Introduction to an SMS Environment – Structure Summary

- Base Configuration – installation defaults
- DATACLAS – JCL defaults
- STORCLAS* – access requirements / required for SMS
- MGMTCLAS – migration / backup attributes
- STORGRP* – groups of volumes
- * at minimum, you need a storage class and storage group
Introduction to ACS Environment

• What is an ACS Routine?
  • User written code
  • Determines which SMS classes and storage groups are assigned
  • Used for both data sets and objects
  • One per type of construct
  • They run at ALLOCATION time
  • Process in order:
    • DATACLAS
    • STORCLAS
    • MGMTCLAS
    • STORGRP
  • REQUIRED even if nothing is set
Introduction to ACS Environment (cont)

- **Write Your Routines**
  - Edit via your favorite editor
  - Saved in a PDS
  - Easy to read / review

- **Translating ACS Routines**
  - Done via ISMF
  - Checks for syntax errors
  - Converts (compiles) ACS source into object and stores it into the SCDS

- **Validating the SMS Configuration**
  - Also done through ISMF
  - Verifies that all classes/groups assigned in the ACS routines exist

- **Activating the SMS Configuration**
  - Loads the SCDS into the ACDS
  - 3 Methods
    - SETSMS SCDS(scdsname) operator command
    - ISMF option 8 (Control Data Set Application) then select option 5 (Activate)
    - Type ACTIVATE on the ISMF command line
ACS Routine Process Flow

1. **JOB Starts**
2. **DATACLAS Routine**
3. **STORCLAS Routine**
4. **STORGRP Routine**
5. **Volume Selection**
6. **Data Set Allocated**

- **MGMTCLAS Routine**
- **MC**
- **STORCLAS Routine**
- **DC**
- **STORGRP Routine**
- **SG**
- **Vol**
Introduction to ACS Environment (cont)

• ACS General Rules

  • *Know your logic before you code*

  • Keep them simple and straightforward
    • Minimize exceptions
    • Maximize FILTLIST usage

  • Keep them easy to maintain and understand
    • Use SELECT instead of IF when possible
    • EXIT the routine as soon as possible
    • Use OTHERWISE whenever possible
    • Comments, comments, comments
Introduction to ACS Environment (cont)

- ACS Language Statements
  - **PROC** - beginning of routine
  - **FILTLIST** – defines filter criteria
  - **DO** – start of statement group
  - **SELECT** – defines a set of conditional statements
  - **IF** – conditional statement
  - **SET** – assigns a read/write variable
  - **WRITE** – sends message to end user
  - **EXIT** – immediately terminates ACS routine
  - **END** – end of statement group
  - /* **COMMENT** */ - comments a line
Read Variables

**READ ONLY Variables**
- 47 different variables
- Majority of the ACS variables
- Contain data and system information
- Reflect what is known at the time of the request
- Can only be used for comparison

**Examples:**
- \&DSORG
- \&DSNTYPE
- \&SIZE
- \&HLQ

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

**The PROC statement must identify which R/W variable it is setting**
A Few “Gotchas”

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

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

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

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

- && is AND, | is OR

- Watch for fall-through logic in your IF and SELECT
Introduction to ACS Environment (cont)

- Write the ACS Routines
  - Saved in a text format

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

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

- Activate the SMS Configuration

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

See your handout and start the lab!