

IBM Education Assistance for z/OS V2R1

Item: IEBCOPY Member Name Pattern Matching
IEBCOPY COPYGROUP
PDSE Space Utilization and Performance Improvement
Element/Component: DFSMSdfp Utilities/PDSE



Agenda

- Trademarks
- Presentation Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Migration & Coexistence Considerations
- Installation
- Presentation Summary
- Appendix



Trademarks

- See url <http://www.ibm.com/legal/copytrade.shtml> for a list of trademarks.



Presentation Objectives

- Learn how to use these improvements for usability and performance in IEBCOPY and PDSE (partitioned data set extended):
 - IEBCOPY COPYGROUP (improvement to COPYGRP)
 - IEBCOPY member name pattern matching
 - PDSE space utilization and performance enhancement (PDSE version 2)



Overview

- Problem Statement / Need Addressed
 - IEBCOPY handles member aliases differently for PDS and PDSE
 - IEBCOPY should support wild cards when identifying members
 - Partial space release should release more space in a PDSE
- Solution
 - Implement new COPYGROUP command that is similar to COPYGRP
 - Support * and % (“wild cards”) in member names
 - Support PDSE version 2
- Benefit / Value
 - Improved usability



Usage & Invocation – IEBCOPY Improved Alias Handling with COPYGROUP

- The purpose of the existing COPYGRP command is to copy members from a PDS or PDSE to a PDS or PDSE.
 - If both the source and destination are PDSs, if you do not specify an alias, it is ignored. This is a common problem with load modules.
 - If either the source or destination is a PDSE, then IEBCOPY automatically copies all aliases. Especially useful with names > 8 chars.
- Users complained about this inconsistency.
- Instead of a new keyword option that means “also copy all aliases”, a new COPYGROUP command will always copy all aliases

- Example:

```
COPYGROUP INDD=INDD,OUT=OUTDD
```

```
SELECT MEMBER=(MOD3,UPDATE4)
```

If these members have any aliases, they will also be copied



Usage & Invocation – IEBCOPY Member Name Pattern Matching

- On the SELECT and EXCLUDE statements you now can specify pattern matching characters (“wild cards”)
 - The asterisk character (*) indicates that any characters within the member name that follow the pattern mask are acceptable.
 - The percent character (%) indicates that any one character within the member name in that position is acceptable.
- It must be part of a COPYGROUP operation
- On an EXCLUDE statement the SELECT statement must specify MEMBER= with filter pattern masking also.
- Filtering will only be performed with primary member names (not aliases)
- You cannot rename the member when it is identified with pattern matching



Usage & Invocation – IEBCOPY Member Name Pattern Matching

Examples

Pattern	Description	Examples of Names That Would Match	Examples of Names That Would <u>Not</u> Match
*	Any name	ABC, A12#\$F	None
CD*	Names starting with 'CD'	CDEFGH	ABCD
*CD	Names ending with 'CD'	ABCD	CDEFGH
CD%	Three-character names starting with 'CD'	CD\$	CD, CDEFG
*CD%	Names with 2nd and 3 rd to last characters being 'CD'	CDE, BCDE, BBCDE	CD
%%	Any two-character name	AB, BC, CD	ABC
**	Not a valid member name pattern	Not Applicable	Not Applicable



Usage & Invocation – Space Utilization & Performance Enhancement (PDSE V2)

- A new type of PDSE is supported in z/OS 2.1
- The purposes are:
 - Change the internal format for improved performance
 - Increase the chances that partial space release will release space due to:
 - SPACE=(,RLSE) in the DD statement when writing
 - Space management by DFSMSHsm for management class
 - Space management by DFSMSdss
 - PARTREL macro



Usage & Invocation – Space Utilization & Performance Enhancement (PDSE V2)

- The only way to cause the new format is to create a PDSE in one of these ways:
 - DSNTYPE=(LIBRARY,2) on the DD statement or dynamic allocation
 - Dynamic allocation key is DALDSNV (X'802C')
 - Dynamic information retrieval key is DINRDSNV (X'802C')
 - Version 2 in data class
 - REFDD referring to a DD statement with DSNTYPE=(LIBRARY,2)
 - LIKE= does not copy the version number
- The ISITMGD macro will return the version number (1 or 2).
- Type 14/15 SMF record mapped by IFGSMF14
 - See SMF14DSVER
- DFSMSdss RESTORE of a PDSE will always result in the dumped version; a pre-allocated PDSE version will become what was dumped
- IDCAMS DCOLLECT was not changed.



Interactions & Dependencies

- Software Dependencies
 - None
- Hardware Dependencies
 - None
- Exploiters
 - Will be supported by ISPF and IPT



Migration & Coexistence Considerations

- Migration actions: none
- Coexistence APARs for releases 12 and 13 for PDSE version 2
 - OA39530, PDSE
 - OA40844, BAM, basic access methods
 - OA41790, ISITMGD macro

The PTFs allow reading and writing but not creating version 2 PDSEs



Installation

- Install coexistence PTFs on downlevel systems that share DASD
- The storage administrator can specify this in the IGDSMSxx member of PARMLIB:
 - PDSE_VERSION({1|2})
 - It identifies the version of the type of data set if it is a PDSE
- It would be safest to avoid PDSE_VERSION(2) until all sharing and backup systems are at 2.1.
 - Then it would be best to set PDSE_VERSION(2)



Presentation Summary

- You can use IEBCOPY to handle PDS member aliases more reliably
- It will be easier to specify member names to IEBCOPY
- Space is more likely to be released when you do partial space release for a PDSE if it is version 2



Appendix

- SC23-6864-0, *z/OS V2R1.0 DFSMSdfp Utilities*
- SC23-6860-0, *z/OS V2R1.0 DFSMSdfp Storage Administration*
- SA23-1385-0, *z/OS V2R1.0 MVS JCL Reference*

