EQ &TSO_LOGON_DATA_DSN)  
  
DO  
SET &MGMTCLAS =ç_____ç  
EXIT  
END /* DO */  
  
WHEN (&DSN EQ &TSO_LIST_DATA_DSN)  
  
DO  
SET &MGMTCLAS =ç_____ç  
EXIT  
END /* DO */
```

Optional - TSO

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR TSO SUBTYPES */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH TSO SUBTYPE THAT HAS A DIFFERENT */  
/* MANAGEMENT CLASS AND IS OTHER THAN LOGON OR LIST SUBTYPES */  
/* *****  
  
    FILTLIST &TSO_DATA_HLQ INCLUDE  
    (¢_____¢,¢_____¢,¢_____¢)  
  
    WHEN (&HLQ EQ &TSO_DATA_HLQ)  
  
        DO  
            SET &MGMTCLAS =¢_____¢  
            EXIT  
        END /* DO */
```

Optional - Test

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR TEST SUBTYPES */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH TEST SUBTYPES THAT HAS A DIFFERENT */  
/* MANAGEMENT CLASS */  
/* *****  
  
FILTLIST &TEST_DATA_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )  
  
WHEN ( &DSN EQ &TEST_DATA_DSN )  
  
  DO  
    SET &MGMTCLAS =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Batch Production

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR BATCH PRODUCTION SUBTYPES */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH BATCH PRODUCTION SUBTYPE THAT HAS A */  
/* DIFFERENT MANAGEMENT CLASS */  
/* *****  
  
FILTLIST &BATCH_PROD_DATA_DSN INCLUDE  
( _____ / _____ / _____ /  
  _____ / _____ / _____ /  
  _____ / _____ / _____ )  
  
WHEN ( &DSN EQ &BATCH_PROD_DATA_DSN )  
  
  DO  
    SET &MGMTCLAS =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Online Production

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR ONLINE PRODUCTION SUBTYPES */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH ONLINE PRODUCTION SUBTYPE THAT HAS A */  
/* DIFFERENT MANAGEMENT CLASS */  
/* *****  
  
FILTLIST &ONLINE_PROD_DATA_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )  
  
WHEN ( &DSN EQ &ONLINE_PROD_DATA_DSN )  
  
  DO  
    SET &MGMTCLAS =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Other

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR OTHER SUBTYPES */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH OTHER SUBTYPE THAT HAS A DIFFERENT */  
/* MANAGEMENT CLASS */  
/* *****  
  
    FILTLIST &_____DATA_DSN INCLUDE  
    (_____,_____,_____,  
     _____,_____,_____,  
     _____,_____,_____)  
  
    WHEN (&DSN EQ &_____DATA_DSN)  
  
        DO  
            SET &MGMTCLAS =ç_____ç  
            EXIT  
        END /* DO */
```

Optional - Tape Mount Management

```
/* ***** */
/* TAILOR AND SET MGMTCLAS FOR TAPE MOUNT MANAGEMENT SUBTYPES IF */
/* THEY EXIST */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH TAPE MOUNT MANAGEMENT SUBTYPE THAT HAS */
/* A DIFFERENT MANAGEMENT CLASS */
/* ***** */

FILTLIST &TAPE_MOUNT_MANAGEMENT_DSN INCLUDE
( _____ , _____ , _____ ,
  _____ , _____ , _____ ,
  _____ , _____ , _____ )

FILTLIST &TAPE_MOUNT_MANAGEMENT_DATACLAS INCLUDE
( ¢ _____ ¢ , ¢ _____ ¢ , ¢ _____ ¢ )

WHEN ( ( &DSN EQ &TAPE_MOUNT_MANAGEMENT_DSN )
OR ( &DATACLAS EQ &TAPE_MOUNT_MANAGEMENT_DATACLAS ) )

DO
    SET &MGMTCLAS = ¢ _____ ¢
    EXIT
END /* DO */
```


Optional - SMS-Managed Tape

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR SMS MANAGED TAPE SUBTYPES IF THEY */  
/* EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SMS MANAGED TAPE SUBTYPE THAT HAS A */  
/* DIFFERENT MANAGEMENT CLASS */  
/* *****  
  
    WHEN (&DSN EQ &SMS_MANAGED_TAPE_DSN)  
  
        DO  
            SET &MGMTCLAS =ç_____ç  
            EXIT  
        END /* DO */
```

Optional - Otherwise Tape

```
/* *****  
/* OTHERWISE STATEMENT FOR TAPE ALLOCATIONS *  
/* *****  
  
    OTHERWISE  
  
        DO  
            SET &MGMTCLAS = ¢_____¢  
            EXIT  
        END /* DO */
```

Optional - Optical Store

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR STORE PROCESSING OF OPTICAL SUBTYPES */  
/* IF THEY EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH OPTICAL SUBTYPE THAT HAS A DIFFERENT */  
/* MANAGEMENT CLASS AND STORE PROCESSING LOGIC */  
/* *****  
  
    FILTLIST &OPTICAL_DSN INCLUDE  
    ( _____ , _____ , _____ ,  
      _____ , _____ , _____ ,  
      _____ , _____ , _____ )  
  
    WHEN ((&DSN EQ &OPTICAL_DSN)  
          AND (&ACSENVIR EQ ¢STORE¢))  
  
        DO  
            SET &MGMTCLAS = ¢ _____ ¢  
            EXIT  
        END /* DO */
```

Optional - Optical Transition and Change

```
/* *****  
/* TAILOR AND SET MGMTCLAS FOR CTRAN AND CHANGE PROCESSING OF */  
/* OPTICAL SUBTYPES IF THEY EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH OPTICAL SUBTYPE THAT HAS A DIFFERENT */  
/* MANAGEMENT CLASS AND CTRAN OR CHANGE PROCESSING LOGIC */  
/* *****
```

```
    WHEN ((&DSN EQ &OPTICAL_DSN)  
          AND (&ACSENVIR EQ ¢CTRANS¢))
```

```
        DO  
            SET &MGMTCLAS = ¢_____¢  
            EXIT  
        END /* DO */
```

```
    WHEN ((&DSN EQ &OPTICAL_DSN)  
          AND (&ACSENVIR EQ ¢CHANGE¢))
```

```
        DO  
            SET &MGMTCLAS = ¢_____¢  
            EXIT  
        END /* DO */
```

Optional - Otherwise Optical

```
/* *****  
/* OTHERWISE STATEMENT FOR OPTICAL ALLOCATIONS *  
/* *****
```

```
OTHERWISE
```

```
DO  
    SET &MGMTCLAS = ¢_____¢  
    EXIT  
END /* DO */
```

Sample - Management Class ACS Code

```
PROC &MGMICLAS

/*****
/* START OF MANAGEMENT CLASS PROC */
/*****
/* CHANGE HISTORY */
/* DATE PERSON DESCRIPTION OF CHANGE */
/*****
/* 95/05/06 DEF INITIAL IMPLEMENTATION FOR STAGE I */
/* DATA */
/* 95/02/05 DEF MINIMAL CONFIGURATION */
/*****

/*-----*/

FILTLIST &VALID_DASD_UNIT INCLUDE
(ç3380ç,ç3390ç,çSYSDAç,çSYSALLDAç,çç,

FILTLIST &VALID_OPTICAL_ACSENVIR INCLUDE
(çSTOREç,çCHANGEç,çCTRANSç)

FILTLIST &SPECIAL_MC_USER INCLUDE
(çFREEMANç)

FILTLIST &SPECIAL_MGMICLAS INCLUDE
(çç)

FILTLIST &TSO_LOGON_DATA_DSN INCLUDE
(*.ISFP.PROFILE)

FILTLIST &SCRATCH_DATA_DSN INCLUDE
(**.LIST,**.OUTLIST,**.LINKLIST,*T%%TEMP*.**)

FILTLIST &TSO_DATA_HLQ INCLUDE
(ç%%T%%ç)

FILTLIST &TEST_DATA_DSN INCLUDE
(*.T*.**)

FILTLIST &BATCH_PROD_DATA_DSN INCLUDE
(*.P*.**)

FILTLIST &CRITICAL_PERFORMANCE_DSN INCLUDE
(FREEMAN.P*.NEEDFAST)

FILTLIST &BATCH_PROD_WORK_DSN INCLUDE
(*.P%%MF*.**)


```

Sample - Management Class ACS Code ...

```
FILTLIST &VALID_TAPE_UNIT INCLUDE
(ç3490ç,ç3480ç,ç3590ç,TAPE*,çT3480ç,çT3490ç,
çT3590ç,çAFF=ç)
```

```
FILTLIST &TAPE_MOUNT_MANAGEMENT_DSN INCLUDE
(çç)
```

```
FILTLIST &SMS_MANAGED_TAPE_DSN INCLUDE
(çç)
```

```
FILTLIST &OPTICAL_DSN INCLUDE
(çç)
```

```
/** _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ */
```

```
SELECT /* ALLOCATION TYPE */
```

```
/*
*****
/* INCLUDE ALL STANDARDS ENFORCEMENT LOGIC IN THIS BLOCK IN */
/* ONE AND ONLY ONE OF EITHER THE DATA CLASS OR STORAGE ACS */
/* ROUTINE, LEAVE BLOCK IN OTHER ACS ROUTINES FOR CONSISTENCY */
*****
```

```
/*
*****
/* START STANDARDS ENFORCEMENT LOGIC BLOCK */
*****
/*
*****
/* END STANDARDS ENFORCEMENT LOGIC BLOCK */
*****
```

```
/*
*****
/* INCLUDE ALL DASD ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* BOTH DASD ALLOCATIONS TO SMS AND TO NON SMS */
*****
```

```
WHEN ((&UNIT EQ &VALID_DASD_UNIT)
AND (&ACSENVIR NE &VALID_OPTICAL_ACSENVIR))
```

```
SELECT /* DASD DATA ALLOCATION */
```

Sample - Management Class ACS Code ...

```
/* *****  
/* SYSTEM TEMPORARY DATA SETS ARE NOT ASSIGNED A MANAGEMENT CLASS */  
/* *****  
/* DUPLICATE THIS BLOCK FOR EACH SET OF SPECIAL USERS THAT ARE */  
/* ALLOWED TO REQUEST SMS SERVICE THROUGH JCL STORCLAS AND MGMTCLAS */  
/* PARAMETERS, THIS MAY INCLUDE ALL USERS FOR THOSE MGMTCLAS */  
/* EXCEPTIONS THAT YOU ARE WILLING TO LET ALL USERS USE */  
/* *****
```

```
WHEN ((&USER EQ &SPECIAL_MC_USER)  
AND (&MGMTCLAS EQ &SPECIAL_MGMTCLAS))
```

```
DO  
    SET &MGMTCLAS = &MGMTCLAS  
    EXIT  
END /* DO */
```

```
/* *****  
/* INCLUDE TSO DATA (IDENTIFIED BY HLQ) FOR SMS IN THIS BLOCK */  
/* *****  
/* DUPLICATE THIS BLOCK FOR EACH SET OF EXCEPTION SERVICES, BASED ON */  
/* DATA CLASSIFICATION, THAT ARE CONTROLLED BY DATA SET NAME LISTS */  
/* *****
```

```
WHEN (&DSN EQ &TSO_LOGON_DATA_DSN)
```

```
DO  
    SET &MGMTCLAS = ꞀMCNOMIGꞀ  
    EXIT  
END /* DO */
```

```
WHEN (&DSN EQ &SCRATCH_DATA_DSN)
```

```
DO  
    SET &MGMTCLAS = ꞀMCINTERMꞀ  
    EXIT  
END /* DO */
```


Sample - Management Class ACS Code ...

```
/* ***** */
/* REPEAT THIS BLOCK FOR EACH DATA CLASSIFICATION TYPE OF TSO DATA, */
/* USING THE FILTLIST AND LOGIC TEST IDENTIFIED FOR THE DATA */
/* CLASSIFICATION DIAGRAM AND ASSIGN THE DEFAULT MGMTCLAS DEFINED */
/* DURING THE MANAGEMENT CLASS DESIGN */
/* ***** */
```

```
WHEN (&HLQ EQ &TSO_DATA_HLQ)
```

```
DO
    SET &MGMTCLAS = %MCEARLY%
    EXIT
END /* DO */
```

```
/* ***** */
/* INCLUDE TEST DATA (IDENTIFIED BY DSN) FOR SMS IN THIS BLOCK */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH DATA CLASSIFICATION */
/* TYPE OF TEST DATA, USING THE FILTLIST AND LOGIC */
/* TEST IDENTIFIED FOR THE DATA CLASSIFICATION */
/* DIAGRAM AND ASSIGN THE DEFAULT MGMTCLAS DEFINED */
/* DURING THE MANAGEMENT CLASS DESIGN */
/* ***** */
```

```
WHEN ((&DSN EQ &TEST_DATA_DSN)
AND (&DSTYPE EQ %GDS%))
```

```
DO
    SET &MGMTCLAS = %MCNOBACK%
    EXIT
END /* DO */
```

```
WHEN (&DSN EQ &TEST_DATA_DSN)
```

```
DO
    SET &MGMTCLAS = %MCSTAND%
    EXIT
END /* DO */
```

Sample - Management Class ACS Code ...

```
/* *****  
/* INCLUDE BATCH PRODUCTION DATA ( IDENTIFIED *  
/* BY DSN) FOR SMS IN THIS BLOCK *  
/* *****  
/* REPEAT THIS BLOCK FOR EACH DATA CLASSIFICATION *  
/* TYPE OF BATCH PRODUCTION DATA, USING THE FILTLIST *  
/* AND LOGIC TEST IDENTIFIED FOR THE DATA CLASSIFICATION *  
/* DIAGRAM AND ASSIGN THE DEFAULT MGMTCLAS DEFINED *  
/* DURING THE MANAGEMENT CLASS DESIGN *  
/* *****  
/* FOR ANY OTHER DATA TYPES NOT IDENTIFIED FOR *  
/* SMS MANAGEMENT, PLACE THEM IN NON SMS STORAGE *  
/* *****
```

```
WHEN (&DSN EQ &CRITICAL_PERFORMANCE_DSN)
```

```
DO  
    SET &MGMTCLAS = &MCNOMIG&  
    EXIT  
END /* DO */
```

```
WHEN (&DSN EQ &BATCH_PROD_WORK_DSN)
```

```
DO  
    SET &MGMTCLAS = &MCEARLY&  
    EXIT  
END /* DO */
```

```
WHEN ((&DSN EQ &BATCH_PROD_DATA_DSN)  
AND (&DSTYPE EQ &GDS&))
```

```
DO  
    SET &MGMTCLAS = &MCEARLY&  
    EXIT  
END /* DO */
```

```
WHEN (&DSN EQ &BATCH_PROD_DATA_DSN)
```

```
DO  
    SET &MGMTCLAS = &MCLATE&  
    EXIT  
END /* DO */
```

Sample - Management Class ACS Code ...

```
OTHERWISE /* NONMANAGED DASD DATA */
```

```
DO
    SET &MGMTCLAS = %MCEARLY%
    EXIT
END /* DO */
```

```
END /* DASD ALLOCATION SELECT */
```

```
/*
*****
/* INCLUDE ALL TAPE ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* BOTH TAPE ALLOCATIONS TO SMS AND TO NON SMS */
/* PRIOR TO SMS MANAGED TAPE OR TAPE MOUNT MANAGEMENT */
/* ALL TAPE ALLOCATION WOULD BE TO NON SMS */
*****
/*
*****
/* START TAPE ALLOCATION LOGIC BLOCK */
*****
/*
*****
/* END TAPE ALLOCATION LOGIC BLOCK */
*****
*/
```

Sample - Management Class ACS Code ...

```

/*****
/* INCLUDE ALL OPTICAL ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* LOGIC FROM PRODUCTS LIKE IAFIC OR IMAGE PLUS HERE */
*****/

/*****
/* START OPTICAL ALLOCATION LOGIC BLOCK */
*****/
/*
*****/
/* END OPTICAL ALLOCATION LOGIC BLOCK */
*****/

        OTHERWISE /* NONMANAGED DEVICES */

                DO
                        SET &MGMTCLAS =%MCEARLY%
                        EXIT
                END /* DO */

        END /* SELECT */

END /* PROC */
```

Data Class Design

Data Class Design

Topics

- Design considerations
- Data class questions
- Definition of data classes
- Update of data classification

Data Class Benefits

- Implement standards for both SMS- and non-SMS-managed data sets

- Support new DFSMS unique functions
 - Compression
 - Sequential data striping
 - PDSEs
 - HFS data
 - Extended addressability VSAM KSDS

- Enhance existing allocation parameters
 - Replace pattern DSCBs
 - Extend data sets to be multivolume
 - Enforce secondary allocation

- Select data for TMM or SMS-managed tape
 - Supply space for TMM
 - Select media type
 - Select transport and cartridge type

Data Class Considerations

- Order of parameter use
 1. Allocation specifications
 2. DCB attributes of referenced data set
 3. Data class definition

- Required for selected DFSMS functions
 - Compression
 - Sequential data striping
 - PDSEs
 - HFS data
 - Extended addressability VSAM KSDS
 - TMM
 - SMS-managed tape

Data Class Recommendations

- During initial DFSMS FIT implementation phase
 - Use only if required for standards enforcement or to support DFSMS functions
 - Limit JCL enhancements to JCL selection from the starter set
- During later DFSMS implementation phases
 - Use DATACLAS=YES
 - Enhance allocation by filling in missing but desired values
 - Assign DATACLAS based on LLQ

Data Class Recommendations ...

- Use DATA SET NAME TYPE for
 - Sequential striping data - (EXTENDED,R) or (EXTENDED,P)
 - Compressed data - (EXTENDED,R) or (EXTENDED,P)
 - VSAM KSDS extended addressability data - (EXTENDED,R) or (EXTENDED,P)
 - HFS data - HFS
 - PDSE libraries - LIBRARY
- Use COMPACTION=YES for compressed data
- Use EXTENDED ADDRESSABILITY=YES for VSAM KSDS extended addressability data
- Use for TMM data
 - AVGREC
 - SPACE PRIMARY
 - SPACE SECONDARY
 - VOLUME COUNT
- Use for SMS-managed tape data
 - MEDIA TYPE
 - RECORDING TECHNOLOGY

Data Class Design Questions

1. Do you plan to use any of the following DFSMS DASD functions?
 - Compression or sequential data striping
 - PDSEs
 - HFS data
 - Extended addressability VSAM KSDS

2. If TMM is used, which functions are needed?
 - Active tape data sets to DASD using TMM
 - Smaller tape data sets to tape using TMM
 - Larger tape data sets to tape using TMM

3. If SMS-managed tape is used, what type of tape?
 - 18-track tapes
 - Standard cartridge media and 36 track tapes
 - Enhanced cartridge media and 36 track tapes
 - 128-track tapes

4. Do you need to fill in allocation parameters?
 - Allow for expansion to a multivolume data set
 - Always provide secondary space allocation

5. Do you want to make the starter set's data classes available through JCL request?

Data Class Design Table

Data Type									
Classification box									
DASD Functions									
Compress. or striping									
PDSE									
HFS									
Extended address.									
TMM Use									
Active to DASD									
Small to tape									
Large to tape									
SMS-Tape Use									
18 track									
36 track and standard cartridge									
36 track and enhanced cartridge									
128 track									
JCL Fill-In									
Multivolume									
Secondary space									
Starter Set									
DATACLAS on JCL									
Data Class ID									

Sample Data Class Answers

Data Type	WK	TS	TS	TS	T	T	T		
Classification box	1	2	3	4	5	6	7		
DASD Functions									
Compress. or striping									
PDSE									
HFS									
Extended address.									
TMM Use									
Active to DASD									
Small to tape									
Large to tape									
SMS-Tape Use									
18 track									
36 track and standard cartridge									
36 track and enhanced cartridge									
128 track									
JCL Fill-In									
Multi-volume									
Secondary space									
Starter Set									
DATACLAS on JCL	X	X	X	X	X	X	X		
Data Class ID	SS	SS	SS	SS	SS	SS	SS		

Sample Data Class Answers ...

Data Type	B P	B P	B P	B P	EX	EX	EX		
Classification box	8	9	10	11	12	13	14		
DASD Functions									
Compress. or striping									
PDSE									
HFS									
Extended address.									
TMM Use									
Active to DASD									
Small to tape									
Large to tape									
SMS-Tape Use									
18 track									
36 track and standard cartridge									
36 track and enhanced cartridge									
128 track									
JCL Fill-In									
Multi-volume									
Secondary space									
Starter Set									
DATACLAS on JCL	X	X	X	X	X	X			
Data Class ID	SS	SS	SS	SS	SS	SS			

Data Class Naming

1.

2.

3.

4.

5.

6.

7.

8.

9.

Sample Data Class Naming

1. DCDIRECT
2. DCENTRY
3. DCHFSDS
4. DCKEYED
5. DCLINEAR
6. DCNONTMM
7. DCTMMACT (tmmactv)
8. DCTMMBKP (tmmbkup)
9. DCTMMTMP (tmmtemp)
10. DCTAPACT (tapactv)
11. DCTAPBKP (tapbkup)
12. DCTAPOFF (taposite)
13. DCTAPTMP (taptemp)
14. DCGDGF80
15. DCGDGV04 (gdgv104)
16. DCDATAF
17. DCDATAV
18. DCLIST (listing)
19. DCLOADLB (loadlib)
20. DCSRCFLB (srcflib)
21. DCSRCVLB (srcvlib)
22. DCCOMP (datacomp)
23. DCHSMDC

Data Class Protection Design

Data Class	JCL Use	Protected	By What

Sample Data Class Protection

Data Class	JCL Use	Protected	By What
DCDIRECT	Yes	No	
DCENTRY	Yes	No	
DCHFSDS	Yes	No	
DCKEYED	Yes	No	
DCLINEAR	Yes	No	
DCNONTMM	Yes	No	
DCTMMACT	Yes	No	
DCTMMBKP	Yes	No	
DCTMMTMP	Yes	No	
DCTAPACT	Yes	No	
DCTAPBKP	Yes	No	
DCTAPOFF	Yes	No	
DCTAPTMP	Yes	No	

Sample Data Class Protection ...

Data Class	JCL Use	Protected	By What
DCGDGF80	Yes	No	
DCGDGV04	Yes	No	
DCDATAF	Yes	No	
DCDATAV	Yes	No	
DCLIST	Yes	No	
DCLOADLB	Yes	No	
DCSRCFLB	Yes	No	
DCSRCVLB	Yes	No	
DCCOMP	Yes	No	
DCHSMDC	Yes	No	

Data Class Parameter Design

DATACLAS NAME	RECORG	RECFM	LRECL	KEYLEN	KEYOFF

Data Class Parameter Design ...

DATACLAS NAME	AVGREC	AVG VALUE	SPACE PRIMARY	SPACE SECONDARY

Data Class Parameter Design ...

DATACLAS NAME	SPACE DIRECTORY	RETPD EXPDT	VOLUME COUNT	IMBED	REPLICATE

Data Class Parameter Design ...

DATACLAS NAME	CISIZE DATA	% FREE SPACE CA	% FREE SPACE CI	SHARE XREGION	SHARE XSYSTEM

Data Class Parameter Design ...

DATACLAS NAME	DATA SET NAME TYPE	COMPACTION	MEDIA TYPE	RECORDING TECHNOLOGY

Sample Data Class Parameters

- Parameters of starter set's data classes used
- Starter set's data classes documented in Appendix B of *MVS/ESA Storage Management Library Implementing System-Managed Storage*, SC26-3123.

Sample - Work Data

WK1

DSTYPE=TEMP

Work

SC=SCFAST

SG=SGWORK

MC=N/A

DC=optionally selected on JCL

Sample - TSO Data

HLQ=%%T%%

TSO

OTHERWISE

DSN=list

DSN=list

TS2

Normal
Data

TS3

LIST, LOG,
ISPF*TEMP.CNTL
type data

TS4

Used at
LOGON

SC=SCNORMAL
SG=SGTSO
MC=MCEARLY
DC=optionally
selected on JCL

SC=SCNORMAL
SG=SGTSO
MC=MCINTERM
DC=optionally
selected on JCL

SC=SCNORMAL
SG=SGTSO
MC=MCNOMIG
DC=optionally
selected on JCL

Sample - Test Data

	Test	HLQ=application list 2LQ=T*	
	OTHERWISE	2LQ=T%%TEMP*	DSTYPE=GDS
T5	T6	T7	
Normal	One Day	GDGs	
SC=SCNORMAL SG=SGISO MC=MCSTAND DC=optionally selected on JCL	SC=SCNORMAL SG=SGISO MC=MCINTERM DC=optionally selected on JCL	SC=SCNORMAL SG=SGISO MC=MCNOBACK DC=optionally selected on JCL	

Sample - Exceptions

Exceptions

	DSN=list and JCL	DSN=list and JCL	JCL STORCLAS=SCNOSMS
EX12	Normal Performance Hand Placed	EX13 Critical Performance Hand Placed	EX14 Non SMS
	SC=SCSPECN SG=as usual MC=as usual DC=optionally selected on JCL	SC=SCSPECF SG=as usual MC=as usual DC=optionally selected on JCL	SC=☐☐

Data Class ACS Coding

Data Class ACS Considerations

- Assign JCL replacement through JCL
- Assign JCL fill-in and support for functions that require data class through ACS logic
- Eventually assign JCL replacement through ACS logic when DATACLAS=YES from JCL
- RACF defaults not used with DFSMS FIT
- Data class ACS routine not invoked for optical data

Mandatory - DASD Allocations and Exceptions

```
/*  
/* TAILOR AND ADD DATA SET SIZE STANDARD HERE IF USED */  
*/
```

```
/*  
/* INCLUDE ALL DASD ALLOCATIONS IN THIS BLOCK */  
*/
```

```
FILTLIST &VALID_DASD_UNIT INCLUDE  
( &3380, &3390, &SYSDA, &SYSALLDA, &,  
  &_____, &_____, &_____ )
```

```
FILTLIST &VALID_OPTICAL_ACSENVIR INCLUDE  
( &STORE, &CHANGE, &CTRANS )
```

```
WHEN ( (&UNIT EQ &VALID_DASD_UNIT)  
AND (&ACSENVIR NE &VALID_OPTICAL_ACSENVIR) )
```

```
SELECT /* DASD DATA ALLOCATION */
```

```
/*  
/* TAILOR AND ADD JCL EXCEPTIONS HERE IF THEY ARE USED */  
*/
```

```
/*  
/* TAILOR AND ADD DATA SET LIST EXCEPTIONS HERE IF THEY ARE USED */  
/* INCLUDE SEQUENTIAL STRIPING, COMPRESSION, PDSE, HFS, VSAM */  
/* EXTENDED ADDRESSING, SECONDARY SPACE, AND MULTIVOLUME */  
*/
```

Mandatory - Temp Data Type

```
/*  
/* TAILOR AND ADD SYSTEM TEMPORARY SUBTYPE HERE IF THEY EXIST */  
*/
```

```
/*  
/* TAILOR AND ADD TEMP SHORT DURATION HERE IF IT EXISTS */  
*/
```

Mandatory - TSO and Test Data Types

```
/*  
*****  
/* TAILOR AND ADD TSO LOGON AND LIST SUBTYPES HERE IF THEY EXIST      */  
/* AND HAVE DATA CLASSES DIFFERENT THAN OTHER TSO SUBTYPES          */  
*****  
*/
```

```
/*  
*****  
/* TAILOR AND ADD OTHER TSO SUBTYPES HERE IF THEY EXIST              */  
*****  
*/
```

```
/*  
*****  
/* TAILOR AND ADD TEST SUBTYPES HERE IF THEY EXIST                    */  
*****  
*/
```

Mandatory - Production Data Types

```
/*  
/* TAILOR AND ADD BATCH PRODUCTION SUBTYPES HERE IF THEY EXIST */  
*/
```

```
/*  
/* TAILOR AND ADD ONLINE PRODUCTION SUBTYPES HERE IF THEY EXIST */  
*/
```

```
/*  
/* TAILOR AND ADD OTHER SUBTYPES HERE IF THEY EXIST */  
*/
```

```
/*  
/* TAILOR AND SET DATACLAS FOR NONMANAGED DASD DATA SETS */  
*/
```

```
OTHERWISE /* NONMANAGED DASD DATA */
```

```
DO  
    SET &DATACLAS = ¢_____¢  
    EXIT  
END /* DO */
```

```
END /* DASD ALLOCATION SELECT */
```

Mandatory - Tape Allocations

```

/*****
/* INCLUDE ALL TAPE ALLOCATIONS IN THIS BLOCK */
/*****

FILTLIST &VALID_TAPE_UNIT INCLUDE
(ç3490ç,ç3480ç,ç3590ç,TAPE*,çT3480ç,çT3490ç,çT3590ç,çAFF=ç,
ç_____ç,ç_____ç,ç_____ç)

WHEN (&UNIT EQ &VALID_TAPE_UNIT)

SELECT /* TAPE ALLOCATIONS */

        WHEN (&UNIT EQ &VALID_TAPE_UNIT)
        SET &DATACLAS = &DATACLAS

/*****
/* TAILOR AND ADD TAPE MOUNT MANAGEMENT SUBTYPES HERE IF THEY EXIST */
/*****

/*****
/* TAILOR AND ADD SMS MANAGED TAPE SUBTYPES HERE IF THEY EXIST */
/*****

/*****
/* ADD OTHERWISE FOR TAPE DATA HERE IF EITHER TAPE MOUNT MANAGEMENT */
/* OR SMS MANAGED TAPE SUBTYPES EXIST */
/*****

END /* TAPE ALLOCATION SELECT */

```

Mandatory - Optical Allocations

```

/*****
/* INCLUDE AN EMPTY OPTICAL ALLOCATIONS BLOCK HERE FOR COMPATIBILITY */
/* WITH THE OTHER ACS ROUTINES */
*****/

/*****
/* START OPTICAL ALLOCATION LOGIC BLOCK */
*****/
/*
*****/
/* END OPTICAL ALLOCATION LOGIC BLOCK */
*****/

/*****
/* THIS OTHERWISE IS FOR ALLOCATIONS OTHER THAN DASD, TAPE, OR */
/* OPTICAL */
*****/

        OTHERWISE /* NONMANAGED DEVICES */

                DO
                        SET &DATACLAS = ¢_____¢
                        EXIT
                END /* DO */

        END /* SELECT */

END /* PROC */
```

Optional - Data Set Naming Standard

```
/* ***** */
/* TAILOR TO ENFORCE DATA SET NAMING STANDARDS */
/* ***** */

FILTLIST &STANDARD_NAME_DSN INCLUDE
(_____,_____,_____
 _____,_____,_____
 _____,_____,_____ )

WHEN (&DSN NE @@)

SELECT /* STANDARDS ENFORCEMENT */

    WHEN (&DSN NE &STANDARD_NAME_DSN)

        DO
            WRITE @(&DSN@) FAILS NAMING STANDARD@
            EXIT CODE(16)
        END /* DO */

    OTHERWISE /* OTHER DASD DATA */

        DO
            END /* DO */

END /* STANDARDS ENFORCEMENT SELECT */
```


Optional - Volume Fencing Standard

```
/*
*****
/* TAILOR TO ENFORCE VOLUME FENCING STANDARDS
*****
*/

FILTLIST &OKAY_____USER INCLUDE
(ϕ_____ϕ,ϕ_____ϕ,ϕ_____ϕ)

FILTLIST &OKAY_____VOL INCLUDE
(ϕ_____ϕ,ϕ_____ϕ,ϕ_____ϕ)

WHEN (&DSN NE ϕϕ)

    SELECT /* STANDARDS ENFORCEMENT */

        WHEN ((&USER NE &OKAY_____USER)
              AND (&ALLVOL NE &OKAY_____VOL))

            DO
                WRITE ϕ(ϕ&USERϕ) CANNOT USE VOUMEe (ϕ&ANYVOLϕ)ϕ
                EXIT CODE(16)
            END /* DO */

        OTHERWISE /* OTHER DASD DATA */

            DO
            END /* DO */

    END /* STANDARDS ENFORCEMENT SELECT */
```

Optional - Data Set Size Standard

```
/* *****  
/* TAILOR TO ENFORCE DATA SET SIZE STANDARDS *  
/* *****  
    WHEN (&DSN NE ¢¢)  
  
        SELECT /* STANDARDS ENFORCEMENT */  
  
            WHEN ((&SIZE GT ____MB)  
                OR (&MAXSIZE GT _____MB))  
  
                DO  
                    WRITE ¢(¢&DSN¢) EXCEEDS SIZE STANDARD¢  
                    EXIT CODE(16)  
                END /* DO */  
  
            OTHERWISE /* OTHER DASD DATA */  
  
                DO  
                END /* DO */  
  
        END /* STANDARDS ENFORCEMENT SELECT */
```

Optional - JCL Exceptions

```
/* ***** */
/* TAILOR AND SET DATACLAS FOR ALLOCATION EXCEPTIONS */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH SET OF SPECIAL USERS THAT ARE */
/* ALLOWED TO REQUEST SMS SERVICE THROUGH DATACLAS AT ALLOCATION */
/* ***** */

FILTLIST &SPECIAL_DC_USER INCLUDE
(  _____  ,  _____  ,  _____  )

FILTLIST &SPECIAL_DATACLAS INCLUDE
(  _____  ,  _____  ,  _____  )

WHEN ( ( &USER EQ &SPECIAL_DC_USER )
AND ( &DATACLAS EQ &SPECIAL_DATACLAS ) )

DO
    SET &DATACLAS = &DATACLAS
    EXIT
END /* DO */
```

Optional - Sequential Striping Exception

```
/* *****  
/* TAILOR AND SET DATACLAS FOR SEQUENTIAL STRIPING IF USED          */  
/* *****  
  
    FILTLIST &SEQUENTIAL_STRIPING_DSN INCLUDE  
    ( _____ / _____ / _____ /  
      _____ / _____ / _____ /  
      _____ / _____ / _____ / )  
  
    WHEN ( &DSN EQ &SEQUENTIAL_STRIPING_DSN )  
  
        DO  
            SET &DATACLAS =ç _____ ç  
            EXIT  
        END /* DO */
```

Optional - Compression Exception

```
/* *****  
/* TAILOR AND SET DATACLAS FOR COMPRESSION IF USED *  
/* *****  
  
FILTLIST &COMPRESSION_DSN INCLUDE  
( _____, _____, _____,  
  _____, _____, _____,  
  _____, _____, _____ )  
  
WHEN ( &DSN EQ &COMPRESSION_DSN )  
  
  DO  
    SET &DATACLAS =ç_____ç  
    EXIT  
  END /* DO */
```

Optional - PDSE Exception

```
/* *****  
/* TAILOR AND SET DATACLAS FOR PDSE DATA IF USED *  
/* *****  
  
FILTLIST &PDSE_DSN INCLUDE  
( _____, _____, _____,  
  _____, _____, _____,  
  _____, _____, _____ )  
  
WHEN ( &DSN EQ &PDSE_DSN )  
  
  DO  
    SET &DATACLAS =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - HFS Exception

```
/* *****  
/* TAILOR AND SET DATACLAS FOR HFS DATA IF USED */  
/* *****  
  
/* DUPLICATE THIS BLOCK FOR EACH SET OF EXCEPTION */  
/* SERVICES, BASED ON DATA SET NAME LISTS */  
  
FILTLIST &HFS_DSN INCLUDE  
( _____, _____, _____,  
  _____, _____, _____,  
  _____, _____, _____ )  
  
WHEN (&DSN EQ &HFS_DSN)  
  
  DO  
    SET &DATACLAS =ç_____ç  
    EXIT  
  END /* DO */
```

Optional - VSAM Extended Addressing Exception

```
/* *****  
/* TAILOR AND SET DATACLAS FOR VSAM EXTENDED ADDRESSING IF USED      */  
/* *****  
  
    FILTLIST &EXTENDED_ADDRESSING_DSN INCLUDE  
    ( _____, _____, _____,  
      _____, _____, _____,  
      _____, _____, _____ )  
  
    WHEN ( &DSN EQ &EXTENDED_ADDRESSING_DSN )  
  
        DO  
            SET &DATACLAS =ç_____ç  
            EXIT  
        END /* DO */
```


Optional - Assign Secondary Space

```
/* *****  
/* TAILOR AND SET DATACLAS FOR ADDING SECONDARY SPACE IF USED      */  
/* *****  
  
    FILTLIST &SECONDARY_SPACE_DSN INCLUDE  
    ( _____ / _____ / _____ /  
      _____ / _____ / _____ /  
      _____ / _____ / _____ )  
  
    WHEN ( &DSN EQ &SECONDARY_SPACE_DSN )  
  
        DO  
            SET &DATACLAS =ç _____ ç  
            EXIT  
        END /* DO */
```

Optional - Allow Multivolume Exception

```
/* *****  
/* TAILOR AND SET DATACLAS FOR ADDING MULTIVOLUME IF USED          */  
/* *****  
  
    FILTLIST &MULTI_VOLUME_DSN INCLUDE  
    ( _____, _____, _____,  
      _____, _____, _____,  
      _____, _____, _____ )  
  
    WHEN ( &DSN EQ &MULTI_VOLUME_DSN )  
  
        DO  
            SET &DATACLAS =ç_____ç  
            EXIT  
        END /* DO */
```

Optional - Data Set List Exceptions

```
/* *****  
/* TAILOR AND SET DATACLAS FOR DATA SET LIST EXCEPTIONS */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SET OF EXCEPTION DATA CLASSES */  
/* BASED ON DATA SET NAME LISTS */  
/* *****  
  
FILTLIST &EXCEPTION_DC_SERVICE_DSN INCLUDE  
( _____ / _____ / _____ /  
  _____ / _____ / _____ /  
  _____ / _____ / _____ )  
  
WHEN ( &DSN EQ &EXCEPTION_DC_SERVICE_DSN )  
  
  DO  
    SET &DATACLAS =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Temp Data Type

```
/* *****  
/* TAILOR AND SET DATACLAS FOR SYSTEM TEMPS IF THEY EXIST          */  
/* *****  
  
    WHEN (&DSTYPE EQ 'TEMP')  
  
        DO  
            SET &DATACLAS = '_____  
            EXIT  
        END /* DO */
```

Optional - Data Set Size

```
/* *****  
/* TAILOR AND SET DATACLAS WHEN YOU WANT TO PROVIDE DIFFERENT      */  
/* SERVICE TO LARGE AND AVERAGE SIZED DATA SETS                    */  
/* *****  
/* YOU CAN ADD THIS LOGIC BLOCK TO ANY SET DATA CLASS SUBTYPES     */  
/* *****  
  
    WHEN ((&SIZE GT ____MB)  
        OR (&MAXSIZE GT ____MB))  
  
        DO  
            SET &DATACLAS = ¢_____¢  
            EXIT  
        END /* END DO */
```

Optional - Temp Short Duration

```
/* *****  
/* TAILOR AND SET DATACLAS FOR TEMP SHORT DURATION SUBTYPES IF THEY */  
/* EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH SUBTYPE OF TEMP SHORT DURATION THAT */  
/* HAS A DIFFERENT DATA CLASS */  
/* *****  
  
    FILTLIST &TEMPPERM_DATA_DSN INCLUDE  
    ( _____, _____, _____,  
      _____, _____, _____,  
      _____, _____, _____ )  
  
    WHEN ( &DSN EQ &TEMPPERM_DATA_DSN )  
  
        DO  
            SET &DATACLAS =ç _____ ç  
            EXIT  
        END /* DO */
```

Optional - TSO Logon and List

```
/* *****  
/* TAILOR AND SET DATACLAS FOR TSO LOGON AND LIST SUBTYPES IF THEY */  
/* HAVE DIFFERENT DATA CLASSES FROM OTHER TSO SUBTYPES */  
/* *****  
  
FILTLIST &TSO_LOGON_DATA_DSN INCLUDE  
(*.ISPF.PROFILE, _____, _____,  
_____, _____, _____)  
  
FILTLIST &TSO_LIST_DATA_DSN INCLUDE  
(**.LIST, **.OUTLIST, **.LINKLIST,  
_____, _____, _____)  
  
WHEN (&DSN EQ &TSO_LOGON_DATA_DSN)  
  
DO  
    SET &DATACLAS =ç_____ç  
    EXIT  
END /* DO */  
  
WHEN (&DSN EQ &TSO_LIST_DATA_DSN)  
  
DO  
    SET &DATACLAS =ç_____ç  
    EXIT  
END /* DO */
```

Optional - TSO

```
/* ***** */
/* TAILOR AND SET DATACLAS FOR TSO SUBTYPES IF THEY EXIST */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH TSO SUBTYPE THAT HAS A DIFFERENT */
/* DATA CLASS AND IS OTHER THAN LOGON OR LIST SUBTYPES */
/* ***** */

FILTLIST &TSO_DATA_HLQ INCLUDE
(¢_____¢,¢_____¢,¢_____¢)

WHEN (&HLQ EQ &TSO_DATA_HLQ)

DO
    SET &DATACLAS =¢_____¢
EXIT
END /* DO */
```


Optional - Test

```
/* *****  
/* TAILOR AND SET DATACLAS FOR TEST SUBTYPES IF THEY EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH TEST SUBTYPE THAT HAS A DIFFERENT */  
/* DATA CLASS */  
/* *****  
  
FILTLIST &TEST_DATA_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )  
  
WHEN ( &DSN EQ &TEST_DATA_DSN )  
  
  DO  
    SET &DATACLAS =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Batch Production

```
/* ***** */
/* TAILOR AND SET DATACLAS FOR BATCH PRODUCTION SUBTYPES IF THEY */
/* EXIST */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH BATCH PRODUCTION SUBTYPE THAT HAS A */
/* DIFFERENT DATA CLASS */
/* ***** */

FILTLIST &BATCH_PROD_DATA_DSN INCLUDE
(
_____/_____/_____/
_____/_____/_____/
_____/_____/_____/
)

WHEN ( &DSN EQ &BATCH_PROD_DATA_DSN)

DO
    SET &DATACLAS =¢_____¢
    EXIT
END /* DO */
```

Optional - Online Production

```
/* *****  
/* TAILOR AND SET DATACLAS FOR ONLINE PRODUCTION SUBTYPES IF THEY */  
/* EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH ONLINE PRODUCTION SUBTYPE THAT HAS A */  
/* DIFFERENT DATA CLASS */  
/* *****  
  
FILTLIST &ONLINE_PROD_DATA_DSN INCLUDE  
( _____ , _____ , _____ ,  
  _____ , _____ , _____ ,  
  _____ , _____ , _____ )  
  
WHEN ( &DSN EQ &ONLINE_PROD_DATA_DSN )  
  
  DO  
    SET &DATACLAS =ç _____ ç  
    EXIT  
  END /* DO */
```

Optional - Other

```
/* *****  
/* TAILOR AND SET DATACLAS FOR OTHER SUBTYPES IF THEY EXIST */  
/* *****  
/* REPEAT THIS BLOCK FOR EACH OTHER SUBTYPE THAT HAS A DIFFERENT */  
/* DATA CLASS */  
/* *****  
  
    FILTLIST &_____DATA_DSN INCLUDE  
    (_____,_____,_____,  
     _____,_____,_____,  
     _____,_____,_____)  
  
    WHEN (&_____DATA_DSN)  
  
        DO  
            SET &DATACLAS =ç_____ç  
            EXIT  
        END /* DO */
```

Optional - Tape Mount Management

```
/* ***** */
/* TAILOR AND SET DATACLAS FOR TAPE MOUNT MANAGEMENT SUBTYPES IF */
/* THEY EXIST */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH TAPE MOUNT MANAGEMENT SUBTYPE THAT HAS */
/* A DIFFERENT DATA CLASS */
/* ***** */

FILTLIST &TAPE_MOUNT_MANAGEMENT_DSN INCLUDE
( _____ , _____ , _____ ,
  _____ , _____ , _____ ,
  _____ , _____ , _____ )

FILTLIST &TAPE_MOUNT_MANAGEMENT_PGM INCLUDE
( $ _____ $ , $ _____ $ , $ _____ $ )

WHEN ( ( &DSN EQ &TAPE_MOUNT_MANAGEMENT_DSN )
OR ( &PGM EQ &TAPE_MOUNT_MANAGEMENT_PGM ) )

DO
  SET &DATACLAS = $ _____ $
  EXIT
END /* DO */
```

Optional - SMS-Managed Tape

```
/* ***** */
/* TAILOR AND SET DATACLAS FOR SMS MANAGED TAPE SUBTYPES IF THEY */
/* EXIST */
/* ***** */
/* REPEAT THIS BLOCK FOR EACH SMS MANAGED TAPE SUBTYPE THAT HAS A */
/* DIFFERENT DATA CLASS */
/* ***** */

FILTLIST &SMS_MANAGED_TAPE_DSN INCLUDE
(
_____/_____/_____/
_____/_____/_____/
_____/_____/_____/
)

WHEN ( &DSN EQ &SMS_MANAGED_TAPE_DSN )

DO
    SET &DATACLAS =ç_____ç
    EXIT
END /* DO */
```

Optional - Otherwise Tape

```
/* *****  
/* OTHERWISE STATEMENT FOR TAPE ALLOCATIONS *  
/* *****  
  
    OTHERWISE  
  
        DO  
            SET &DATACLAS = ¢_____¢  
            EXIT  
        END /* DO */
```


Sample - Data Class ACS Code ...

```
/* ***** */
/* INCLUDE ALL DASD ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* BOTH DASD ALLOCATIONS TO SMS AND TO NON SMS */
/* ***** */

      WHEN ((&UNIT EQ &VALID_DASD_UNIT)
            AND (&ACSENVIR NE &VALID_OPTICAL_ACSENVIR))

            SELECT /* DASD DATA ALLOCATION */

/* ***** */
/* DO NOT MANAGE DATA FOR SPECIAL USERS THAT HAVE BEEN */
/* GIVEN AUTHORITY TO USE DATACLAS=SCNONSMS ON THEIR JCL */
/* THIS ALLOWS SELECTIVE OVERRIDE OF THE DATA CLASS */
/* ACS LOGIC DURING AN EMERGENCY SITUATION SO DATA THAT */
/* WOULD NORMALLY BE MANAGED BY SMS CAN BE ALLOCATED OUTSIDE OF SMS */
/* ***** */

            WHEN (&DATACLAS NE çç)

                    DO
                            SET &DATACLAS = &DATACLAS
                            EXIT
                    END /* DO */

/* ***** */
/* FOR ANY OTHER DATA TYPES NOT IDENTIFIED FOR */
/* SMS MANAGEMENT, PLACE THEM IN NON SMS STORAGE */
/* ***** */

            OTHERWISE /* NONMANAGED DASD DATA */

                    DO
                            SET &DATACLAS = çç
                            EXIT
                    END /* DO */

            END /* DASD ALLOCATION SELECT */
```

Sample - Data Class ACS Code ...

```
/* ***** */
/* INCLUDE ALL TAPE ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* BOTH TAPE ALLOCATIONS TO SMS AND TO NON SMS */
/* PRIOR TO SMS MANAGED TAPE OR TAPE MOUNT MANAGEMENT */
/* ALL TAPE ALLOCATION WOULD BE TO NON SMS */
/* ***** */

/* ***** */
/* START TAPE ALLOCATION LOGIC BLOCK */
/* ***** */
/* */
/* ***** */
/* END TAPE ALLOCATION LOGIC BLOCK */
/* ***** */
```

Sample - Data Class ACS Code ...

```

/*****
/* INCLUDE ALL OPTICAL ALLOCATIONS IN THIS BLOCK, INCLUDE */
/* LOGIC FROM PRODUCTS LIKE IAFIC OR IMAGE PLUS HERE */
*****/

/*****
/* START OPTICAL ALLOCATION LOGIC BLOCK */
*****/
/*
*****/
/* END OPTICAL ALLOCATION LOGIC BLOCK */
*****/

        OTHERWISE /* NONMANAGED DEVICES */

                DO
                        SET &DATACLAS =çç
                        EXIT
                END /* DO */

        END /* SELECT */

END /* PROC */
```

Phased Implementation

Phased Implementation Design

Topics

- Design considerations
- Phased implementation question set
- Definition of phased implementation steps
- Update of data classification

Phased Implementation Benefits

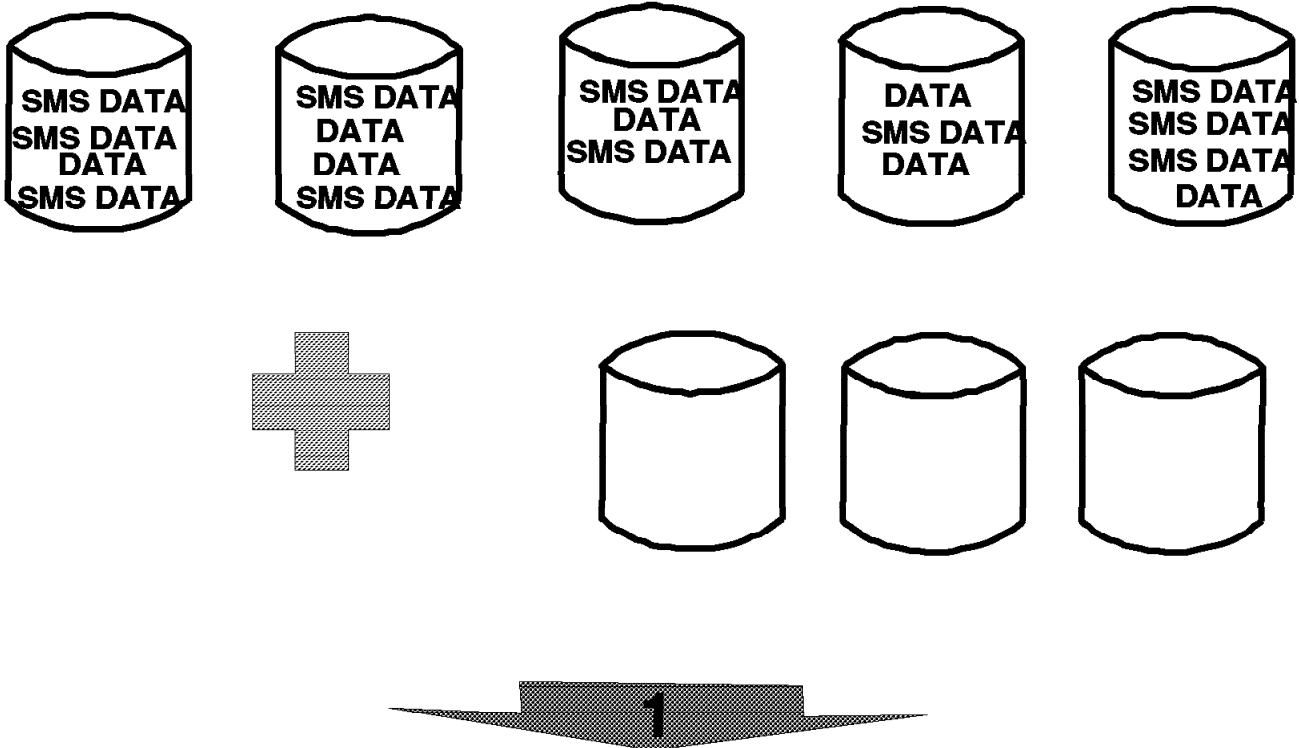
- Implement phases according to installation's needs
 - Lowest impact
 - Greatest benefit
- Learn from early implementation experience
- Convert large amounts of data in small steps
- Improve testing through structured process
- Tailor conversion of online data to best possible time

Phased Implementation Considerations

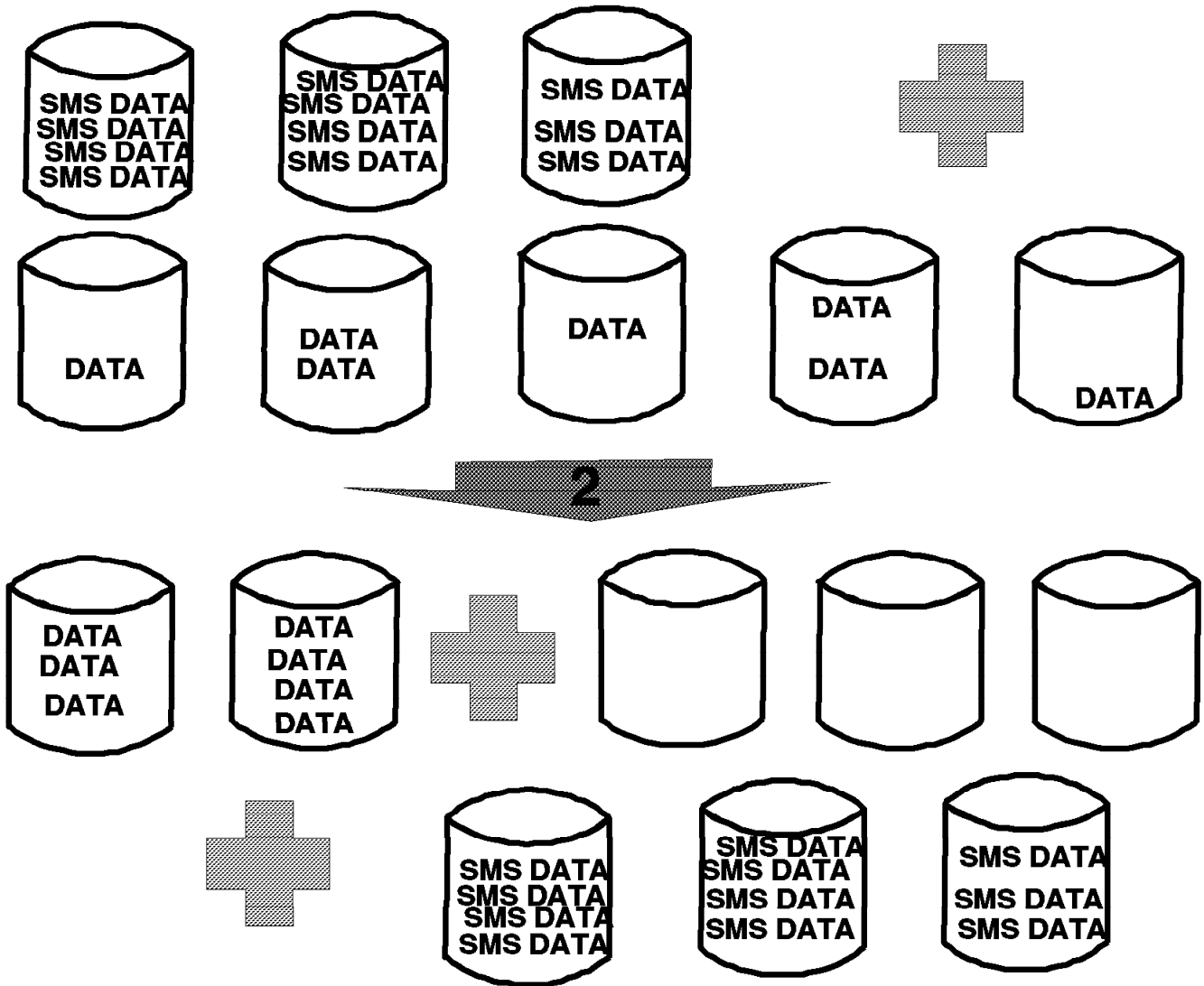
- Determine windows when data can be converted
- Amount of seed DASD affects the speed of conversion
- Allow time for contingencies during data conversion windows
- DFDSS dump and then copy is faster than dump and restore
- Logical copy and dump provide flexibility
- Not possible to convert data during prime shift if there is no seed DASD

Phased Implementation Alternatives

**DUMP, COPY TO SEED DASD,
CONSOLIDATE, FREE DASD**

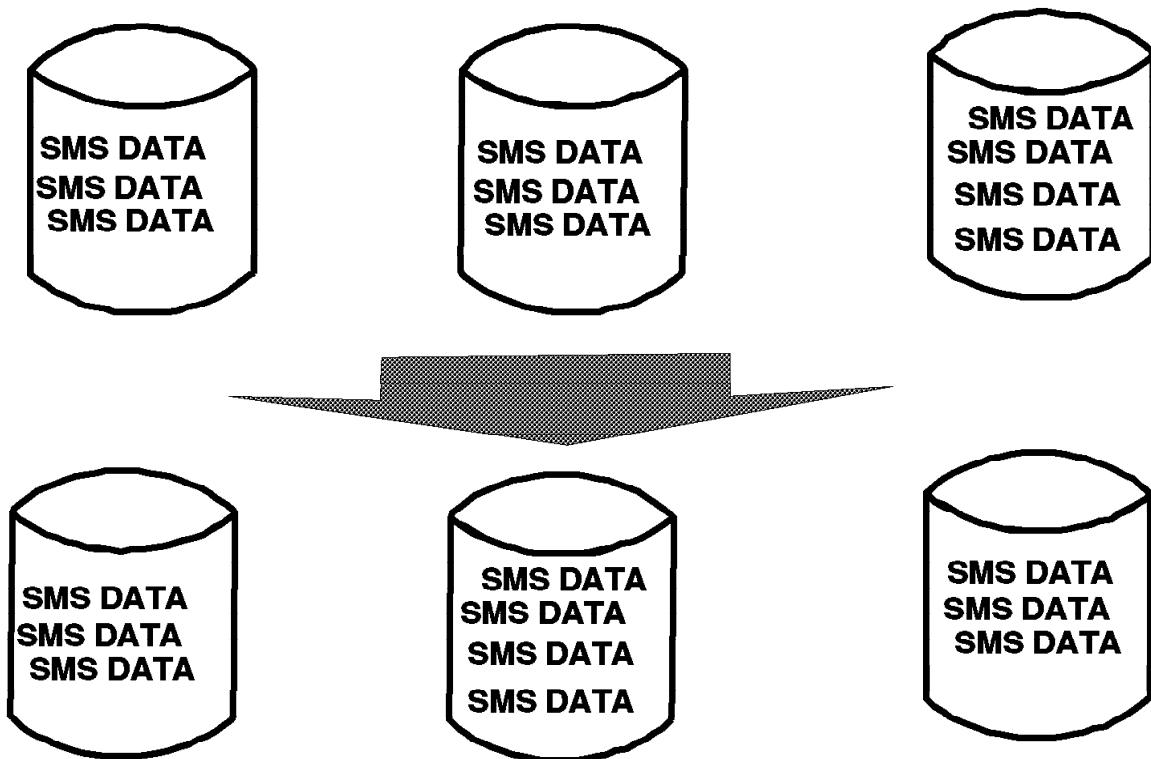


Phased Implementation Alternatives ...



Phased Implementation Alternatives ...

DUMP, INITIALIZE, RESTORE



Phased Implementation Recommendations

- Implement one data type at a time
- Start with lowest impact data or functions
- Pilot each data type or function
 - Pilot users from within systems support
 - Pilot users from willing users or applications
- Consider continuing order based on lowest impact
- Move unallocated data during prime time
- Back up data before or during data conversion
- Avoid CONVERTV when possible
- Avoid complex functions and online data in initial implementation
- Consider how to free DASD space early in data conversion
 - GDGs
 - Data with early deletion or migration service

Phased Implementation Design Questions

1. Do you want to convert all data within the data type at one time?
2. How do you want to order your phases?
 - By lowest impact
 - By highest benefit
3. How much seed DASD is available (80% of total capacity)?
4. How many concurrent data conversion jobs do you want to run?
 - One
 - Two
 - Four
 - Six
5. How long is the weeknight window for data movement?
 - Short - 2 hours
 - Single shift - 5 hours
 - Double shift - 10 hours
6. How long is the weekend window for data movement?
 - Double shift - 10 hours
 - All weekend - 25 hours

Between Design Questions 2 and 3

- Tailor space analysis job
- Include analysis for each group of data that is to be converted at the same time
- Record the gigabytes for each type of data
- Assume that all online data must be converted in a specific window
- Assume that one-half of batch data can be converted in prime time
- Assume that three-quarters of TSO and test data can be converted in prime time

Phased Implementation Design Table

Data Type						
Classification boxes						
Order By						
GBs						
GB to move						
GB in prime time						
GB during window						
Seed DASD GB						
Window Needed						
Hours at 500 MB/hour						
Number jobs						
Total hours						
Max seed hours						
Weeknight Window						
2 hours						
5 hours						
10 hours						
Weekend Window						
10 hours						
25 hours						
Number Phases						
Weeknights						
Weekends						
Phase IDs						

Sample Phased Implementation Design

Data Type	WK	TS	T	B P		
Classification boxes	1	2-4	5-7	8-12		
Order By						
Impact	L	M	M	H		
Impact	1	2	3	4		
GBs						
GB to move	0	14	24	44		
GB in prime time	0	11	19	22		
GB during window	0	3	5	22		
Seed DASD GB	13	13	13	13		
Window Needed						
Hours at 500 MB/hour	0	6	10	44		
Number jobs	0	4	4	4		
Total hours	0	1.5	2.5	11		
Max seed hours	0	6.5	6.5	6.5		
Weeknight Window						
2 hours	X			X		
5 hours		X	X			
10 hours						
Weekend Window						
10 hours	X			X		
25 hours		X	X			
Number Phases						
Weeknights	0	1	1	0		
Weekends	0	0	0	2		
Phase IDs	1-2	3-5	6-7	8-10		

Implementation Phases

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.
- 9.
- 10.
- 11.
- 12.
- 13.
- 14.
- 15.
- 16.
- 17.
- 18.
- 19.

Sample Implementation Phases

1. Work - pilot of system userids
2. Work - all data
3. TSO - pilot of storage administrator and systems programming userids
4. TSO - pilot of system userids
5. TSO - all data
6. Test - pilot of selected application HLQs
7. Test - all data
8. Batch production - pilot of selected application HLQs
9. Batch production - first half of applications by HLQs
10. Batch production - second half of applications

Sample - Work Data

Phase 2 WK1 DSTYPE=TEMP
Phase 1 Work PILOT=userid list

SC=SCFAST
SG=SGWORK
MC=N/A
DC=optionally selected on JCL

Sample - TSO Data

Phase 5		HLQ=%%T%%
Phase 3	TSO	PILOT1=userid list
Phase 4		PILOT2=userid list

	OTHERWISE	DSN=list	DSN=list
TS2	TS3	TS4	
Normal Data	LIST, LOG, ISPF*TEMP.CNTL type data	Used at LOGON	
SC=SCNORMAL SG=SGTSO MC=MCEARLY DC=optionally selected on JCL	SC=SCNORMAL SG=SGTSO MC=MCINTERM DC=optionally selected on JCL	SC=SCNORMAL SG=SGTSO MC=MCNOMIG DC=optionally selected on JCL	

Sample - Test Data

Phase 7
Phase 6

Test

HLQ=application list
2LQ=T*
PILOT=HLQ list

T5	T6	T7
Normal	One Day	GDGs
OTHERWISE	2LQ=T%%TEMP*	DSTYPE=GDS
SC=SCNORMAL SG=SGISO MC=MCSTAND DC=optionally selected on JCL	SC=SCNORMAL SG=SGISO MC=MCINTERM DC=optionally selected on JCL	SC=SCNORMAL SG=SGISO MC=MCNOBACK DC=optionally selected on JCL

Sample - Exceptions

Exceptions

DSN=list	DSN=list	JCL STORCLAS=SCNOSMS
EX12 Normal Performance Hand Placed	EX13 Critical Performance Hand Placed	EX14 Non SMS
SC=SCSPECN SG=as usual MC=as usual DC=optionally selected on JCL	SC=SCSPECF SG=as usual MC=as usual DC=optionally selected on JCL	SC=☐☐

Testing

Testing

Topics

- Testing considerations
- Testing questions
- Definition of testing steps

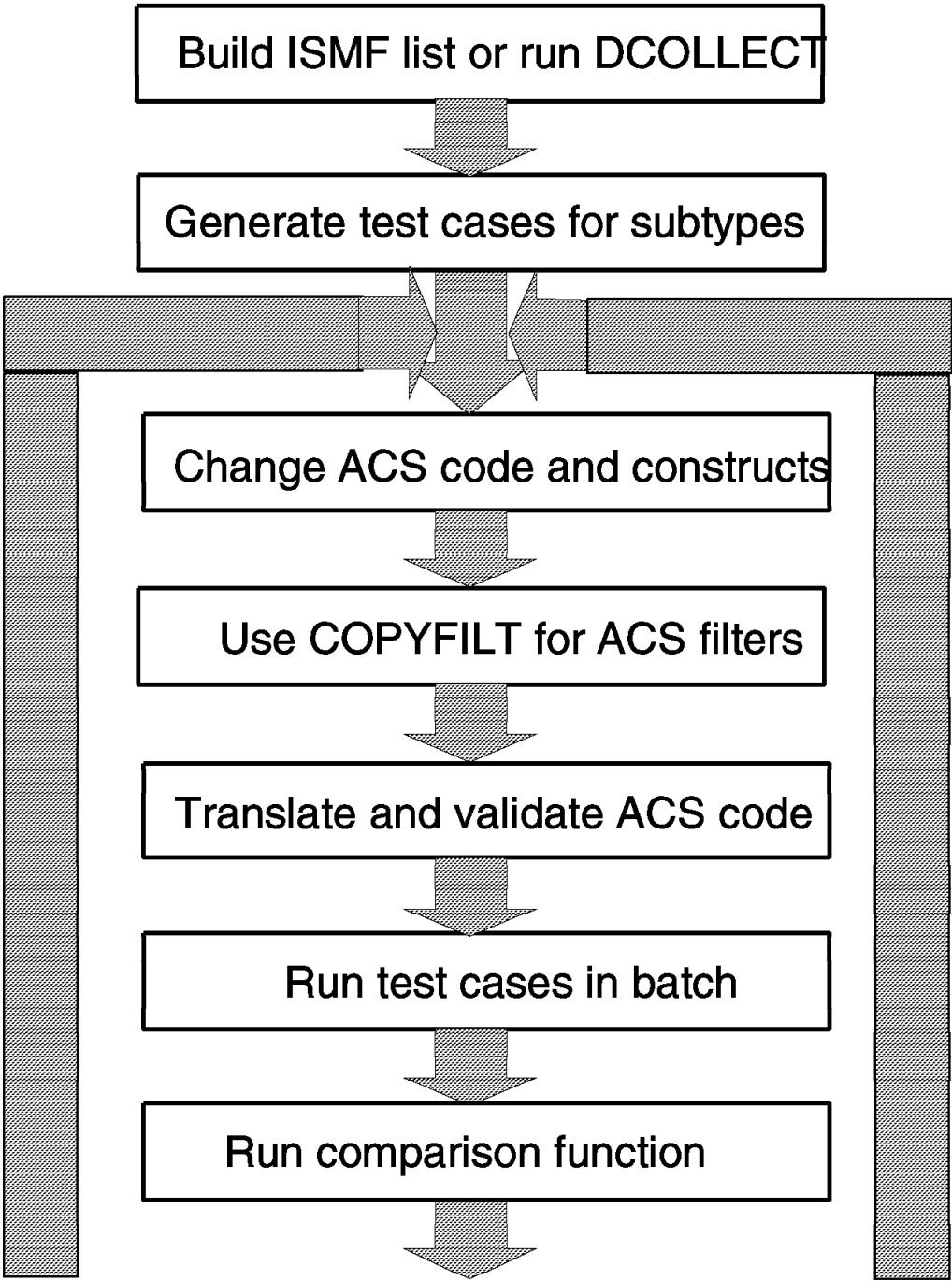
Testing Benefits

- Management class service has powerful implications
 - Deletion
 - Backup
- Correct ACS class and group assignment can be verified in advance
- Actual data set assignments can be verified in advance
- Confidence that ACS routines work correctly with your data
- Assurance that new changes work and old functions continue to work

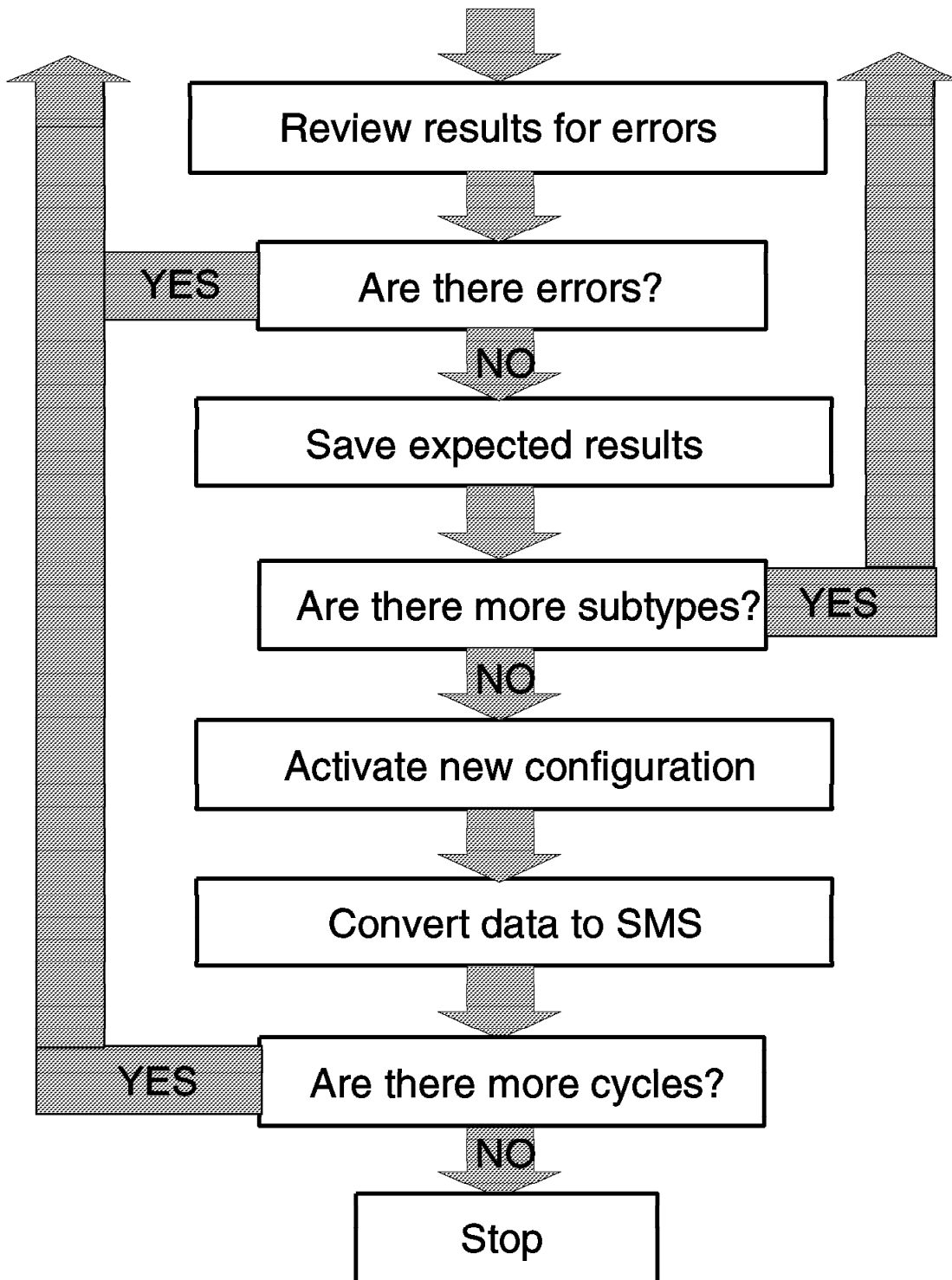
Testing Considerations

- Level of automation to use during testing
 - Use NaviQuest and ISMF
 - Use ISMF only
- Amount of data to use for testing
 - All data sets
 - Sampling of data sets
- Method of building test cases
 - ISMF data set lists
 - DCOLLECT D records
 - Storage class exit and SMF
 - VMA
 - Manual entry
- Extent of regression testing
- Must test nonmanaged data types
- Must test exceptions
- Use of storage class exit from CBIPO materials to display ACS variables for debugging

Recommended Testing Process



Recommended Testing Process ...



Testing Recommendations

- Use both NaviQuest and ISMF
- Place test cases for all subtypes into initial test bed
- Run testing as ISMF in batch
- Keep all FILTLISTs in a separate PDS member
- Implement configuration change management

Testing Questions

1. How do you want to collect data to build the test cases?

- ISMF data set lists
- DCOLLECT D records
- Storage class exit and SMF
- VMA
- Manual entry

2. How much data do you want to use for testing?

- All data sets
- Sampling of data sets

3. Are nonmanaged data types well defined?

Testing Table

Data Type	Non-SMS			
Classification box				
Subtype prefix				
Phase number				
Creation				
ISMF				
DCOLLECT				
SC exit				
VMA				
Manual				
Amount				
All data				
Sample				
Selection				
Filter				
ACS Use				
GROUP				
USER				
JOB				
ACCT JOB				
PGM				
Expected Results				
STORCLAS				
STORGRP				
MGMTCLAS				
DATACLAS				

Sample Testing Answers

Data Type	Non-SMS	WK	TS	TS
Classification box		1	2	3
Subtype prefix	NEVR	WK1	TS2	TS3
Phase number		1,2	3-5	3-5
Creation				
ISMF	X	X	X	X
DCOLLECT				
SC exit				
VMA				
Manual				
Amount				
All data				
Sample	X	X	X	X
Selection				
Filter	HLQ+2LQ	TEMP	HLQ	DSN
ACS Use				
GROUP				
USER		X	X	X
JOB				
ACCT JOB				
PGM				
Expected Results				
STORCLAS		SCFAST	SCNORMAL	SCNORMAL
STORGRP		SGWORK	SGTSO	SGTSO
MGMTCLAS			MCEARLY	MCINTERM
DATACLAS				

Sample Testing Answers ...

Data Type	TS	T	T	T
Classification box	4	5	6	7
Subtype prefix	TS4	T5	T6	T7
Phase number	3-5	6,7	6,7	6,7
Creation				
ISMF	X	X	X	X
DCOLLECT				
SC exit				
VMA				
Manual				
Amount				
All data				
Sample	X	X	X	X
Selection				
Filter	DSN	HLQ+2LQ	HLQ+2LQ	HLQ+2LQ +GDS
ACS Use				
GROUP				
USER	X	X	X	X
JOB				
ACCT JOB				
PGM				
Expected Results				
STORCLAS	SCNORMAL	SCNORMAL	SCNORMAL	SCNORMAL
STORGRP	SGTSO	SGTSO	SGTSO	SGTSO
MGMTCLAS	MCNOMIG	MCSTAND	MCINTERM	MCNOBACK
DATACLAS				

Sample Testing Answers ...

Data Type	BP	BP	BP	BP
Classification box	8	9	10	11
Subtype prefix	BP8	BP9	BP10-	BP11
Phase number	8-10	8-10	8-10	8-10
Creation				
ISMF	X	X	X	X
DCOLLECT				
SC exit				
VMA				
Manual				
Amount				
All data	X	X	X	X
Sample				
Selection				
Filter	HLQ+2LQ	HLQ+2LQ	DSN	HLQ+2LQ +GDS
ACS Use				
GROUP				
USER	X	X	X	X
JOB				
ACCT JOB				
PGM				
Expected Results				
STORCLAS	SCNORMAL	SCNORMAL	SCFAST	SCNORMAL
STORGRP	SGPRIME	SGPRIME	SGPRIME	SGPRIME
MGMTCLAS	MCEARLY	MCLATE	MCNOMIG	MCEARLY
DATACLAS				

Sample Testing Answers ...

Data Type	EX	EX	EX	
Classification box	12	13	14	
Subtype prefix	EX12	EX13	EX14	
Phase number				
Creation				
ISMF				
DCOLLECT				
SC exit				
VMA				
Manual	X	X	X	
Amount				
All data				
Sample	X	X	X	
Selection				
Filter	DSN	DSN	DSN	
ACS Use				
GROUP				
USER	X	X	X	
JOB				
ACCT JOB				
PGM				
Expected Results				
STORCLAS	SCSPECN	SCSPECF	SCNOSMS	
STORGRP	any	any		
MGMTCLAS	any	any		
DATACLAS				

Testing Cycles

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.
- 9.
- 10.
- 11.
- 12.
- 13.
- 14.
- 15.
- 16.
- 17.
- 18.
- 19.

Sample Testing Cycles

1. Nonmanaged

- NEVR - SYS%.** and *.P*.IMS*.**

2. Work

- WK1 - DSTYPE=TEMP

3. TSO

- TS2 - %%%T%%%.**
- TS3 - %%%T%%%.**.LIST (and LOG or ISPFTEMP.CNTL)
- TS4 - %%%T%%%.ISPF.PROFILE (and others)

4. Test

- T5 - *.T*.**
- T6 - *.T%%TEMP*.**
- T7 - *.T*.** and DSTYPE=GDS

5. Batch production

- BP8 - *.P%%MF*.**
- BP9 - *.P*.**
- BP10 - *.P*.CRITDATA
- BP11 - *.P*.** and DSTYPE=GDS

6. Exceptions

- EX12 - STORCLAS=SCSPECN
- EX13 - STORCLAS=SCSPECF
- EX14 - STORCLAS=SCNOSMS

Sample Ordered Testing Cycles

1. Nonmanaged

- a. NEVR - SYS%.** and *.P*.IMS*.**

2. Work

- a. WK1 - DSTYPE=TEMP

3. TSO

- a. TS3 - %%%T%%%.** .LIST (and LOG or ISPFTEMP.CNTL)
- b. TS4 - %%%T%%%.ISPF.PROFILE (and others)
- c. TS2 - %%%T%%%.**

4. Test

- a. T7 - *.T*.** and DSTYPE=GDS
- b. T6 - *.T%%TEMP*.**
- c. T5 - *.T*.**

5. Batch production

- a. BP10 - *.P*.CRITDATA
- b. BP11 - *.P*.** and DSTYPE=GDS
- c. BP8 - *.P%%MF*.**
- d. BP9 - *.P*.**

6. Exceptions

- a. EX12 - STORCLAS=SCSPECN
- b. EX13 - STORCLAS=SCSPECF
- c. EX14 - STORCLAS=SCNOSMS

Sample Test Case Data Sets

- NEVR - SYS%.** and *.P*.IMS*.**

NEVR - SYS1.PROCLIB

NEVR - APPL1.PABC.IMS.DATABASE

- WK1 - DSTYPE=TEMP

WK1 - SYS95243.T085927.RA000.TSOTDEF.R0104192

- TS3 - %%%T%%%.** .LIST

TS3 - TSOTDEF.MYTSO.LIST

- TS4 - %%%T%%%.ISPF.PROFILE

TS4 - TSOTDEF.ISPF.PROFILE

- TS2 - %%%T%%%.**

TS2 - TSOTDEF.MY.TSODATA

- T7 - *.T*.** and DSTYPE=GDS

T7 - APPL1.TABC.TESTGDG.G0001V00

- T6 - *.T%%TEMP*.**

T6 - APPL1.TABTEMP.TESTDATA

- T5 - *.T*.**

T5 - APPL1.TABC.TESTDATA

Sample Test Case Data Sets ...

- BP10 - *.P*.CRITDATA

BP10 - APPL1.PABC.CRITDATA

- BP11 - *.P*.** and DSTYPE=GDS

BP11 - APPL1.PABC.PRODGDG.G0001V00

- BP8 - *.P%%MF*.**

BP8 - APPL1.PABMF.PRODDATA

- BP9 - *.P*.**

BP9 - APPL1.PABC.PRODMAST

- EX12 - STORCLAS=SCSPECN

EX12 - APPL1.TABC.HANDPLAC and STORCLAS=SCSPECN

- EX13 - STORCLAS=SCSPECF

EX13 - APPL1.PABC.HANDPLAC and STORCLAS=SCSPECF

- EX14 - STORCLAS=SCNOSMS

EX14 - APPL1.PABC.NOSMS and STORCLAS=SCNOSMS

Sample Non-SMS Data

Nonmanaged

HLQ=SYS%
NEVR
System

HLQ=application list
2LQ=P*
NEVR
Online
Production
Database

Sample Non-SMS Testing (Cycle 1)

- Change ACS code -> Run test cases
 - All data sets assigned STORCLAS=""
- Run comparison function
 - No comparison for the first testing cycle. This cycle creates the baseline for future comparisons
- Review errors -> Save results - all prefixes:

STORCLAS=""
STORGRP = ""
MGMTCLAS=""
DATACLAS=""
Any other values are errors
- Are there more subtypes?
 - No, there are no more subtypes for the baseline.
- Activate new configuration -> Convert data to SMS
 - You do not activate or convert for the baseline cycle.
- Are there more cycles?
 - Yes, continue with cycle 2.

Sample - Work Data

Phase 2 WK1 DSTYPE=TEMP
Phase 1 Work PILOT=userid list

SC=SCFAST
SG=SGWORK
MC=N/A
DC=optionally selected on JCL

Sample Work Testing (Cycle 2)

- Change ACS code -> Run test cases
 - Assign WK1 to SMS
 - All other data sets assigned STORCLAS=""
- Run comparison function
 - All WK1 test cases should be flagged as exceptions.
- Review errors -> Save results - WK1 prefix
 - STORCLAS=SCFAST
 - STORGRP =SGWORK
 - MGMTCLAS=""
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - No, there are no more subtypes for work data.
- Activate new configuration -> Convert data to SMS
 - Activate new SCDS and work data will be SMS-managed.
- Are there more cycles?
 - Yes, continue with cycle 3.

Sample - TSO Data

Phase 5 HLQ=%%T%%
Phase 3 TSO PILOT1=userid list
Phase 4 PILOT2=userid list

	OTHERWISE	DSN=list	DSN=list
TS2	TS3	TS4	
Normal Data	LIST, LOG, ISPF*TEMP.CNTL type data	Used at LOGON	
SC=SCNORMAL SG=SGTSO MC=MCEARLY DC=optionally selected on JCL	SC=SCNORMAL SG=SGTSO MC=MCINTERM DC=optionally selected on JCL	SC=SCNORMAL SG=SGTSO MC=MCNOMIG DC=optionally selected on JCL	

Sample - TSO Testing (Cycle 3)

- Change ACS code -> Run test cases - CYCLE 3A
 - Assign TS3 and WK1 to SMS
 - All other data sets assigned `STORCLAS=""`

- Run comparison function
 - All TS3 test cases should be flagged as exceptions

- Review errors -> Save results - TS3 prefix

```
STORCLAS=SCNORMAL
STORGRP =SGTSO
MGMTCLAS=MCINTERM
DATACLAS=""
Any other values are errors
```

- Are there more subtypes?
 - Yes, there are two more subtypes of TSO data.
- Change ACS code -> Run test cases - CYCLE 3B
 - Assign TS3-4 and WK1 to SMS
 - All other data sets assigned `STORCLAS=""`

- Run comparison function
 - All TS4 test cases should be flagged as exceptions

Sample - TSO Testing (Cycle 3) ...

- Review errors -> Save results - TS4 prefix
 - STORCLAS=SCNORMAL
 - STORGRP =SGTSO
 - MGMTCLAS=MCNOMIG
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - Yes, there is one more subtype of TSO data.
- Change ACS code -> Run test cases - CYCLE 3C
 - Assign TS2-4 and WK1 to SMS
 - All other data sets assigned STORCLAS=""
- Run comparison function
 - All TS2 test cases should be flagged as exceptions
- Review errors -> Save results - TS2 prefix
 - STORCLAS=SCNORMAL
 - STORGRP =SGTSO
 - MGMTCLAS=MCEARLY
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - No, there are no more subtypes for TSO data.
- Activate new configuration -> Convert data to SMS
 - Activate new SCDS and work and TSO data will be SMS-managed.
- Are there more cycles?
 - Yes, continue with cycle 4.

Sample - Test Data

Phase 7
Phase 6

Test

HLQ=application list
2LQ=T*
PILOT=HLQ list

T5	T6	T7
Normal	One Day	GDGs
OTHERWISE	2LQ=T%%TEMP*	DSTYPE=GDS
SC=SCNORMAL SG=SGISO MC=MCSTAND DC=optionally selected on JCL	SC=SCNORMAL SG=SGISO MC=MCINTERM DC=optionally selected on JCL	SC=SCNORMAL SG=SGISO MC=MCNOBACK DC=optionally selected on JCL

Sample - Test Testing (Cycle 4)

- Change ACS code -> Run test cases - CYCLE 4A
 - Assign T7, TS2-4 and WK1 to SMS
 - All other data sets assigned STORCLAS=""

- Run comparison function

- All T7 test cases should be flagged as exceptions

- Review errors -> Save results - T7 prefix

```
STORCLAS=SCNORMAL
STORGRP =SGTSO
MGMTCLAS=MCNOBACK
DATACLAS=""
Any other values are errors
```

- Are there more subtypes?

- Yes, there are two more subtypes of test data.

- Change ACS code -> Run test cases - CYCLE 4B

- Assign T6-7, TS2-4, and WK1 to SMS
- All other data sets assigned STORCLAS=""

- Run comparison function

- All T6 test cases should be flagged as exceptions

Sample - Test Testing (Cycle 4) ...

- Review errors -> Save results - T6 prefix
 - STORCLAS=SCNORMAL
 - STORGRP =SGTSO
 - MGMTCLAS=MCINTERM
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - Yes, there is one more subtype of test data.
- Change ACS code -> Run test cases - CYCLE 4C
 - Assign T5-7, TS2-4, and WK1 to SMS
 - All other data sets assigned STORCLAS=""
- Run comparison function
 - All T5 test cases should be flagged as exceptions
- Review errors -> Save results - T5 prefix
 - STORCLAS=SCNORMAL
 - STORGRP =SGTSO
 - MGMTCLAS=MCSTAND
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - No, there are no more subtypes for test data.
- Activate new configuration -> Convert data to SMS
 - Activate new SCDS and work, TSO, and test data will be SMS-managed.
- Are there more cycles?
 - Yes, continue with cycle 5.

Sample - Batch Production Data

Phase 10 Batch HLQ=application list
 Production 2LQ=P*
Phase 8 PILOT=HLQ list
Phase 9 PHASE1=HLQ list

2LQ=P%%MF*	OTHERWISE	DSN=list
BP8 Working Files	BP9 Master Files	BP10 Critical Performance Files
SC=SCNORMAL SG=SGPRIME MC=MCEARLY DC=optionally selected on JCL	SC=SCNORMAL SG=SGPRIME MC=MCLATE DC=optionally selected on JCL	SC=SCFAST SG=SGPRIME MC=MCNOMIG DC=optionally selected on JCL

BP11
 DSTYPE=GDS
 GDGs

SC=SCNORMAL
SG=SGPRIME
MC=MCEARLY
DC=optionally selected on JCL

Sample - Batch Production Testing (Cycle 5)

- Change ACS code -> Run test cases - CYCLE 5A
 - Assign BP10, T5-7, TS2-4, and WK1 to SMS
 - All other data sets assigned STORCLAS=""
- Run comparison function
 - All BP10 test cases should be flagged as exceptions
- Review errors -> Save results - BP10 prefix
 - STORCLAS=SCFAST
 - STORGRP =SGPRIME
 - MGMTCLAS=MCNOMIG
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - Yes, there are three more subtypes of batch production data.
- Change ACS code -> Run test cases - CYCLE 5B
 - Assign BP10-11, T5-7, TS2-4, and WK1 to SMS
 - All other data sets assigned STORCLAS=""
- Run comparison function
 - All BP11 test cases should be flagged as exceptions

Sample - Batch Production Testing (Cycle 5) ...

- Review errors -> Save results - BP11 prefix
 - STORCLAS=SCNORMAL
 - STORGRP =SGPRIME
 - MGMTCLAS=MCEARLY
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - Yes, there are two more subtypes of batch production data.
- Change ACS code -> Run test cases - CYCLE 5C
 - Assign BP8,10-11, T5-7, TS2-4, and WK1 to SMS
 - All other data sets assigned STORCLAS=""
- Run comparison function
 - All BP8 test cases should be flagged as exceptions
- Review errors -> Save results - BP8 prefix
 - STORCLAS=SCNORMAL
 - STORGRP =SGPRIME
 - MGMTCLAS=MCEARLY
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - Yes, there is one more subtype of batch production data.

Sample - Batch Production Testing (Cycle 5) ...

- Change ACS code -> Run test cases - CYCLE 5D
 - Assign BP8-11, T5-7, TS2-4, and WK1 to SMS
 - All other data sets assigned STORCLAS=""

- Run comparison function

- All BP9 test cases should be flagged as exceptions

- Review errors -> Save results - BP9 prefix

```
STORCLAS=SCNORMAL
STORGRP =SGPRIME
MGMTCLAS=MCLATE
DATACLAS=""
Any other values are errors
```

- Are there more subtypes?

- No, there are no more subtypes for batch production data.

- Activate new configuration -> Convert data to SMS

- Activate new SCDS and work, TSO, test, and batch production data will be SMS-managed.

- Are there more cycles?

- Yes, continue with cycle 6.

Sample - Exceptions

Exceptions

DSN=list	DSN=list	JCL
EX12 Normal Performance Hand Placed	EX13 Critical Performance Hand Placed	STORCLAS=SCNOSMS
SC=SCSPECN SG=as usual MC=as usual DC=optionally selected on JCL	SC=SCSPECF SG=as usual MC=as usual DC=optionally selected on JCL	SC=☐☐

Sample - Exceptions Testing (Cycle 6)

- Change ACS code -> Run test cases - CYCLE 6A
 - Assign EX12, BP8-11, T5-7, TS2-4, and WK1 to SMS
 - All other data sets assigned STORCLAS=""

- Run comparison function
 - All EX12 test cases should be flagged as exceptions

- Review errors -> Save results - EX12 prefix

STORCLAS=SCSPECN

STORGRP =any valid storage group name

MGMTCLAS=any valid management class name

DATACLAS=""

Any other values are errors

- Are there more subtypes?
 - Yes, there are two more subtypes of exception data.
- Change ACS code -> Run test cases - CYCLE 6B
 - Assign EX12-13, BP8-11, T5-7, TS2-4, and WK1 to SMS
 - All other data sets assigned STORCLAS=""
- Run comparison function
 - All EX13 test cases should be flagged as exceptions

Sample - Exceptions Testing (Cycle 6) ...

- Review errors -> Save results - EX13 prefix
 - STORCLAS=SCSPECF
 - STORGRP =any valid storage group name
 - MGMTCLAS=any valid management class name
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - Yes, there is one more subtype of exception data.
- Change ACS code -> Run test cases - CYCLE 6C
 - Assign EX12-14, BP8-11, T5-7, TS2-4, and WK1 to SMS
 - All other data sets assigned STORCLAS=""
- Run comparison function - CYCLE 6c
 - All EX14 test cases should be flagged as exceptions
- Review errors -> Save results - EX14 prefix
 - STORCLAS=SCNOSMS
 - STORGRP = ""
 - MGMTCLAS=""
 - DATACLAS=""
 - Any other values are errors
- Are there more subtypes?
 - No, there are no more subtypes for exception data.
- Activate new configuration -> Convert data to SMS
 - Activate new SCDS and all planned data will be SMS-managed.
- Are there more cycles?
 - No, you have completed the DF SMSFIT testing

Staged Data Conversion

Staged Data Conversion

Topics

- Staged data conversion considerations
- Staged data conversion questions
- Definition of staged data conversion steps

Staged Data Conversion Benefits

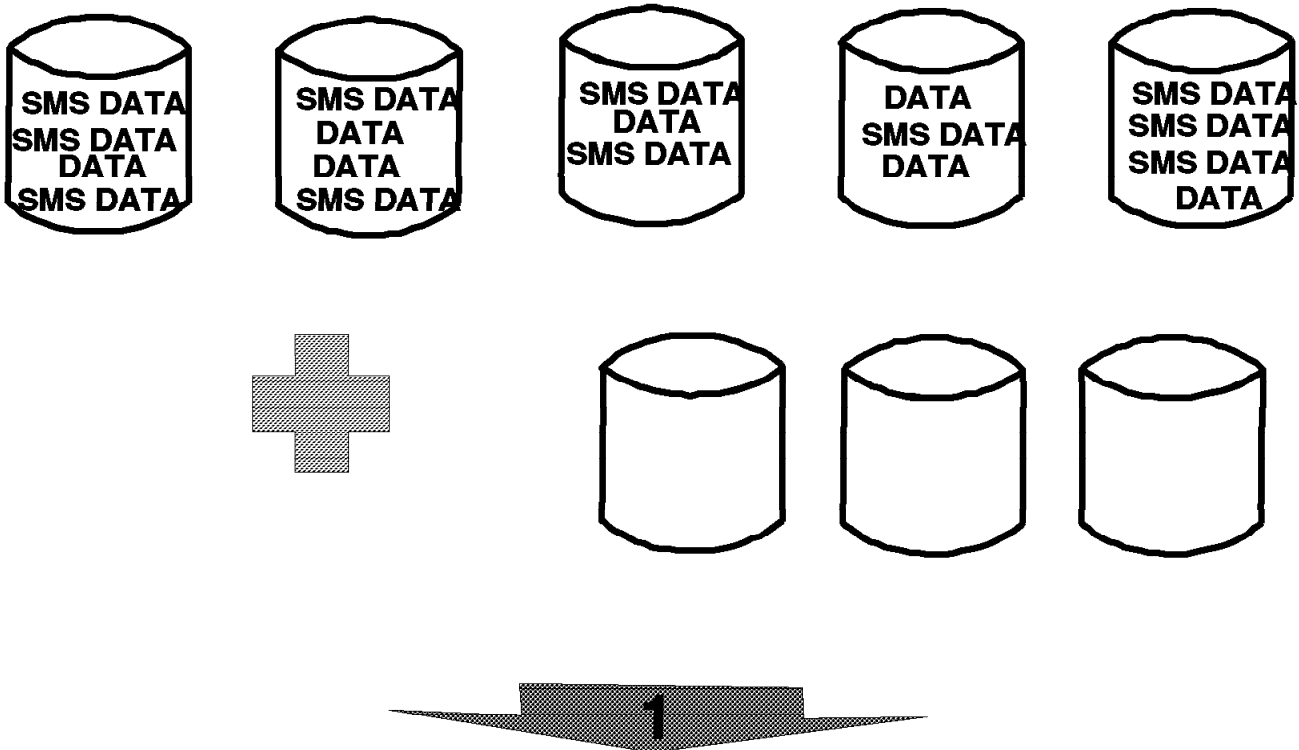
- Learn from early data conversion experience
- Convert large amounts of data in small steps
- Tailor conversion of online data to best possible time

Staged Data Conversion Considerations

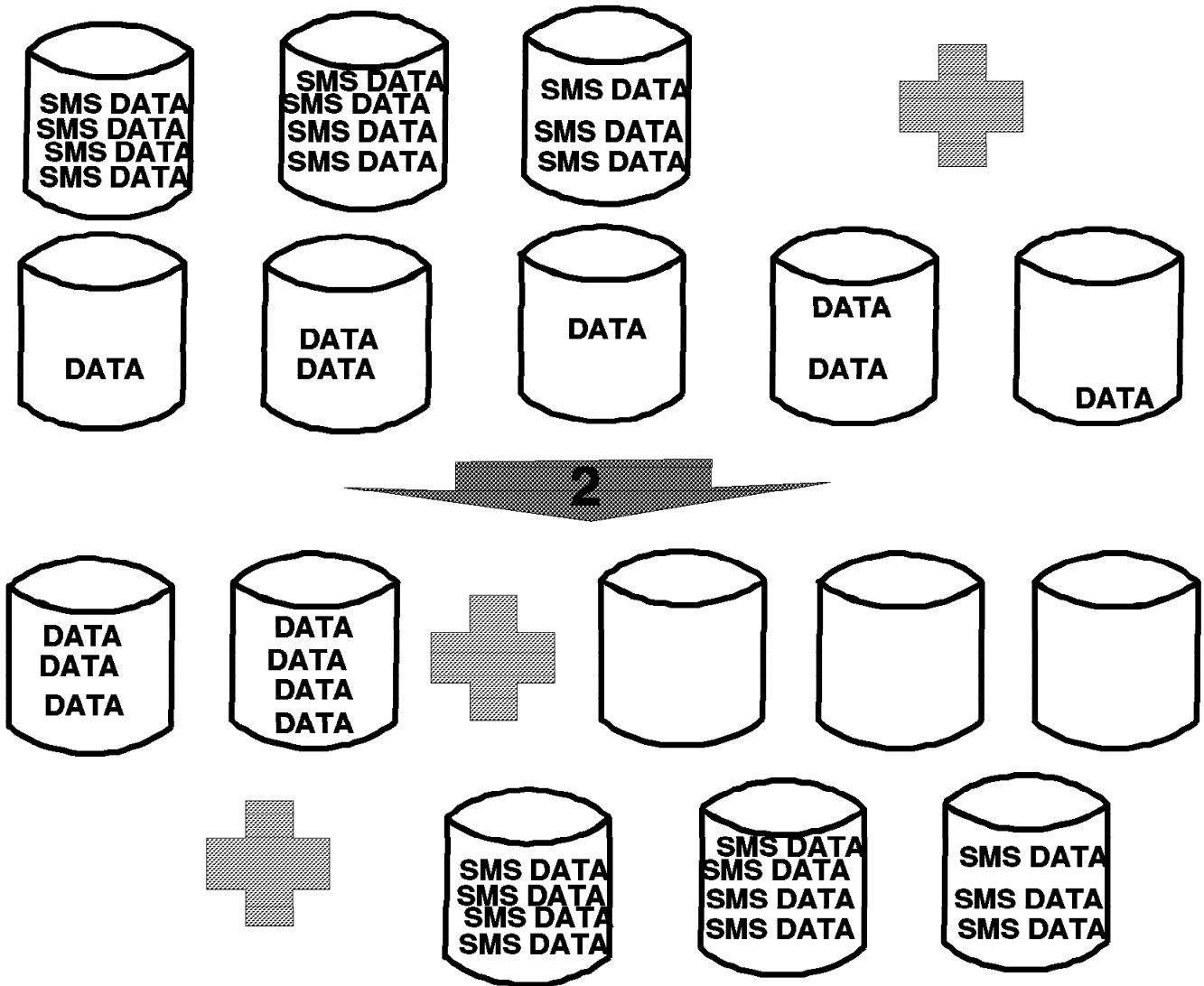
- Determine windows when data can be converted
- Amount of seed DASD affects the speed of conversion
- Allow time for contingencies during data conversion windows
- Activities during prime shift
 - Initialize seed DASD volumes for SMS
 - Activate the new ACS routines
 - Move data to SMS-managed seed DASD volumes by data set name filtering or full volume
- Activities during weeknight or weekend windows
 - Move the remaining data that was active during the prime shift to SMS-managed seed DASD volumes
 - Dump volumes by data set name filtering or full volume
 - Initialize dumped volumes for SMS
 - Activate the new ACS routines
 - Restore dumped data by data set name filtering or full volume to SMS-management

Staged Data Conversion Alternatives

**DUMP, COPY TO SEED DASD,
CONSOLIDATE, FREE DASD**

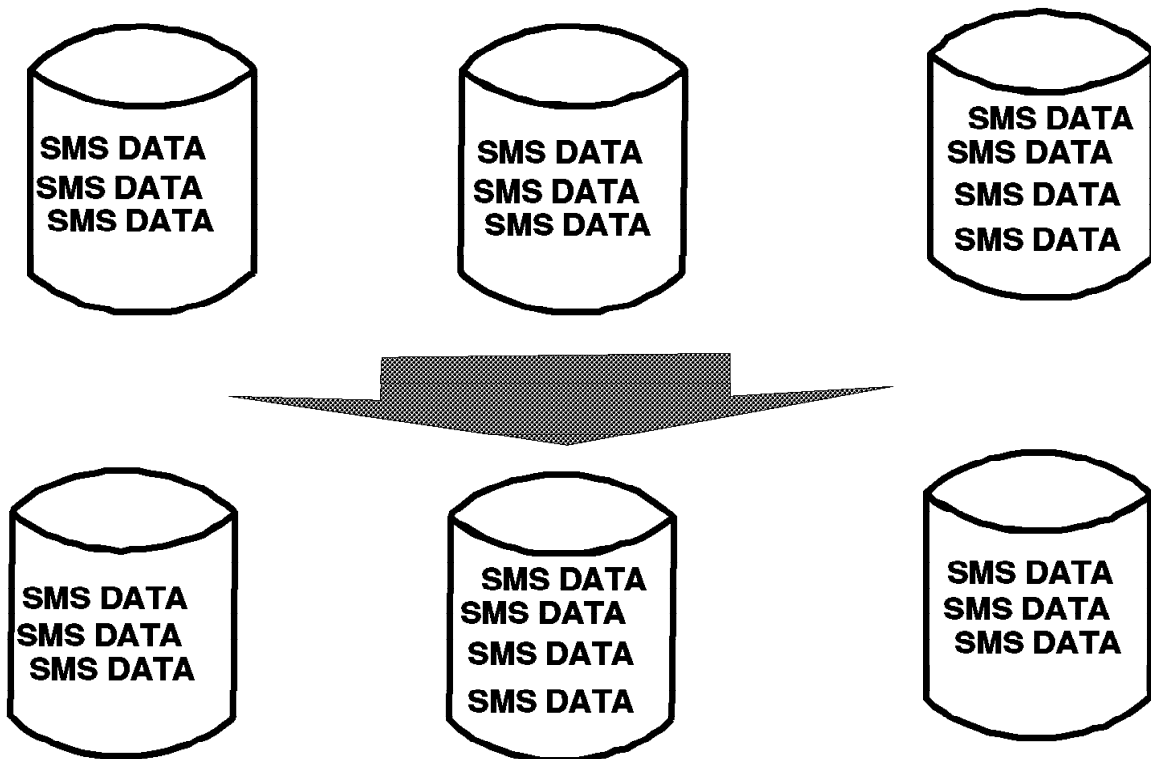


Staged Data Conversion Alternatives ...



Staged Data Conversion Alternatives ...

DUMP, INITIALIZE, RESTORE



Staged Data Conversion Recommendations

- Move unallocated data during prime time
- Use seed DASD when possible
- Allow time for recovery in each window
- Avoid CONVERTV (use dump, initialize, and restore)

Staged Data Conversion Questions

1. Is seed DASD available?

2. Which conversion method is to be used?

- Move data to seed DASD volumes
- Dump and restore data to same volumes
- Both

3. When will the data be converted to SMS-management?

- Prime shift
- Weeknight window
- Weekend window

Staged Data Conversion Table

Data Type				
Classification box				
Phase number				
Filter				
Seed DASD				
Volumes				
GBs				
Conversion Method				
Conversion Time				
Prime shift				
Weeknight window				
Weekend window				

Sample Staged Data Conversion

Data Type	WK	WK	TS	TS
Classification box	1	1	2-4	2-4
Phase number	1	2	3	4
Filter	TEMP+ USERID	TEMP	HLQ+DSN	HLQ+DSN
Seed DASD				
Volumes	6	6	6	6
GBs	13	13	13	13
Conversion Method	Seed DASD	Seed DASD	Seed DASD	Seed DASD
Conversion Time				
Prime shift	N	Y	Y	Y
Weeknight window	1	1	1	1
Weekend window				

Sample Staged Data Conversion ...

Data Type	TS	T	T	
Classification box	2-4	5-7	5-7	
Phase number	5	6	7	
Filter	HLQ+DSN	HLQ+2LQ +GDS	HLQ+2LQ +GDS	
Seed DASD				
Volumes	6	6	6	
GBs	13	13	13	
Conversion Method	Seed DASD	Seed DASD	Seed DASD	
Conversion Time				
Prime shift	Y	Y	Y	
Weeknight window	1	1	1	
Weekend window				

Sample Staged Data Conversion ...

Data Type	BP	BP	BP	
Classification box	8-11	8-11	8-11	
Phase number	8	9	10	
Filter	HLQ+2LQ +GDS	HLQ+2LQ +GDS	HLQ+2LQ +GDS	
Seed DASD				
Volumes	6	6	6	
GBs	13	13	13	
Conversion Method	Seed DASD	Both	Both	
Conversion Time				
Prime shift	Y	N	N	
Weeknight window	1			
Weekend window		1	1	

Staged Data Conversion

- Day 1 - prime shift -
- Day 1 - weeknight window -
- Days 2 and 3 - weekend window -
- Day 4 - prime shift -
- Day 4 - weeknight window -
- Day 5 - prime shift -
- Day 5 - weeknight window -
- Day 6 - prime shift -
- Day 6 - weeknight window -
- Day 7 - prime shift -
- Day 7 - weeknight window -
- Day 8 - prime shift -
- Day 8 - weeknight window -
- Days 9 and 10 - weekend window -

Staged Data Conversion ...

- Day 11 - prime shift -
- Day 11 - weeknight window -
- Day 12 - prime shift -
- Day 12 - weeknight window -
- Day 13 - prime shift -
- Day 13 - weeknight window -
- Day 14 - prime shift -
- Day 14 - weeknight window -
- Day 15 - prime shift -
- Day 15 - weeknight window -
- Days 16 and 17 - weekend window -
- Days 23 and 24 - weekend window -

Sample Staged Data Conversion Phases

- Day 1 - weeknight window - phase 1
 - Initialize seed DASD for work pilot
 - Activate ACS for work pilot data
- Days 2 and 3 - weekend window - phases 2 and 3
 - Initialize nonmanaged work volumes for SMS
 - Activate ACS for all work data
 - Initialize seed DASD for both TSO pilots
 - Activate ACS for first TSO pilot data
 - Move all first TSO pilot data
- Day 4 - prime shift - phase 4
 - Activate ACS for second TSO pilot data
 - Move inactive second TSO pilot data
- Day 4 - weeknight window - phase 4
 - Move remaining second TSO pilot data
- Day 5 - prime shift - phase 5
 - Initialize seed DASD for all TSO data
 - Activate ACS for all TSO data
 - Move all inactive TSO data
- Day 5 - weeknight window - phase 5
 - Move all remaining TSO data
 - Consolidate nonmanaged data on old TSO volumes to reclaim seed DASD

Sample Staged Data Conversion Phases ...

- Day 6 - prime shift - phase 6
 - Initialize seed DASD for test pilot
 - Activate ACS for test pilot data
 - Move inactive test pilot data
- Day 6 - weeknight window - phase 6
 - Move remaining test pilot data
- Day 7 - prime shift
 - No data conversion
- Day 7 - weeknight window
 - No data conversion
- Day 8 - prime shift - phase 7
 - Initialize seed DASD for all test
 - Activate ACS for all test data
 - Move all inactive test data
- Day 8 - weeknight window - phase 7
 - Move all remaining test data
 - Consolidate nonmanaged data on old test volumes to reclaim seed DASD

Sample Staged Data Conversion Phases ...

- Days 9 and 10 - weekend window - phase 8
 - Initialize seed DASD for batch production pilot
 - Activate ACS for batch production pilot data
 - Move all batch production pilot data
- Day 11 - prime shift
 - No data conversion
- Day 11 - weeknight window
 - No data conversion
- Day 12 - prime shift
 - No data conversion
- Day 12 - weeknight window
 - No data conversion
- Day 13 - prime shift
 - No data conversion
- Day 13 - weeknight window
 - No data conversion
- Day 14 - prime shift
 - No data conversion
- Day 14 - weeknight window
 - No data conversion

Sample Staged Data Conversion Phases ...

- Day 15 - prime shift - phase 9
 - No data conversion
- Day 15 - weeknight window
 - No data conversion
- Days 16 and 17 - weekend window - phase 9
 - Dump old batch production volumes
 - Initialize selected old batch production volumes to SMS
 - Activate ACS for first-half batch production data
 - Restore all first-half batch production data to SMS and second-half batch production data to remaining old batch production volumes
- Days 23 and 24 - weekend window - phase 10
 - Dump remaining old batch production volumes
 - Initialize remaining old batch production volumes to SMS
 - Activate ACS for all batch production data
 - Restore all remaining batch production data

International Technical Support Organization**DFSMS FIT:****Fast Implementation Techniques Forms and Foils****October 1995****Publication No. SG24-2570-00**

Your feedback is very important to help us maintain the quality of ITSO Bulletins. **Please fill out this questionnaire and return it using one of the following methods:**

- Mail it to the address on the back (postage paid in U.S. only)
- Give it to an IBM marketing representative for mailing
- Fax it to: Your International Access Code + 1 914 432 8246
- Send a note to REDBOOK@VNET.IBM.COM

Please rate on a scale of 1 to 5 the subjects below.

(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction	_____		
Organization of the book	_____	Grammar/punctuation/spelling	_____
Accuracy of the information	_____	Ease of reading and understanding	_____
Relevance of the information	_____	Ease of finding information	_____
Completeness of the information	_____	Level of technical detail	_____
Value of illustrations	_____	Print quality	_____

Please answer the following questions:

- a) If you are an employee of IBM or its subsidiaries:
- | | | |
|--|----------|---------|
| Do you provide billable services for 20% or more of your time? | Yes_____ | No_____ |
| Are you in a Services Organization? | Yes_____ | No_____ |
- b) Are you working in the USA? Yes_____ No_____
- c) Was the Bulletin published in time for your needs? Yes_____ No_____
- d) Did this Bulletin meet your needs? Yes_____ No_____

If no, please explain:

What other topics would you like to see in this Bulletin?

What other Technical Bulletins would you like to see published?

Comments/Suggestions: (THANK YOU FOR YOUR FEEDBACK!)

Name

Address

Company or Organization

Phone No.



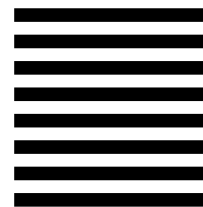
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM International Technical Support Organization
Department 471, Building 070B
5600 COTTLE ROAD
SAN JOSE CA
USA 95193-0001



Fold and Tape

Please do not staple

Fold and Tape



Printed in U.S.A.

SG24-2570-00

