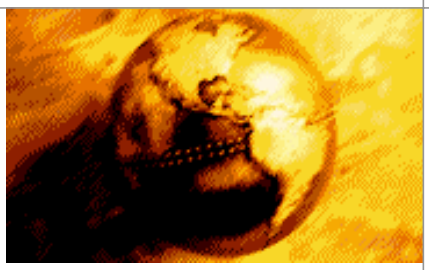
Systems Guide

10101010101011
10101010101011

Release 9.0.2
October 2000

011
1001



CA-Disk™

2000 Computer Associates International, Inc.
All Rights Reserved

Computer Associates International, Inc.
Storage Management Division
625 East Carnegie Drive, Suite 200
San Bernardino, California 92408-3510

01DS100P012-902

Proprietary/Trade Secret Notice and Warranty Disclaimer

This documentation and the software it describes, constitute the proprietary, confidential, and valuable trade secret information of Computer Associates International, Inc., and is licensed either "AS IS" or with a limited warranty, as set forth in the applicable license agreement. NO OTHER WARRANTY IS EXPRESSED AND NONE SHALL BE IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE OR FOR A PARTICULAR PURPOSE.

Right to Revise

CA-Disk is undergoing constant revision, refinement, and expansion to include additional features. The current documentation is believed to be accurate and reflects the current version of the software. Important information not available when this book was published is contained in the Cover Letter that accompanied the distribution tape.

Computer Associates International, Inc., reserves the right to revise this publication from time to time and to make changes in the content hereof without the obligation to notify any person or entity of such revisions or changes. No part of this Installation Guide can be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, for any purpose, without the express written permission of Computer Associates, Inc.

Use, duplication or disclosure by the Government is subject to restrictions as set forth in Sections 52.227-14 and 52.227-19 of the FAR Commercial Computer Software Restricted Rights Clause and/or the Rights in Technical Data and Computer Software Clause at DFAR 252.227.7013 as applicable.

© 2000 Computer Associates International, Inc., All Rights Reserved.

Printed in the USA

Customer Support

In the United States, contact:
Computer Associates International, Inc.
OS/390 Storage
10811 International Drive
Rancho Cordova, California 95670
(800) 889-0226

Outside the United States, contact your local Computer Associates customer support representative.

Contents

CHAPTER 1 *Files Data Set Management* *13*

General Information 13
Integrity *13*
Physical Blocks *14*
Logical Blocks *14*
Logical Records *14*
Files Data Set Structure 14
Control Blocks *14*
Allocation Control Record *14*
File Control Block *15*
Logical Index Block *15*
Logical Data Block *15*
Subfiles of the Files Data Set 15
Restrictions *16*
Subfile Entry Definition *16*
Subfile Definition Parameters *16*
Files Data Set Management 17
Formatting the Files Data Set *17*
Files Data Set Status Report *18*
Field Descriptions — Status Report *18*
Files Data Set Capacity 20
Estimating Capacity Requirements *20*
Estimating Subfile Capacity Requirements *20*
Estimating Physical Space Requirements *21*
Performance Considerations *21*
Capacity Considerations 21
Block Splits *21*
Overflow Blocks *21*
Extending Allocation *21*
Files Data Set Utilities 22

Unload, Reload, And Reorg *22*
Unload *22*
UNLOAD Command and Parameters *24*
RELOAD *25*
RELOAD Command and Parameters *25*
REORG Subfiles *26*
Files Data Set Logging **26**
Activation *26*
Naming the Log Data Sets *27*
Format of the Log Data Sets *27*
Managing the Logs *27*
Recovery *28*
Running the Utility *28*
JCL — Files Data Set Recovery Utility *29*
FRECOVER Command and Parameters *29*
Sample - Files Data Set Recovery Report *30*
Subfile Dump Utility **30**
JCL — Subfile Dump Utility *30*
Subfile Dump Command for the Files Data Set *31*
Sample Subfile Dump Utility *31*
Merging Files Data Sets **32**
Common File Handler Interface **33**
File Handler Interface *34*
Function Service Types *34*
Interface for Sequential Reads *35*
Processing Support *36*
Support Modules and Functions *37*
Service Level Supporting Modules *38*

CHAPTER 2 *Tape Management* *39*

Allocation Considerations **40**
DDNAMES Used *40*
Dynamic Allocation Versus Supplied JCL *40*
Allocation Support for Masstor's M860 Device *41*
Allocation Support for StorageTek's Redwood Device *41*
Allocation Support for IBM's Magstar Device *41*
Producing Duplicate Archive Tapes Concurrently *42*
Specifying Blocksizes for Archive/Backup Tapes *42*
Creating Multivolume Tape Data Sets *43*
IDRC Support in 3480/90 Cartridge Devices *43*
Running Archive/Backup in Simulate Mode *44*
Running Multiple Archive/Backup Jobs Concurrently *44*
Cataloging the Tape Archives *44*
Tape GDG Support (No Longer Recommended) *44*
Assigning Tape Expiration Dates *45*
Tapepool Considerations **45**
Tapepool Updates from ISPF *47*
Tapepool Update Utility *48*
LIST Command *48*
ADD Command *48*

DELETE Command *49*
ERASE Command *49*
Tape and Cartridge Switching — Length Considerations **49**
For TAPE (3420) Devices *50*
For Cartridge (3480/3490) Devices *51*
Tape Format **51**

CHAPTER 3 *Tailoring Options* *52*

General Information **52**
Suggested System Parameters **53**
System Parameters for Reports: *53*
System Parameters for Messages *54*
System Parameter for Lines Per Page *54*
System Parameter for Separator Pages *54*
User Exit for All Sysout Print Lines *54*
System Parameter for Files Data Set Logging *54*
System Parameters for Archive/Backup *55*
System Parameters for Restore/Recover *55*
Performance Tips **55**
Report Processing *55*
PDS Compress *55*
Files Data Set *56*
I/O Buffering and Memory Requirements *56*
Archive/Backup Considerations **57**
Activating Data Compression *57*
Using CA-Compress Within The Archives *58*
Specifying the Archive Medium *59*
Using Multiple Types of Media *60*
Naming the Archives *61*
JCL Overrides and DSNAME Usage *62*
Assigning an Expiration Date to the Archive Data Set *62*
Archiving to Disk: Requirements and Recommendations *62*
Creating and Using Dual Primary Volumes *67*
Dual Primary Volume Feature Overview *68*
Dual Primary Volume Details by Function *69*
Using Archive Functions to Manage Active Data **71**
Archiving the Archives is Not Allowed *72*
Archive Integrity and Security **73**
Archive Tape Protection *73*
Checks Against Overwriting Valid Tapes *73*
Tape Status After Abends *74*
Archive Disk Protection *75*
Processing PDSs that Contain Anomalies **75**
Specifying your Options for PDS Error Handling *76*
PDS Anomaly Code Descriptions and Notes *77*
Processing Defined for Each Anomaly *77*
Anomalous Conditions Not Regarded as Errors *80*
Error Messages and Associated Processing for Anomalies *81*
Implementing Support for Masstor M860 ..... **81**

Distinguishing the M860 Device *81*
Additional Implementation Needs for the M860 ..... *82*
Implementing Support for StorageTek Redwood **82**
Distinguishing the Redwood Device *83*
Additional Implementation Needs for Redwood *83*
Implementing Support for IBM's Magstar **84**
Distinguishing the Magstar Device *84*
Additional Implementation Needs and Notes for Magstar *85*
ISPF Custom Reports **85**
Coding ISPF Report Data Conversion Modules *87*
Coding ISPF Report Sort Conversion Modules *88*
User-Specified Condition Codes **89**
DASD Threshold Manager **91**
Overview *91*
Threshold Manager Control Statement Syntax *92*
Threshold Manager Control Hierarchy *94*
JCL - Threshold Manager *94*
Keyword, Parameter, and Subparameter Definitions *95*
Implementation Considerations *98*
Threshold Manager User Exit *99*
Creating and Using Dual Primary Volumes *100*
Important Considerations of the Dual Primary Volume Process *100*
Dual Primary Volume Feature Overview *101*
Dual Primary Volume Details by Function *101*

CHAPTER 4 *Sysparms* *105*

System Parameter Names Reserved for Users *106*
Overriding Sysparms Instream *106*
System Parameters **107**
System Parameter Definitions **122**

CHAPTER 5 *User Exits* *201*

General Information **201**
Using Previous Releases of User Exits *202*
User Exit Cross Reference Matrix **203**
User Exit Descriptions **206**

CHAPTER 6 *Message Formatting* *266*

Module Name — ADSUT028 ..... *266*
Message Formatting **267**

CHAPTER 7 *Problem Analysis/Aids* *269*

When You Call In **270**

Sending Information to the Technical Support Center 272
Diagnostic Tools 272
Module Intercept Control Stack *272*
Finding the Module Intercept Control Stack *273*
Example of Module Intercept Control Stack *274*
MI Fast Path *275*
DMS Diagnostic Area *276*
Module Intercept Tracing *276*
MI Trace Control Statement Format *279*
MI Trace Output *280*
Archive Tape Analysis Utility 281
Executing the Utility *281*
Modifying the SYSUT1 DD Statement *282*
Command Input *282*
Sample Report *283*
User Exit — Archive Analysis *284*
Archive Extract Utility 284
Output from the Extract Utility *284*
Executing the Extract Utility *285*
EXTRACT command *285*
JCL Example *286*
Modifying the SYSUT1 and SYSUT2 DD Statements *286*
Archive Fix Utility 287
Modifying the SYSUT1 DD Statement *288*
Command Input *288*
User Exit — Fix Utility: Parameters Passed *289*
Repair Archvols (REPARCH) Utility 289
Processing Considerations *290*
Output *290*
JCL — REPARCH Utility *290*
REPARCH Command and Parameters *291*
Load Library Status Utility 292
JCL — Load Library Status Utility *292*
Loadlib Command and Parameters *293*
Sample Report *293*
Track Dump Utility 294
TRACKDMP Command and Parameters *294*
Diagnostic DD Statements 295
Archive Tape Recovery Utility 296
Executing the Utility *296*
Modifying the SYSUT1 DD Statement *297*
Command Input *297*
Sample Report *298*
Condition Codes *298*

CHAPTER 8 *Modifying CA-Disk* *299*

Assembly and Link-Edit of CA-Disk Modules 299
Issuing Messages 301

Relinking TSO and ISPF Control Modules 301
Adding Non-CA-Disk Modules 301
Coding for an Interactive Environment 301
Module Call Interface 302
User Exits *302*
Parameter Library Support - Common Parmlib Interface 302
Common Sysparms Interface 304
System Parameter Validation - Parmedit Member 304
Adding User Functions to CA-Disk ISPF 305
Modifying CA-Disk ISPF Functions 307

CHAPTER 9 *DSCL Processing* *308*

Control Block Structure 308

CHAPTER 10 *Security Processing* *311*

Statement of Security Direction 311
Security Overview *312*
Resources Protected by the CA-Disk Base System 313
Resources Protected by the Security Interfaces 314
Resources not Protected by CA-Disk 316
Security and the CA-Disk Auto-Restore Function 317
CA-Disk Security Interface General Information *317*
Running Without a CA-Disk Security Interface 318
Password-Indicated Dataset Support 318
What Authority Does CA-Disk Look For? 318
CA-Disk Authority Level Translation Tables for ACF2 ..... *319*
Protection of Subfiles Under ACF2 ..... *321*
Protection of Commands Under ACF2 ..... *321*
Protection of User Datasets Under ACF2 ..... *323*
Functions that Need no Dataset/Volume Authorization *326*
Authority Level Translation Tables for Top Secret *327*
Protection of Subfiles Under Top Secret *328*
Protection of Commands Under Top Secret *328*
Protection of User Datasets Under Top Secret *330*
Functions that Need no Dataset/Volume Authorization *334*
Authority Level Translation Tables for RACF *335*
Protection of Subfiles Under RACF *336*
Protection of Commands Under RACF *337*
Protection of User Datasets Under RACF *338*
Functions that Need no Dataset/Volume Authorization *344*
RACF Discrete Profile Handling 344
RACF Terms Defined *344*
RACF and VSAM Datasets *345*
Backing Up RACF Discrete Profiles (RACFBKUP) *345*
Restoring RACF-Indicated Datasets Having no Profiles *346*

Reconciling CA-Disk and RACF Records *346*
Check Command Parameter Description *348*
RACF Checking Logic Summary in CA-Disk *349*
Use of RACF Macros **350**
RACF User Exits **350**
RACF Installation Exits **351**
DSNINDEX RACF Flags *352*
Format of CA-Disk-Saved Profiles **352**
RACF Encode Profile (Format 2) *352*
Conversion of CA-Disk RACF Profile Encoding *353*
Management of CA-Disk-Saved Profiles *354*
CA-Disk-Saved Profile List Utility 2 ..... **354**
DD Statement Explanations *354*
Select Command *354*
CA-Disk-Saved Profile Maintenance Utility **355**
Four different commands are accepted by the utility: *356*
Display Command *356*
Delete Command *357*
Create Command *357*
Rename Command *358*
Execution Examples: *358*
Changing the Value of RACFDVOL **359**

CHAPTER 11 *Report Processing* *360*

Overview **360**
REPORTNAME Table — Member REPORTNM *361*
Format-1 / Format-4 DSCB Appendage Fields **362**
Tables for Report Formatting **362**
Format Table Examples *364*
Header Entry Examples *365*
Detail Entry Examples *366*
Total Entry Examples *367*
Report Modification Example *367*
Example of Owner Table *369*
Standard Multiple Volume Dictionary Report Table *370*
Example of Record Format for Owner Record *371*
Standard Multiple Volume Dictionary Record Format *371*
Report Logic Diagram **373**
Report Processing - ISAM *375*
ISAM Space Restrictions *377*
VTOC ENTRIES- Format 1, 2 and 3 DSCBs *377*
ISAM Report Logic Overview *378*
ISAM Variables Examined *380*
ISAM Format-2 DSCB Offsets *382*

CHAPTER 12 *Algorithms* *383*

Bytes that Fit on an Output Track *384*

VSAM Idle Space Calculation 384
VSAM New Name Algorithm *385*
Data Compression/Decompression 386
Data Compression Techniques *387*
Data Compression Technique 0 (DCCMPEX0) *388*
Compression Results *389*
Overview of Space Allocation for Archival to DISK 389
Space Allocation Algorithm *390*
General Comparison of Allocation in Prior Releases *391*
Creating a Disk Archive Data Set *391*
PDS Compression Threshold Calculations 392
Directory Used Space Percentage *393*
Member Data Used Space Percentage *393*
PDS Compression Space Calculations 394
Directory Block Allocation *394*
Member Data Space Allocation *394*
Secondary Space Allocation *395*

CHAPTER 13 *Record Formats* *396*

Report Record Format - (RPTNALLOCS) 398
Volume Allocation Summary Report — ALLOCS *398*
Table Used for Printing ALLOCS Report *399*
Report Record Format — RPTNAMVDIC 400
A-FORM of MVDICT Report *400*
Table Used to Print MVD Report by Owner *401*
Report Record Format — (RPTNATTRBS) 402
Volume Attribute Summary Report *402*
Table Used to Print ATTRIBS Report *403*
Report Record Format — (RPTNBMVDIC) 404
B-FORM MVDICT Report *404*
Table Used to Print MVDICT Report *405*
Report Record Format — (RPTNDEVDSB) 406
BIlling Device Report *406*
Table Used for Printing Billing Device Report *407*
Report Record Format — (RPTNDISTRS) 408
Volume Distribution Summary Report *408*
Table Used for Printing DISTRS Report *409*
Report Record Format — (RPTNDSNDSB) 410
Billing by User Code *410*
Table Used for Printing Billing by User Code *411*
Report Record Layout — (RPTNDSNX) 412
Record Report Format *413*
Report Record Layout — (RPTNVOLX) 414
Report Record Format *415*
Report Record Format — (RPTNDSINDX) 415
Data Set Index Report *415*
Table Used for Printing Data Set Index Report *416*
Report Record Format — (RPTNDSUTIL) 417

Data Set Utilization Report *417*
Table Used for Printing DSUTIL Report *418*
Report Record Format — RPTNDAYWEK **419**
Utilization by Date and Time Report *419*
Table Used for Printing DAYWEK Report *420*
Report Record Format — (RPTNFREESP) **421**
Volume Freespace Report *421*
Table Used to Print Volume Freespace Report *422*
Report Record Format — (RPTNLAYOUT) **423**
Volume Layout Report *423*
Table Used for Volume Layout Report *424*
Report Record Format — (RPTNMAPPER) **425**
Volume Mapping Report *425*
Table Used to Print Volume Mapping Report *426*
Report Record Format — (RPTNMEMBER) **427**
PDS Directory / Member Report *427*
Table Used to Print MEMBER Report *428*
Report Record Format — (RPTNMVDICT) **429**
Multiple Volume Data Set Dictionary Report *429*
Table Used to Print MVDICT Report *430*
Report Record Format — (RPTNPDSTAT) **431**
Partitioned Data Set Status Report *431*
Table Used to Print PDS Status Report *432*
Report Record Format — (RPTNPOCCHH, RPTNPODUMP, and RPTNPOTTRD) **433**
Partitioned Data Set Directory (CCHHR, DUMP, TTRD) *433*
Report Record Format — (RPTNPOXREF, RPTNPO1REF, and RPTNPO2REF) **433**
Data Set Name Cross-Reference for PDS Member Names *433*
Report Record Format — (RPTNRACFDN) **434**
RACF Profile X-REF in Dsname Sequence *434*
Table Used to Print X-REF Report *435*
Report Record Format — (RPTNRACFPF) **436**
RACF Profile X-REF in Profile Name Sequence *436*
Table Used to Print X-REF Report *437*
Report Record Format — (RPTNSVDICT) **438**
Single Volume Dictionary Report *438*
Table Used to Print SVDICT Report *439*
Report Record Format — (RPTNVOLDSB) **440**
Billing Volume Report *440*
Table Used for Printing Billing Volume Report *441*
Report Record Format — (ARCHDSNVOL) **442**
Archive/Backup/Delete/Expire Report *442*
Table Used to Print Archive Report *443*
Report Record Format — (ARCHVOLDSN) **444**
Archive/Backup/Delete/Expire Report *444*
Table Used to Print Archive / Retain Report *445*
Report Record Format — (MIGRDSNVOL) **446**
Migrate Report *446*
Table Used for Printing Migrate Report *447*
Report Record Format — (MIGRVOLDSN) **448**

Migrate Report *448*
Table Used for Printing Migrate Report in VOL/DSN Order *449*
Report Record Format — (RPTSTATS) **450**
Restored Data Sets Statistics Report *450*
Table Used to Print Restored Data Sets Statistics Report *450*
Report Record Format — (RPTPLST2) **451**
Restored Data Sets Tape Pull List Report *451*
Table Used to Print Restored Data Sets Tape Pull List Report *451*
Report Record Format — (VSAMCLDATA) **452**
VSAM Cluster Data Record *452*
Report Record Format — (RPTPLST1) **454**
Restored Data Sets Disk Archive List Report *454*
Table Used to Print Restored Data Sets Disk Archive List Report *455*
Report Record Format — (RPTDSNEX) **456**
VBACKUP Extent Report *456*
Table Used for Printing VBACKUP Extent Report *457*
Subfile Record Format — (ARCHCMDS) **458**
Deferred Archive Requests *458*
Subfile Record Format — (ARCHVOLS) **459**
Subfile Record Format — (DASDSPCB) **460**
DASD BILLING *460*
Subfile Record Format — (DMSPARMS) **461**
TSO Dynamic Restore *461*
Subfile Record Format — (DMSPOOLS) **461**
For Tapes in the Tape Pools *461*
Subfile Record Format — (DSNINDEX) **462**
For Archived Data Sets *462*
Subfile Record Format — (MIGRECAT) **465**
PS Migration to Tape *465*
Subfile Record Format — (RACFENCD) **465**
RACF Encoded Profile Names *465*
Subfile Record Format — (RESTCMDS) **466**
Deferred Restore Requests *466*
Subfile Record Format — (RETEXCLD) **467**
Grace Periods for Restored Data Sets *467*
Record Formats — Files Data Set Log Records **467**
Files Data Set Log Record Layout *467*
Archive Tape Organization **468**
Archive Tape Format *468*
Archive Tape Format (Volume-level Processing) **473**
Archive Tape Format for Volume-level Backup Tapes *473*

CHAPTER 14 *PARMLIB* *474*

Purpose **474**
Updating Member Data **475**
Backup Considerations **475**

PARMLIB Specifications 476
Examples *476*
Special Rules for NAMEREFS *476*
Special Rules for POOLDEFS *477*
PARMLIB Syntax Validation Utility *478*
Alternate DDname for User Tables 478
User PARMLIB Members 479
Billing Members in PARMLIB *480*
Billing User Code Definition Member *480*
DASD Pool List *481*
Auto-Restore DSNAME Pool Member *482*
Auto-Restore Source Volume Pool Member *483*
Exclusion Table for Data Compression *484*
Inclusion Table for Data Compression *484*
Exemption Lists - Generalized *485*
Exemption List for Implicit Archive (Retain) *485*
GDG DASD Generations List *486*
Migration Exemption List *487*
PDS Anomaly Table *487*
Reblocking Member *488*
Retention Period By Prefix List *488*
Retention Control Volume Specification Table *489*
Security List for CA-Disk Data Set Protection *489*
Selection Lists — Generalized *490*
System Parameter List *491*
TSO/ISPF USERID Authorization List *492*
User-Specified Condition Codes *492*

CHAPTER 15 *JCL DD Statements* *494*

DD Statement Explanations 495

APPENDIX A *DSCB Appendage Fields* *499*

Format-1 Fields 499
Format-4 Fields 500

APPENDIX B *Contents of INSTALL Library* *502*

APPENDIX C *ENQ/DEQ/Reserve Usage* *506*

CHAPTER 1

Files Data Set Management

General Information

CA-Disk uses a direct access data set called the Files Data Set to store information about the data sets it operates on. It is a single volume data set that must be in one extent. The files data set contains multiple subfiles of data. The term “subfile” refers to a group of logically related records kept within the files data set. Each subfile is assigned an eight-character name by which it is identified. Subfiles are independent groups of records that do not necessarily have any relationship to records in other subfiles of the data set. They are processed as if they were separate data sets.This technique provides the convenience of having multiple DASD files without the task of having to support each one independently.

An EXCP I/O interface is used to retrieve, insert, update and delete records. Records are referenced by their specific keys within each subfile, but can also be retrieved sequentially by subfile name alone, if desired.

For example, one subfile used by CA-Disk is the data set index record subfile (DSNINDEX). This subfile contains one record for each data set currently in the CA-Disk archives. When a data set is archived, a record is placed in the DSNINDEX subfile. When the data set is restored, the DSNINDEX subfile is accessed to determine the volume serial of the tape or disk containing the archived version of the data set and its location within that volume. These records remain in the subfile until they are deleted by the MERGE or Index maintenance processing routines.

Integrity

The processing technique implemented ensures data set integrity during concurrent update (see “Basic Conventions and Requirements”) in both the Loosely Coupled Multi Processor (LCMP) and Tightly Coupled Multi Processor (TCMP) environments. For greatest efficiency, a cross system enqueue package is recommended in a JES2 environment.

For related information, see the sysparm description for "RSUPPRESn" on page 172.

Except for specified utility functions, all JCL DD statements that reference the files data set can have shared disposition (DISP=SHR) coded in the JCL statement.

Physical Blocks

The physical blocksize of the files data set is the only value common to all of its subfiles. The blocksize is assigned at files initialization time.

Logical Blocks

Internal to CA-Disk processing is the concept of Logical Blocks. A logical block is simply a set of contiguous physical blocks accessed by the EXCP interface. The files data set EXCP I/O interface automatically determines the Logical Block based on the capacity requirements used when the files data set is initialized.

The Logical Block is unique to each subfile unlike the physical blocksize which is common to all subfiles. Depending on the capacity requirements for each subfile, the number of physical blocks per Logical Block can vary. The DSNINDEX subfile currently utilizes the maximum Logical Block which is 10 physical blocks per Logical Block.

Each logical block is accessed by Relative Byte Address (RBA). The relative byte address of each logical block is stored in the first four bytes of each block.

Logical Records

This term describes the data records that are stored in the logical data block. One logical record represents one subfile record. The logical data block will contain a fixed number of logical records.

Files Data Set Structure

Control Blocks

The files data set contains 4 major control blocks. The Allocation Control Record, the File Control Block, the Logical Index Block and the Logical Data Block. At file initialization, each subfile is represented by a subfile Logical Index Block and a subfile Logical Data Block. (The files data set control blocks and their contents are further described in "Record Formats" on page 396.)

Allocation Control Record

The Allocation Control Record (ACR) describes the allocation of the entire files data set. It always resides in the first physical block of the files data set and is predefined offsets into the block.

File Control Block

The File Control Block (FCB) is defined for each subfile and contains the subfile definition entry. (Used by the Files Status Report.) The FCB is divided into two parts. A FCB prefix defines the number of subfiles defined and always follows the allocation control record.

Logical Index Block

Only one Logical Index Block (LIB) is created for every subfile defined by the FILEDEFN member. The LIB contains one entry for every logical data block in the subfile. The logical data block entry contains the highest key value and a four byte RBA pointer to the logical data block that contains the key.

To retrieve a specific record, the logical index block is searched to determine the relative byte address of the logical data block that has a high key equal to or greater that the key of the desired record. The RBA is then used to read the logical data block into memory. The logical data block is then searched for find the record.

Logical Data Block

The Logical Data Block (LLB) contains logical records comprising the subfile. The number of logical records that reside in a single logical data block is equal to the blocking factor of the subfile.

Note: A logical data block will never contain a partial record.

Subfiles of the Files Data Set

ARCHCMDS - A record is placed in this file for each ARCHIVE command to be processed on a deferred basis; this includes TSO and batch requests. The deferred command requests are eventually deleted subsequent to a batch DISPOSE job that processes the queued requests.

ARCHVOLS - A record exists in this file for each archive volume (tape or disk) to which one or more data sets were archived. An entry is also present for each backup copy of the archive volumes.

DASDSPCB - The accumulation of DASD space billing statistics are kept in this file.

DEFERDSP - This file was used prior to Release 7.0. It is no longer documented, and will no longer be supported in subsequent releases. If you are not using the “Deferred Retain” function then you do not need to define this subfile.

DMSPARMS - This file is used only if the TSO DYNAMIC (immediate) RESTORE function is used. It is used to control the number of concurrent dynamic restores in progress.

DMSPOOLS - This file maintains the status of all tape volumes defined to various CA-Disk tape pools. Each volume is marked as being either an available scratch tape or an in-use tape.

DSNINDEX - This file contains an entry for each data set backed up or archived either EXPLICITLY or IMPLICITLY by the system. The records are keyed by data set name, and multiple versions of a given data set can be present.

MIGRECAT - An entry for each data set placed on the backup tape created during MIGRATION OF SEQUENTIAL data sets to tape is maintained in this file to facilitate recataloging all of the data sets should a primary tape be damaged or lost. CA-Disk does not provide any formal automated means for managing the entries in this subfile because there is no way of knowing how long to keep them. For this reason, this subfile will need to be purged occasionally to keep it from becoming excessively large.

RACFENCD - A record is placed in this file to cross-reference each RACF-protected data set and its associated CA-Disk-generated profile.

RESTCMDS - A record is placed in this file for each RESTORE command to be processed on a deferred basis; this includes TSO and batch requests. The deferred command requests are eventually deleted subsequent to a batch DISPOSE job.

RETEXCLD - If sysparm RETEXCLD is specified with a value of Y, a record is placed in this file for each data set restored. An installation-defined grace period is assigned to each. An unexpired grace period prevents implicit archive from archiving the data set again.

Restrictions

A maximum of 12 subfiles is presently supported. However, you can limit this even further by specifying sysparm FILESCNT. For more details, please review the sysparm description for "FILESCNT12" on page 147.

Subfile Entry Definition

All subfiles to be used are defined to CA-Disk by entries in the FILEDEFN member of parmlib. During files initialization, the member entries are read and used to determine how the data set is to be formatted.

A sample set of subfile definition entries is distributed with the system as member FDSAMPLE in parmlib. To activate the definitions, you must create a member named FILEDEFN and copy in the sample values. An entry is required in the FILEDEFN member for each subfile of the files data set that will be used.

With the exception of the DSNINDEX subfile, the capacity values supplied in FDSAMPLE should be sufficient for a substantial period of time. All of the fields within FILEDEFN are described below:

Subfile Definition Parameters

The subfile definition parameters discussed here are found in dsn=dms.parmlib(FILEDEFN). When you are defining subfiles, remember that the maximum number of subfiles that can be defined is 12.

TABLE 1. FILEDEFN Record Format

Position	Field Name	Description
1-8	Subfile Name	The 8-character name of the subfile being defined
9-11	Logical Record Length	The 3-digit length of the records that will constitute the subfile.

TABLE 1. FILEDEFN Record Format

Position	Field Name	Description
12-14	Key Length	The 3-digit length of the record keys.
15		Character “C” to indicate capacity field follows.
16-23	Capacity (maximum record count) needed	The number of records the subfile must be able to index. This number is the theoretical maximum allowed; for example, if inserts were in the perfect sequence to fill every data block -- in other words, when totally “compressed”, this number of records can be indexed. In the real world, inserts are random, resulting in “block splits” which cause embedded freespace (as much as 50 percent) in any given data block (or possibly all blocks). For this reason, it is recommended that you set the capacity value to roughly twice the number of records you estimate will be inserted.
24	Shared DASD	The letter Y in this column indicates the file can be accessed concurrently by multiple tasks.
25	No Duplicates Allowed	The letter “Y” in this column indicates that the file does not permit entries with duplicate keys.
26	Deferred Write	The letter “Y” in this column indicates that the nature of the processing of this file does not require that each record added be written back immediately. This flag should be specified as “Y” only for file name DASDSPCB. All others should be specified as “N”.
27	File validate	The letter “Y” in this column indicates that information regarding the file structure is to be validated during file reload.

Files Data Set Management

Formatting the Files Data Set

After the FILEDEFN member has been created, the files data set can be created and formatted with the following FILEINIT. During file initialization, the FILEDEFN member entries are read and used to determine how the files data set is to be formatted. These FILEDEFN member entries are accessed only during file initialization, after which all information is carried within the data set itself. A Files Data Set Status report will be created upon completion of the FILEINIT JCL.

```
//FILEINIT EXEC PGM=ADSMI002,PARM=ADSDM100,REGION=512K
//STEPLIB   DD  DISP=SHR,DSN=SAMS.DISK.LOADLIB
//ABNLDUMP  DD  DUMMY
//FILES     DD  DISP=(,CATLG,DELETE),DSN=SAMS.DISK.FILES,
//         UNIT=SYSDA,SPACE=(CYL,10,,CONTIG),
//         DCB=(DSORG=DA)
//MSGPRINT  DD  SYSOUT=A
//PARMLIB   DD  DISP=SHR,DSN=SAMS.DISK.PARMLIB
//SYSPRINT  DD  SYSOUT=A
//SYSUDUMP  DD  SYSOUT=A
```

The subfiles are formatted for optimum performance based upon the physical blocksize of the files data set. The physical blocksize of the files data set is the only value common to all of its subfiles. CA-Disk automatically assigns the optimum blocksize according to the device type being used. (You can override the default BLKSIZE by specifying a value in your JCL, but it is NOT recommended.) The major concern is to specify an appropriate value in the FILEDEFN entries.

Note: Please review the topic "Files Data Set Capacity" on page 20.

Files Data Set Status Report

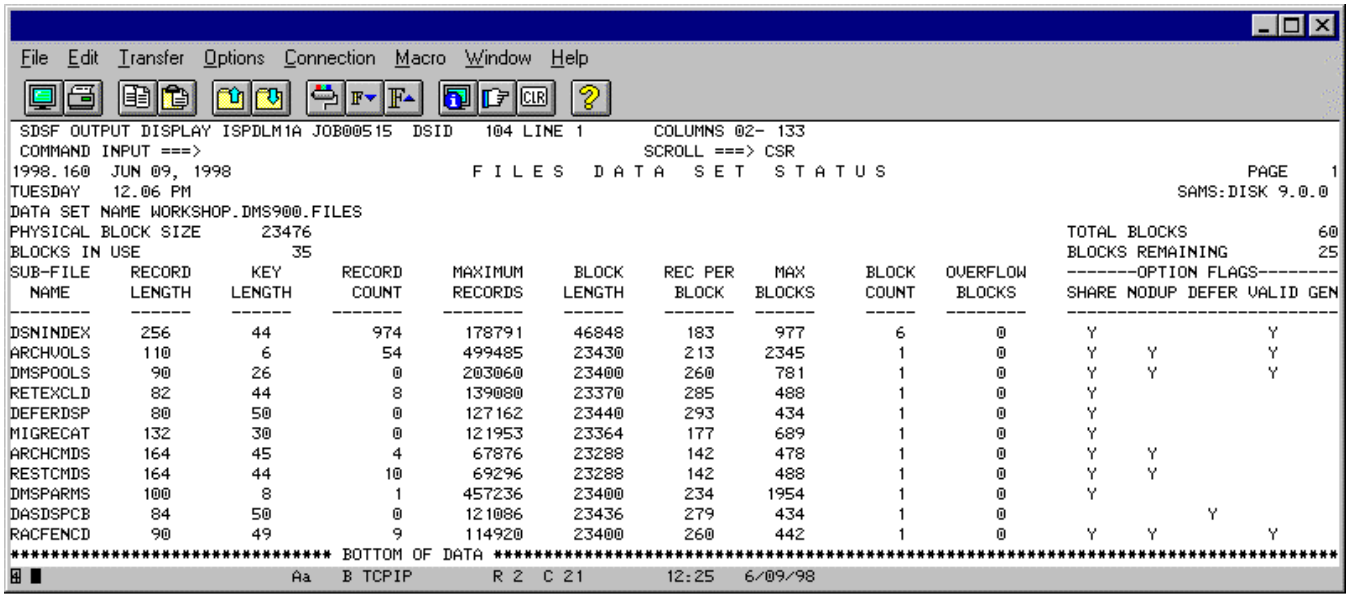
The following JCL can be used to produce this report.

```
//STATUS    EXEC PGM=ADSMI002,PARM=ADSDM107,REGION=512K
//STEPLIB    DD  DISP=SHR,DSN=SAMS.DISK.LOADLIB
//ABNLDUMP   DD  DUMMY
//FILES      DD  DISP=SHR,DSN=SAMS.DISK.FILES
//MSGPRINT   DD  SYSOUT=A
//PARMLIB    DD  DISP=SHR,DSN=SAMS.DISK.PARMLIB
//SYSPRINT   DD  SYSOUT=A
//SYSUDUMP   DD  SYSOUT=A
```

This JCL produces a report that indicates the current usage for each subfile defined to the files data set. A critical field that should be inspected regularly is the OVERFLOW BLOCK COUNT. Whenever overflow blocks exist, the definition entry for each subfile with overflow blocks must have its record capacity increased. The entire files data set should then be reorganized immediately. Failure to do so will result in performance degradation and unpredictable results.

Adding VALIDATE to the PARM field (PARM='ADSDM107,VALIDATE') will cause the subfiles that allow validation to be verified. Files data set valication processing confirms the integrity of index/data address pointers and key sequence within the Files data set. If processing is terminated with a user abend, see "Recovery" on page 28 for instructions of Files data set recovery. Otherwise review the validation messages 0359 on page 38 in the Message Manual for more information.

FIGURE 1. Sample Files Data Set STATUS Report



```
SDSF OUTPUT DISPLAY ISPDLM1A JOB00515  DSID   104 LINE 1       COLUMNS 02- 133
COMMAND INPUT ===>                                           SCROLL ===> CSR
1998.160  JUN 09, 1998                    F I L E S   D A T A   S E T   S T A T U S                                  PAGE    1
TUESDAY   12.06 PM                                                                                                 SAMS:DISK 9.0.0
DATA SET NAME WORKSHOP.DMS900.FILES
PHYSICAL BLOCK SIZE      23476                                                                   TOTAL BLOCKS            60
BLOCKS IN USE               35                                                                   BLOCKS REMAINING        25
SUB-FILE    RECORD      KEY      RECORD     MAXIMUM     BLOCK    REC PER    MAX      BLOCK   OVERFLOW   -------OPTION FLAGS-------
  NAME      LENGTH    LENGTH      COUNT     RECORDS    LENGTH     BLOCK   BLOCKS     COUNT    BLOCKS    SHARE NODUP DEFER VALID GEN
--------    ------    ------    -------    --------    ------    -------  ------     -----   --------   --------------------------
DSNINDEX     256        44         974      178791     46848      183      977        6        0         Y                 Y
ARCHVOLS     110         6          54      499485     23430      213     2345        1        0         Y     Y           Y
DMSPOOLS      90        26           0      203060     23400      260      781        1        0         Y     Y           Y
RETEXCLD      82        44           8      139080     23370      285      488        1        0         Y
DEFERDSP      80        50           0      127162     23440      293      434        1        0         Y
MIGRECAT     132        30           0      121953     23364      177      689        1        0         Y
ARCHCMDS     164        45           4       67876     23288      142      478        1        0         Y     Y
RESTCMDS     164        44          10       69296     23288      142      488        1        0         Y     Y
DMSPARMS     100         8           1      457236     23400      234     1954        1        0         Y
DASDSPCB      84        50           0      121086     23436      279      434        1        0                     Y
RACFENCD      90        49           9      114920     23400      260      442        1        0         Y     Y           Y
****************************** BOTTOM OF DATA ******************************************************************************
                          Aa   B TCPIP        R 2  C 21        12:25   6/09/98
```

Field Descriptions — Status Report

CA-Disk v.r.m - The release number will appear where v.r.m is shown.

Physical block size - The actual block size for the files data set. An optimum value for the device is automatically assigned by CA-Disk unless you specify a BLKSIZE value in the initialization JCL.

Total blocks - The total number of physical blocks within the files data set. Some of these will contain data (blocks in use), and others will be “free blocks”, available for record inserts (blocks remaining).

Blocks in use - When initialized, the files data set is formatted from beginning to end with physical blocks of binary zeros (see **Total blocks**). A minimum of one index block and one data block are then “allocated” to each subfile. All other blocks are “free” (see **Blocks remaining**), and are subsequently allocated only when record inserts require more data blocks. The “blocks in use” number is the total number of physical blocks that have been allocated due to record inserts.

Blocks remaining - The number of physical blocks initially formatted with binary zeros that have not yet had any records inserted. The number of unused, or free blocks in the data set.

Subfile name - The name of either a CA-Disk or user-defined subfile.

Record length - The length of the logical record within the named subfile.

Key length - The length of the key portion within the logical record. For example, the key to the DSNINDEX record is the data set name, which is 44 bytes of the 256-byte record.

Record count - The current number of records contained within the subfile.

Maximum records - The maximum number of records that can be inserted and still indexed. This value is computed at initialization time from the capacity fields you specified in the FILEDEFN member of parmlib. When the actual record count exceeds half of this maximum count, you should consider specifying a higher capacity value in the FILEDEFN entry.

Block length - Records for each subfile are grouped together to form a “logical” block that is also a “best fit” into a physical block (or multiples of physical blocks). Therefore the value displayed should be slightly less than a multiple of the physical block size.

Rec per block - The number of records grouped together to form a “logical” block for the named subfile.

Max blocks - The maximum number of logical data blocks that can be indexed for the subfile. Multiplying (max blocks) (rec per block) results in the maximum number of records that can be inserted and still indexed.

Block count - The number of logical data blocks in use (indexed) for the subfile.

Overflow blocks - When an additional logical data block is allocated to a subfile but there is no room left in the index to keep track of it, it is called an overflow block. Overflow blocks are chained from the relevant data blocks, and will cause considerable overhead or other unpredictable errors. The capacity for the subfile should be increased, and the files reorganized.

Option flags

Share: Y	indicates that integrity is maintained during concurrent access by multiple tasks.
Nodup: Y	indicates that entries with duplicate keys are not allowed.
Defer: Y	indicates that I/O can be buffered. This option requires special processing and is supported for only the DASDSPCB subfile.
Valid: Y	activates internal validation routines when deemed appropriate.

Files Data Set Capacity

Estimating Capacity Requirements

There are two requirements for estimating files data set capacity. The first requirement is estimating the number of records the files data set should contain (FILEDEFN Specifications). The second requirement is understanding how much space (Physical space) should be allocated on the device based on the first requirement.

Maximum Capacity

Assuming that the files data set is 100% DSNINDEX, the maximum capacity on a 3390-9 is 3.4 GB. That equates to:

- 14.2 million records
- 65,536 tracks
- 4,369 cylinders

Depending on what other subfiles are formatted in the files data set, the actual capacity could be much less.

Note: For the files data set and all it's utilities to work properly, the recommended maximum capacity should be kept between 9 and 10 million records.

Estimating Subfile Capacity Requirements

Since the DSNINDEX subfile contains an entry for every data set backed up or archived, a beginning estimate is simply the number of data sets that are to be backed up with CA-Disk. If CA-Disk is to be used to back up all of your packs you need to answer three questions.

1. What is the total number of data sets on these packs?
2. How many versions of each data set do you want to keep?
   - Non-SMS and/or SMS
   - Archives and/or Backups
3. How will my estimates comply to the Index Maintenance policies that manages what remains and/or is deleted from the files data set.

Estimating Physical Space Requirements

$$Total\ number\ of\ tracks\ = \frac{(Subfile\ Capacity\ Requirements)}{(Records\ per\ Track)}$$

Performance Considerations

The files data set EXCP interface is designed to optimize performance at the logical block level. Normally, higher capacity requirements utilize larger blocksizes yet the EXCP interface is designed to access these blocks in a single I/O, without performance penalty.

If your DSNINDEX subfile capacity requirement is substantially less than the maximum capacity requirement for the device type and you are experiencing a high number of block splits and/or performance degradation, use the maximum capacity requirement for the device.

Capacity Considerations

Block Splits

Since the logical block is kept in alphabetical order, but the inserts are in random order, an insert can require splitting a FULL logical block into two half-full blocks before the record can be inserted in its proper sequence.

Assume the worst case, that all blocks have been split and are only half full. If a subsequent series of updates requires inserts to the same block (names happen to be alphabetically adjacent), that block will need to be split also, but it cannot be. The insert will fail because all blocks are “in use”, even though each is only half full. Notice that at this point, inserts are failing when only half of the theoretical capacity has been reached.

If the insert locations just happen to fall alphabetically into the correct blocks the theoretical capacity could be reached without block splits.

Overflow Blocks

If the logical indexing capacity is actually reached before it is noticed, processing will still continue as long as there is physical space left in the data set to allocate the needed data blocks. These non-indexed data blocks are called overflow blocks, which seriously degrade future processing and can cause unpredictable results. Messages are issued whenever this condition occurs. The Files Data Set Status report will indicate the overflow blocks. You should increase the capacity field for the subfile in the FILEDEFN member, and then use the REORG utility on the files data set as soon as possible.

Extending Allocation

When 100,000 bytes or fewer remain in the entire files data set, a message will be issued every time an entry is added. The message states that the files should be enlarged. This is done by unloading the current records

(UNLOAD), and then reloading (RELOAD) them into a data set with a larger space allocation. If these messages go unheeded, an appropriate message is issued and a user abend is issued when the physical space is exhausted.

Please review your policies concerning Index Maintenance. Mis-management of the files data set can impact capacity issues.

Files Data Set Utilities

Unload, Reload, And Reorg

Three utility functions are provided to backup and reorganize the files data set: UNLOAD, RELOAD, and REORG ALL.

- UNLOAD - unloads the files data set to a sequential data set to create a backup.
- RELOAD - reloads the files data set from the sequential data set.
- REORG - reorganizes the files data set by unloading the records to a sequential file, sorting them, and reloading them.

Selection of a subset of the files data set is permissible during both the unload and reload operations. Caution should be exercised, however, when specific subfiles are selected since the ENTIRE files data set is REFORMATTED EVERY time a RELOAD is done (that is, a RELOAD does not insert records at the logical subfile level leaving the other subfiles intact). All subfiles are reformatted and then the unloaded records are reloaded. Care must be taken to ensure that all subfiles with records to be saved are both unloaded and reloaded.

The unload/reload support will normally be used for one of two purposes. One role is to permit the allocation of a larger files data set to accommodate file growth. The new file capacities must be entered in the FILEDEFN member prior to executing the reload program. Since the reload program reformats the entire files data set, destroying the contents of all subfiles, every subfile must first be unloaded.

The other role is to re-order the data blocks and reduce block splits. The files data set itself can be large enough, but growth in one or more subfiles causes an increasing number of block splits which increases the number of I/Os performed.

Unload

JCL — UNLOAD Files

When you UNLOAD ALL subfiles and specify DAYS=(nn), indicating log data set retention, CA-Disk will dynamically allocate the sequential data set(s) into which the files will be unloaded. If it is present in the JCL, the SEQFILES DD is used. If you let CA-Disk dynamically allocate this file, your unload copy of the files data set will be called “files.dsname.current-date.current-time”. If the files data set name exceeds an allowable amount for the date appendage, truncation of the files data set name will take place at position 27 of the files data set name. Note that the enqueue on the files data set is held for the entire UNLOAD process. When the UNLOAD is complete, “B” log

files (review the topic "Naming the Log Data Sets" on page 27 for details) older than the retention period specified by parameter DAYS= will be deleted. When the sequential data set is dynamically allocated, the expiration date used for the data set is taken from sysparm DYNEXPDT.

“I” log files, those created by ISPF (review the topic "Naming the Log Data Sets" on page 27 for details) are not selected for deletion at this time, as they can represent recovery data for more than one files data set. To delete these log data sets, select them using the DSCL CREDT criteria, and then issue the archive or delete action command.

The SIMULATE parameter is provided with the UNLOAD function of CA-Disk. When specified, all reports and messages will be generated, but the actual UNLOAD will not take place until the SIMULATE parameter is removed from the command.

The unloading of the contents of the files data set from disk to tape (a sequential data set on disk works equally well) is accomplished with the following JCL. A DSN=any.gdg(+1) is suggested.

FIGURE 2. Sample UNLOAD JCL and Command

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                        (UNLOAD) - 01.00                 Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000100 //XXXXXXXX JOB (XXXX,XXX),'NNNNNNNN',CLASS=X,MSGCLASS=T,NOTIFY=&USERID
 000200 //UNLD     EXEC PGM=ADSMI000,PARM=ADSDM177,REGION=4096K
 000300 //STEPLIB   DD  DISP=SHR,DSN=WORKSHOP.R90.LOADLIB
 000400 //ABNLDUMP  DD  DUMMY
 000500 //CMDPRINT  DD  SYSOUT=*
 000600 //FILES     DD  DISP=SHR,DSN=SAMS.DISK.FILES
 000700 //MSGPRINT  DD  SYSOUT=*
 000800 //PARMLIB   DD  DISP=SHR,DSN=SAMS.DISK.PARMLIB
 000900 //          DD  DISP=SHR,DSN=WORKSHOP.R90.PARMLIB
 001000 //SYSPRINT  DD  SYSOUT=*
 001100 //SYSUDUMP  DD  SYSOUT=*
 001200 //SEQFILES  DD  DISP=(,CATLG,DELETE),
 001300 //         DSN=SAMS.DISK.SEQFILES,
 001400 //         DCB=(LRECL=268,BLKSIZE=6144,DSORG=PS,RECFM=VB),
 001500 //         SPACE=(6144,(360,360),RLSE),UNIT=SYSALLDA
 001600 //SYSIN     DD  *
 001700  UNLOAD ALL
 001800 /*
 ****** ***************************** Bottom of Data ***************************
 4 ■                        Aa   A TCPIP        R 4  C 15        13:53  11/25/97
```

UNLOAD Command and Parameters

```
UNLOAD ALL,FILES=,SIM,DAYS=,UNIT=,PRI=,SEC=
```

ALL - The presence of the parameter causes ALL subfiles to be unloaded.

FILES= - A list of one or more subfile names can be specified to confine the output to a subset of the files data set. Omission of this OPTIONAL parameter results in all files being unloaded.

SIM - This parameter causes the output reports to be produced as if processing had been performed; however no data sets are unloaded or deleted.

DAYS= - This parameter specifies the retention period for CA-Disk-generated recovery log data sets. Log data sets older than the specified number of days will be deleted following completion of the UNLOAD. CA-Disk will not delete batch log data sets that are less than 1 day old. If DAYS=0 is entered, CA-Disk processes the log data sets using DAYS=1. This prevents CA-Disk from deleting log data sets that are currently being built. No processing against archived log data sets is done.

UNIT= - This is a list of names for the units allocated for the UNLOAD data set when the unload is considered a backup (ALL is specified and the DAYS= parameter is used.) If the unit is a disk device, the esoteric name should be used. If the unit is a tape device, enter TAPE, 3480, DYN1 through DYN9 (DYNxUNIT sysparm values will be used as needed).

PRI= - This is the primary space allocation amount in cylinders for the UNLOAD data set. The default amount is 1 cylinder.

SEC= - This is the secondary space allocation amount in cylinders for the UNLOAD data set. The default amount is 1 cylinder.

RELOAD

JCL — RELOAD Files

The RELOADING of an unloaded set of files to the files data set is accomplished with the following JCL. Care should be taken to ensure that the sequential input file contains all of the needed files, since the reload reformats the entire files data set prior to doing the reload.

FIGURE 3. Sample RELOAD JCL and Command

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                      (RELOAD) - 01.00                 Columns 00001 00072
 Command ===> _                                             Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000100 //XXXXXXXX JOB (XXXX,XXX),'NNNNNNNN',
 000200 //         CLASS=X,
 000300 //         MSGCLASS=T,
 000400 //         NOTIFY=&USERID
 000500 //RELD      EXEC PGM=ADSMI002,PARM=ADSDM192,REGION=4096K
 000600 //STEPLIB    DD  DISP=SHR,DSN=WORKSHOP.R90.LOADLIB
 000700 //ABNLDUMP   DD  DUMMY
 000800 //CMDPRINT   DD  SYSOUT=*
 000900 //FILES      DD  DISP=SHR,DSN=SAMS.DISK.FILES
 001000 //MSGPRINT   DD  SYSOUT=*
 001100 //PARMLIB    DD  DISP=SHR,DSN=SAMS.DISK.PARMLIB
 001200 //           DD  DISP=SHR,DSN=WORKSHOP.R90.PARMLIB
 001300 //SYSPRINT   DD  SYSOUT=*
 001400 //SYSUDUMP   DD  SYSOUT=*
 001500 //SEQFILES   DD  DISP=OLD,DSN=SAMS.DISK.SEQFILES
 001600 //SYSIN      DD  *
 001700  RELOAD ALL,FORMAT
 001800 /*
 ****** ***************************** Bottom of Data ***************************
 4 ▌                     Aa   A TCPIP        R 4  C 15        14:06  11/25/97
```

RELOAD Command and Parameters

```
RELOAD ALL,FORMAT,MODULES=,FILES=,FREESP=
```

ALL - This parameter causes all of the unloaded records to be reloaded.

FILES= - A list of one or more subfile names can be specified to confine the load to a subset of the records on the sequential storage medium. Omission of this parameter results in all records from the sequential data set being loaded, regardless of the subfile from which they came.

FORMAT - This parameter causes the files data set to be entirely reformatted prior to loading the records from the specified subfiles.

MODULES= - A list of one or more names of modules to be given control in order to inspect or alter each record reloaded.

FREESP= - By default, RELOAD processing reloads subfile records into each logical data block, until the maximum record limit is reached. After which a BLOCK SPLIT is required to insert other entries into the logical bock. Specifying this parameter with a percentage value helps alleviate block splits. This parameter is limited to only the DSNINDEX subfile.

For example, a value of 20 allows 20 percent of free space for each logical data block.

REORG Subfiles

If any of the subfiles of the files data set go into overflow blocks and the capacity increase and reorganization are not done (see "Files Data Set Status Report" on page 18), the possibility exists that new inserts or deletes can cause the proper key sequence for some records to be DESTROYED. No records are lost, but the improper key sequencing can cause unpredictable results. If this condition occurs, the reorganization utility can be used to reorder all subfile records into their correct sequence.

```
//REORG   EXEC REORG
//SYSIN    DD  *
 RELOAD ALL,FORMAT
```

Note: The RELOAD command described above can be used to restrict the reload to a subset of the files.

A message is printed during the REORG to indicate when it starts reloading the files. If the system abends after this message, a RESTART procedure has been provided to complete the reload. The RELOAD command is used during the RESTART PROCESSING.

```
//RESTART  EXEC RESTART
//SYSIN     DD  *
 RELOAD ALL,FORMAT
```

Files Data Set Logging

Activation

To activate the files data set logging feature, you must specify sysparm FILOGNAM with the high-level node of the data set name that will be used for the logs. The default value for FILOGNAM is blanks, which disables the logging feature. FILOGNAM must be placed in member SYSPARMS (described in "System Parameter List" on page 491) and must not be overridden in individual job steps. This ensures that all CA-Disk processing affecting each files data set will have a log associated with it. Changing log files attributes (such as blocksize or record length) should be excluded from allocation products processing. Volume redirection within FILEUNIT or adding volumes to avoid a space abend is allowed.

For more information, please review the sysparm "FILOGNAMxxxxxxxxxxxxxxxxxxxxxxxxxx" on page 147.

To direct the size and destination of the files data set logs, see the sysparm descriptions for "FILESPAC1500U" on page 147 and "FILEUNITnnnnnnnn" on page 147.

Naming the Log Data Sets

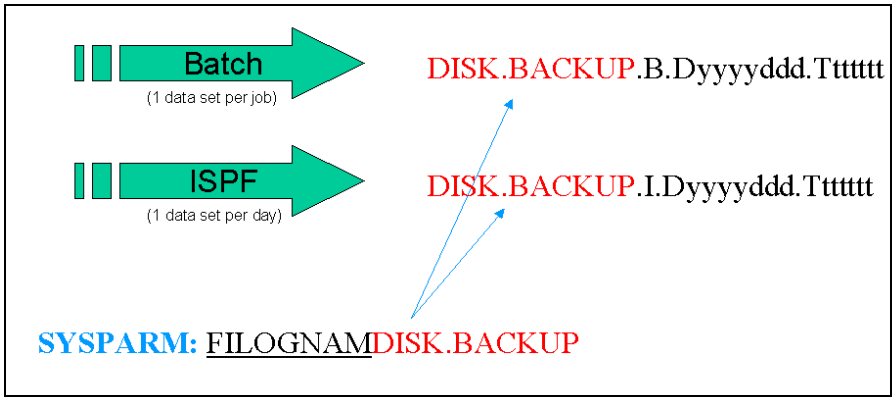
CA-Disk generates data set names for the log data sets as they are created. The high-level node of the data set name is the value you specify for sysparm FILOGNAM. To this node CA-Disk will append a second-level node of I or B, indicating how the log was generated:

I — indicates that the log was generated from an ISPF interactive request, or from a DMSAR auto-restore.

B — indicates that the log was generated from a batch job.

This is followed by a date/time stamp generated during dynamic allocation of the data set. One batch log data set will be created for each CA-Disk job run, but only one ISPF log data set will be created daily.

FIGURE 4. Files Data Set LOGGING Generated DSNAME



Format of the Log Data Sets

In the log data set (Batch or ISPF), each log record is preceded by the files data set identifier record which contains the name of the Files Data Set. This enables CA-Disk to support logging and recovery to multiple files data sets.

Managing the Logs

With the logging function activated, CA-Disk will create and accumulate a large quantity of log data sets over time. The two types of logs (see Figure 4) are managed in two totally different ways.

B (Batch) Logs - The UNLOAD utility can be run with a DAYS= parameter to delete old batch logs, and retain only those logs created more recently than the number of days specified on the parameter. This feature only works with the ALL parameter specified. Running UNLOAD with a list of files specified will cause the processing of the log data sets to be bypassed.

I (ISPF) Logs - These logs represent recovery data for more than one files data set and must be deleted manually. However, you can automate this process using DSCL. For instance, if you backup your files data sets nightly, the following command can be useful in automatically deleting old ISPF logs:

```
//SYSIN    DD  *
FIND DSNAME=DISK.BACKUP.I/
SELECT CRITERIA=(CREDT,LT,TODAY-2)
   DELETE
//SYSPARMS DD  *
FILOGNAMDISK.BACKUP
```

Recovery

CA-Disk provides the ability to recover the files data set should it become lost or damaged. To use the files data set recovery utility, you must first activate the files data set logging feature. The recovery utility uses records from the log data sets as input, and updates the appropriate backup copy of the files data set to achieve the forward recovery.

The FRECOVER command of the files data set recovery utility initiates files data set recovery. The input to FRECOVER is the audit log data sets created during files data set logging. Review the topic "Activation" on page 26 for more information.

In the FRECOVER command, you specify the files data set to be recovered and the audit log data sets to be selected. Audit log data sets are selected based on: time and date parameters, and the high level qualifier name of the audit log data sets in the sysparm FILOGNAM. The audit log records from the audit log data sets selected are used to cause forward recovery of the files data set.

Running the Utility

Before you run this utility in live mode, run in simulate mode to verify the parameters are correct and the correct logs are being selected. When running the FRECOVER utility for lost or damaged files, first initialize a new files data set and RELOAD the current backup copy into the files data set or RELOAD the most current copy of the backup into the damaged files data set, then do the following steps accordingly.

To recover a lost files data set, run the FRECOVER utility with the following:

- Initialize a new files data set and RELOAD it with the most current UNLOAD of the files data set prior to the time of loss.
- Run the FRECOVER utility with the audit log data sets from the Files Data Set Logging starting from the date and time of the UNLOAD used in the first step and ending at the current date and time.

To recover a damaged files data set, run the FRECOVER utility with the following:

- Determine when the time of damage occurred. Rename the audit log data sets to exclude from the recovery process. Save the renamed audit log and other information for future analysis. Review the I log data sets created on the day the damage occurred. I logs are created only once per day (For more information on I logs see "Naming the Log Data Sets" on page 27 of this manual).
- Initialize a new files data set and RELOAD it with the most current UNLOAD of the files data set prior to the time of damage.
- Run the FRECOVER utility with the audit log data sets from the Files Data Set Logging starting from the date and time of the UNLOAD used in the second step and ending at the current date and time.

Internal Processing

The files data set recovery utility first selects the audit log data sets within the time and date parameters specified in the FRECOVER command. The time and date parameters should not proceed the date and time of the last files data set backup. Next, it looks inside these data sets and selects the records that were created for the files data set to recover, and that are within the time and date parameters. If any of the audit log records in a batch audit log data set do not fall within the specified time and date, the data set is bypassed.

The selected audit log records are sorted by the files data set subfile name and the audit log record time and date. The subfile name, subfile function and subfile record (which are the original parameters used to update the files data set) are extracted from the audit log record. These subfile parameters are then used to update the files data set.

Security Considerations

The utility checks the user access to the subfile you want to recover before it updates the subfile, if you have subfile security set up at your installation. However, the user access to any of the data within the subfile is not checked, such as the data set name within the DSNINDEX record. We recommend that you protect the FRECOVER command using the SECURCMD sysparm and the SECURDSN member of parmlib to prevent unauthorized use of this utility.

JCL — Files Data Set Recovery Utility

The following JCL is used for the forward recovery of the files data set.

```
//FRECOVER EXEC FRECOVER
//SYSIN     DD  *
 FRECOVER FILES=filesdsn,BTIME=hhmmss,BDATE=date,ETIME=hhmmss,EDATE=date,SIM
```

FRECOVER Command and Parameters

```
FRECOVER SIMULATE,FILES=,BTIME=,BDATE=,ETIME=,EDATE=
```

SIMULATE - Specify this parameter to select the audit log data sets and audit log records which would be used to update the files data set if running in live mode. A report will be printed.

FILES= - This parameter is required to specify the files data set to be recovered.

BTIME= - This parameter is required to specify the begin time of the audit log data sets and audit log records to be selected for files data set update. It must be in hhmmss format, hh=hour, mm=minutes, and ss=seconds.

BDATE= - This parameter is required to specify the begin date of the audit log data sets and audit log records to be selected for files data set update. This can be in any accepted CA-Disk format.

ETIME= - This parameter is required to specify the end time of the audit log data sets and audit log records to be selected for files data set update. It must be in hhmmss format, hh=hour, mm=minutes, and ss=seconds.

EDATE= - This parameter is required to specify the end date of the audit log data sets and audit log records to be selected for files data set update. This can be in any accepted CA-Disk format.

Sample - Files Data Set Recovery Report

FIGURE 5. Sample FRECOVER Report

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  Display  Filter  View  Print  Options  Help
SDSF OUTPUT DISPLAY SBDLMFRE JOB00119  DSID   104 LINE 99      COLUMNS 02- 133
COMMAND INPUT ===>                                           SCROLL ===> PAGE
1997.315  NOV 11, 1997                    F I L E S   D A T A   S E T   R E C O V E R Y                          PAGE    3
TUESDAY   10.33 PM                                                                                          SAMS:DISK 9.0.0
                                                  AUDIT LOG RECORD UPDATE REPORT
                                        FILES DATA SET =  SBST2.DMS.FILES
SUBFILE   FILES  DMS        SUBFILE                                          NEXT 50 BYTES OF SUBFILE
NAME      FUNC   FUNCTION   KEY                                              STARTING AFTER SUBFILE KEY
--------  ----   --------   ------------------------------------------------ ------------------------------------------------
ARCHCMDS  SR     DISPOSEA   SBST4.OUT.RPT                                    SBST2D  ../..      .    DEVE03
ARCHCMDS  SR     DISPOSEA   SBST4.OUT.RPTM                                   SBST2D  ../..      .    DEVE03
ARCHCMDS  SR     DISPOSEA   SBST4.OUT.RPT2                                   SBST2D  ../..      .    DEVE03
ARCHCMDS  SR     DISPOSEA   SBST4.SPFLOG1.LIST                               SBST2D  ../..      .    DEVE03
ARCHVOLS  SI     BACKUP     @99999                                                 DEVE06......      SBST2.PRIM.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99998                                                 DEVE08......      SBST2.COPY.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99997                                                 DEVE06......      SBST2.PRIM.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99996                                                 DEVE08......      SBST2.COPY.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99995                                                 DEVE06......      SBST2.PRIM.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99994                                                 DEVE08......      SBST2.COPY.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99993                                                 DEVE06......      SBST2.PRIM.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99992                                                 DEVE08......      SBST2.COPY.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99991                                                 DEVE06......      SBST2.PROD.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99990                                                 DEVE08......      SBST2.TEST.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99989                                                 DEVE06......      SBST2.PROD.DMSC.D1997281.T
ARCHVOLS  SI     BACKUP     @99988                                                 DEVE08......      SBST2.TEST.DMSC.D1997281.T
                                Aa   B TCPIP        R 4  C 21        23:12  11/11/97
```

Subfile Dump Utility

JCL — Subfile Dump Utility

A batch hex dump of any record from any subfile within the files data set can be achieved by running the following JCL.

```
//DUMP      EXEC PGM=ADSMI002,PARM=ADSDM613,REGION=1024K
//STEPLIB    DD  DISP=SHR,DSN=SAMS.DISK.LOADLIB
//ABNLDUMP   DD  DUMMY
//CMDPRINT   DD  SYSOUT=*
//FILES      DD  DISP=SHR,DSN=SAMS.DISK.FILES
//MSGPRINT   DD  SYSOUT=*
//PARMLIB    DD  DISP=SHR,DSN=SAMS.DISK.PARMLIB
//SYSPRINT   DD  SYSOUT=*
//SYSUDUMP   DD  SYSOUT=*
//SYSIN      DD  *
FILEDUMP FILE=subfilename,KEY=keyofrecord,VERSION=versionnumber
```

Subfile Dump Command for the Files Data Set

The command to dump one or more of the files is as follows.

```
FILEDUMP FILE=,KEY=,VERSION=
```

FILE= - This parameter is required and must specify any one of the CA-Disk subfiles.

KEY= - This required parameter is used to specify the key of the record desired. Possible values for this parameter include:

TABLE 2. Possible Values for the KEY Parameter of the Subfile Dump Utility

SUBFILE	KEY=	SUBFILE	KEY=
ARCHCMDS	Data set name to be archived	MIGRECAT	First 5 primary volumes
ARCHVOLS	Archive Key	RACFENCD	Data set name
DASDSPCB	Volser + data set name	RESTCMDS	Data set name to be restored
DMSPOOLS	Volser of tape in the pool	RETEXCLD	Data set name
DSNINDEX	Data set name		

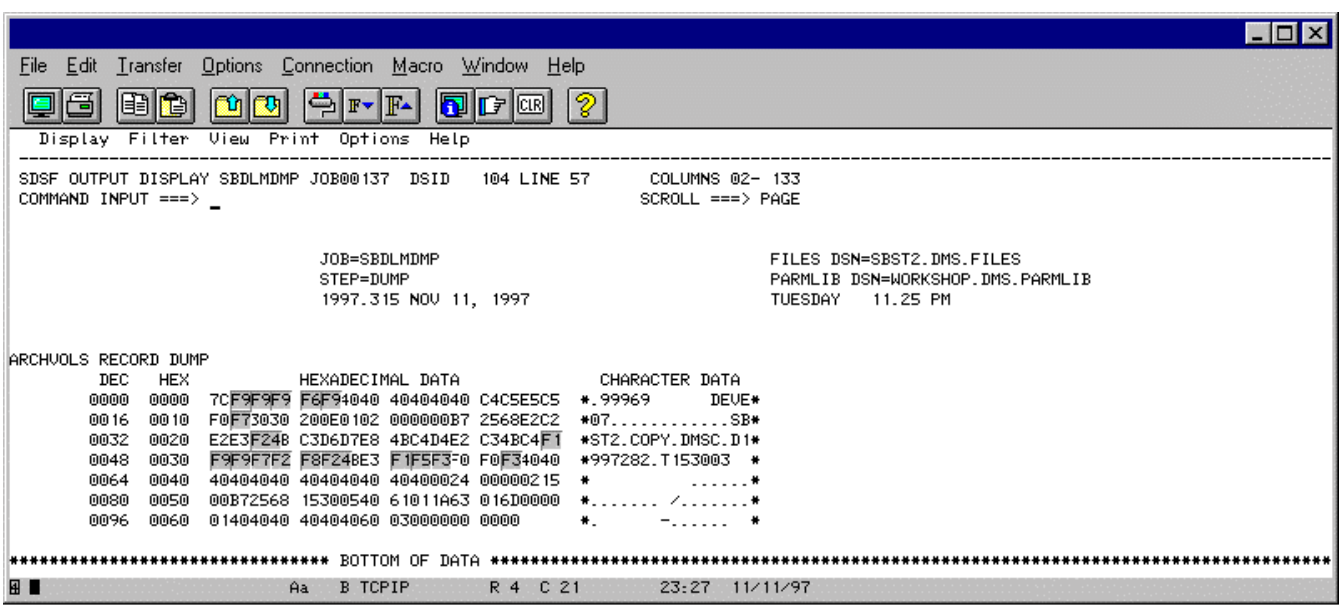
VERSION= - This is an optional parameter, used to specify up to the 99th version of a data set in the DSNINDEX subfile. This parameter is acceptable only for the DSNINDEX subfile. If no version parameter is specified, a default of 0 is used. The version parameter is expressed as a positive number from 0 to 99, with 0 indicating the most current version of the data set.

Sample Subfile Dump Utility

A sample of the command and the output produced by the command can be seen below.

```
FILEDUMP FILE=ARCHVOLS,KEY=@99969
```

FIGURE 6. Sample SUBFILE Dump Utility



Merging Files Data Sets

The previously described RELOAD ALL / REORG procedure is designed to process a single files data set. On some occasions it can be desirable to merge the contents of two or more files data sets, or to perhaps resequence a single files data set from its unloaded backup copy. The following describes the requirements of such an operation.

1. Ensure that all of the ARCHVOLS records being merged have unique keys; that is, the same key does not appear in more than one files data set. Having the same tape in more than one files data set should be extremely rare, but if archiving to disk has been done using more than just one of the files data sets being merged, duplicate keys are probably present. In any case, you must identify all ARCHIVE KEYS that appear in more than one of the files data sets.

   To ensure that no duplicate records exist, unload the ARCHVOLS subfile of each files data set being merged to separate sequential data sets. Sort these sequential data sets by ARCHIVE KEY into a single output data set using the following sort control field:

   ```
   SORT FIELDS=(5,8,CH,A,13,50,CH,A)
   ```

   You can either PURGE (with SCRATCH=NO) any duplicates found from each files data set, or keep one of them and purge the others. In either case, the objective is to arrive at a status where all keys are unique, before proceeding. For syntax, see "PURGE Command and Parameters" on page 327 in the User's Guide.

   Note: Do not use these sorted data sets as input to RELOAD. The purpose of these sorted data sets is to assist you in identifying duplicate ARCHIVE KEYS, and nothing else.

2. After the PURGE process is complete and all keys in all ARCHVOLS subfiles are unique, customize and submit INSTALL member FILESME.

   FILESME has the following steps:

S1UNL11	unloads all subfiles with the exception of DSNINDEX from input files data set #1
S2UNL12	unloads all subfiles with the exception of DSNINDEX from input files data set #2
S3SORT	sorts the temporary data sets from the first two steps and creates a single sequential data set
S4UNL21	unloads and sorts the DSNINDEX subfile from input files data set #1
S5UNL22	unloads and sorts the DSNINDEX subfile from input files data set #2
S6SORT2	sorts the temporary data sets storing the DSNINDEX records and creates a single sequential data set
S7RELOAD	reloads the new files data set using the 2 sorted sequential data sets as input

   Note: The FILESME procedure assumes that both the input files data sets still exist on DASD, and the new output files data set has been preallocated and properly cataloged.

3. Execute the LISTD command and visually verify that the new files data set is accurate and in the correct sequence.

4. Use the REBUILD function to reinsert records for any duplicate ARCHVOLS keys that were purged. The duplicate keys for the disk archives are eliminated because the rebuild function will generate new and unique keys.

Finally, before any LIVE processing is attempted against the new files data set, simulation jobs should be executed, especially IXMAINT.

Common File Handler Interface

Subfile records in the files data set can be accessed by user- written modules to either read or write records in the same manner as CA-Disk modules that interface with the file handler.

Records in the files data set can be referenced both by user exit modules invoked by CA-Disk and/or user programs that are run independently of CA-Disk. It is only necessary that the file handler be called via the MI interface module that is shipped with the system. For best results, all user programs and modules should be compiled using the High Level Assembler. This release of CA-Disk was compiled using version 2.0 of the High Level Assembler, and using Assembler H, especially without slacmods, can cause unpredictable results.

It should be noted, however, that most CA-Disk modules are neither reentrant nor reusable. When programming for interactive applications, more detailed knowledge of program internals will be required to ensure proper execution.

A unique record, identified by supplying the key to the record, can be retrieved from a subfile within the CA-Disk files data set by executing a call in the following manner. If the subfile permits duplicate keys, a request to retrieve a unique record will return the first (most recently added) record for the supplied key.

FIGURE 7. Call to Retrieve DSNINDEX Record

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                                                      Columns 00001 00072
 Command ===> ____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000100          MVC  DSNKEY,=CL44'PAY.MASTER'
 000200 * RETRIEVE THE DATA SET INDEX RECORD
 000300          CALL MI,(ADSDM095,RU,FILENAME,DSNREC,RESULT),VL
 000400          CLC  RESULT,=H1
 000500          BH   NOTFOUND
 000600 *
 000700 ADSDM095 DC   CL8'ADSDM095'  FILE HANDLER CONTROL MODULE
 000800 RU       DC   CL2'RU'        SERVICE TYPE = RETRIEVE UNIQUE
 000900 FILENAME DC   CL8'DSNINDEX'  SUBFILE NAME TO ACCESS
 001000 DSNREC   DS  0CL256
 001100 DSNKEY   DC   CL44' '        KEY OF RECORD TO RETRIEVE
 001200          DC   CL212' '       DATA FROM RETRIEVED INDEX REC
 001300 RESULT   DC   H              RESULT
 ****** ***************************** Bottom of Data ***************************

 4 ▌                    Aa   A TCPIP       R 4  C 15        9:19   3/11/98
```

File Handler Interface

```
CALL MI,(ADSDM095,FUNCTION,FILENAME,KEY&RECAREA,RESULT),VL
```

Where:

ADSDM095 - Is always the 8-character name of the File Interface Control module. ADSDM095 DC CL8'ADSDM095'

FUNCTION - A 2-character mnemonic specifying the type of service to be performed.

FILENAME - The 8-character name of the logical subfile to be accessed. It must be in the file definition entries.

KEY& RECAREA - A work area for record placement, the length of which is specified in the corresponding file definition entry. For unique retrieval of a record, the key of the desired record must be placed in the high-order portion.

RESULT - The halfword return code set by the file handler to reflect the conclusion of the requested service.

Function Service Types

RU - RETRIEVE UNIQUE

This service will return the first occurrence of one or more records having the key that is left-justified in the work area. The length of the key is indicated in the file definition entry.

1. requested record returned
2. key not found

The logical position within the file for the purpose of sequential retrieval when a return code of 2 is passed is the point at which the requested record would have been.

Other than the first occurrence of a given key can be uniquely retrieved by appending to the key in the record area a suffix that consists of a number in parentheses. The number specifies the desired relative occurrence of the designated key where the most recent version is 0.

RN - RETRIEVE NEXT

The next sequential record will be returned by use of this service. It is required that the key for the previous service be present in the record area for maintaining logical position.

To retrieve the first record in a file, it is necessary that the key portion of the record area be initialized with spaces prior to this type of call.

1. next record returned
2. end of file

SI - STORE INSERT

A record is added to the file with the use of this service. Duplicate keys are permitted based on the file definition, with the most recently inserted record being presented first on sequential retrieval.

1. record inserted
2. insert of duplicate keyed record not allowed

SR - STORE REPLACE

A record is updated by use of this service after it has been retrieved. However, a record can be replaced without a preceding retrieval. If you are accessing a subfile that allows duplicate keys, you should position to the correct record before doing the SR function. You can position to a particular record by using the RU function with a relative record number in parentheses or by using the RU function to position to the first record and using the RN function interactively until positioned correctly.

1. record replaced
2. record not found

SD - STORE DELETE

This service is provided for elimination of records from a file.

1. record deleted
2. record not found

There is no requirement for a record to be retrieved prior to a delete call.

CL - CLOSE THE FILES

This service closes the files data set and frees all associated buffers. There are no other parameters for this option. (The other service requests will automatically (re)open the files if needed.)

DW - DEFERRED WRITE BUFFER PURGE

This service applies only to the DASD billing subfile DASDSPCB. Service requests to modify records in this subfile are kept in memory until a different data block is to be updated. Therefore update programs must do a final call using this option to write out the last updated buffer. The record area and result parameters are ignored, if specified, and can be omitted for this service.

For example:

```
CALL MI, (ADSDM095,DW,DASDSPCB),VL
```

Interface for Sequential Reads

CA-Disk also provides an alternative to a series of RN calls to the common file handler interface, which provides much **higher performance** for reading a large number of records sequentially. This high-performance file handler interface keeps each block of records in a buffer until all records have been returned. Once the block has been read into the buffer, this interface will not see any updates done to the block on disk by the ADSDM095 interface.

The intent of this faster interface is for programs that list or scan the files data set, without concern for intermixed updates. The structure of this interface's parameter list is identical to that just presented. However, the interface module is ADSDM463 instead of ADSDM095, and only two function types are valid: RN and RP.

```
CALL MI,(ADSDM463,FUNCTION,FILENAME,KEY&RECAREA,RESULT),VL
```

Where:

ADSDM463 - Is always the 8-character name of the File Interface Control module. ADSDM463 DC CL8'ADSDM463'

FUNCTION - A 2-character mnemonic specifying the type of service to be performed, either RP or RN.

- RP — REPOSITION. Use this service to start a series of RN calls. It resets internal pointers so that you can provide a different key (that is, a new starting location) in subsequent series of read next (RN) calls. Any other parameters specified on an RP call are ignored.

- RN — RETRIEVE NEXT. This service will return the next sequential record. To start a series of RN calls, you must provide the key of the starting location in the KEY&RECAREA parameter of the first call. Any key value, including CA-Disk pattern-matching characters, can be used to establish a starting location. A key of spaces identifies the starting position as the very first record in the file. This initial call will return the first record in the KEY&RECAREA parameter.

  If you do not alter the key at this point, subsequent RN calls will read the following records in ascending key sequence. If you want to start reading another series of records, issue a reposition (RP) call, then provide the key for the new starting position in the KEY&RECAREA parameter in the first RN call in the series. Continue the series with the remaining RN calls.

RC	Meaning
1	next record returned
2	end of file

Processing Support

In the event that an invalid service or file name is passed to the file handler, it will abend the task with a user code of 0016 after printing an appropriate message.

To facilitate the reuse of DASD space within the files data set, a free queue of data blocks is maintained for each subfile. Blocks are assigned to this queue when they contain no more records. Conversely, when a block split occurs during record insertion, the free queue will be inspected before additional space is acquired from the common free area.

When an insert attempt that requires a new data block is made, but both the free queue chain for the subfile and the common area free queue are exhausted, the task will be abended with an appropriate message. Prior to such termination, warning messages will be produced each time an allocation to a subfile occurs when the residual free space falls below the 100,000- byte threshold.

A warning message will appear each time a block split occurs for a file and no capacity remains in the index block. This condition does not render the respective file inoperative, but will result in additional overhead when referencing blocks without index entries. The condition is remedied by reorganizing the file after updating the maximum record count field in the file definition entry for the appropriate subfile.

Support Modules and Functions

ADSDM095 — COMMON FILE HANDLER INTERFACE

This module accomplishes the following:

- Reads in FCB (FCB entries for all subfiles).
- Locates proper FCB entry for requested subfile.
- Identifies requested service type.
- ENQ/RESERVE issued for all non-deferred store/update requests.
- Calls subcontrol modules to perform requested service.
- DEQ/RELEASE issued for all non-deferred store/update requests.

ADSDM463 — INTERFACE FOR SEQUENTIAL READS

This *High Performance* module accomplishes the following:

- ADSDM106 — SUBCONTROL MODULE
- Reads in index block(s) for requested subfile unless
  1. Subfile is using deferred writes (DW)

     or
  2. Sysparm INDXREADN is specified
- Calls appropriate service level module.
- ADSDM087 — SI, STORE INSERT

  Note: This is the only service type that can cause block splits.

```
ADSDM088 ... SR, STORE REPLACE
ADSDM089 ... SD, STORE DELETE
ADSDM096 ... RN, RETRIEVE NEXT
ADSDM097 ... RU, RETRIEVE UNIQUE
```

Service Level Supporting Modules

ADSDM083 — Physical DASD block processor

OPTIONS = open; close; read; write; blksize; dsname

ADSDM081 — Read logical (subfile) block

ADSDM082 — Write logical (subfile) block

ADSDM580 — Locate record by key

Note: This module checks for a deferred write (DW) subfile being processed. If the input key maps to a data block not currently in the buffer and if the current buffer has been updated (write pending flag is on), it purges/writes out the updated block and then reads in the new block needed for the specified key. Since an updated block is held in memory for perhaps several updates, protection against loss of data due to a concurrently running task cannot be provided and the ENQ/RESERVE are not done.

ADSDM086 — Block split handler

Note: A block split is caused by a record insert into a full block. A free block is obtained from the free chain off the subfile's index block or, if none exists there, then from the free area controlled by the allocation control record. The “top half” (high key portion) of the existing block is moved to the “free” block. The record insert is then made to the appropriate upper or lower half-block.

ADSDM084 — Construct keyed file index block

ADSDM079 — Allocate free space control module

ADSDM093 — Validate the file control blocks

This module follows all data block chains and:

- Verifies block count
- Checks for loops in the block pointers
- Checks for logical EOFs
- Checks for ascending keys
- Checks that index key equals block key
- Verifies overflow block chains

CHAPTER 2

Tape Management

This section describes the following topics regarding the management of archive/backup tapes:

- “Allocation Considerations” on page 40
  - “DDNAMES Used” on page 40
  - “Dynamic Allocation Versus Supplied JCL” on page 40
  - “Allocation Support for Masstor's M860 Device” on page 41
  - “Allocation Support for StorageTek's Redwood Device” on page 41
  - “Allocation Support for IBM's Magstar Device” on page 41
  - “Producing Duplicate Archive Tapes Concurrently” on page 42
  - “Specifying Blocksizes for Archive/Backup Tapes” on page 42
  - “Creating Multivolume Tape Data Sets” on page 43
  - “IDRC Support in 3480/90 Cartridge Devices” on page 43
  - “Running Archive/Backup in Simulate Mode” on page 44
  - “Running Multiple Archive/Backup Jobs Concurrently” on page 44
  - “Cataloging the Tape Archives” on page 44
  - “Tape GDG Support (No Longer Recommended)” on page 44
  - “Assigning Tape Expiration Dates” on page 45
- “Tapepool Considerations” on page 45
- “Tape and Cartridge Switching — Length Considerations” on page 49
- “Tape Format” on page 51
- “Archive/Backup Considerations” on page 57

Additional information related to managing tapes can be found in the following:

User's Guide

- “IXMAINT Utility” on page 320 in the User’s Guide
- “Archive Index Rebuild Utility” on page 332 in the User’s Guide
- “MERGE” on page 340 in the User’s Guide

Installation Guide

- “Customizing the CA-Disk Tape Management Support” on page 85 in the Installation Guide

Allocation Considerations

DDNAMES Used

CA-Disk uses the following DDNAMES for tape and cartridge units:

- //ARCHIVE0 — to create the primary archive tape/cartridge
- //ARCHIVEC — to create the duplicate copy tape/cartridge
- //ARCHIVEn — to create the primary merge output tape/cartridge
- //ARCHIVCn — to create the duplicate copy merge tape/cartridge
- //ARCHIVES — to read an archive 3420 tape (i.e., restore or merge)
- //ARCHIVER — to read an archive cartridge (i.e., restore or merge)

Dynamic Allocation Versus Supplied JCL

In the MVS environment, CA-Disk dynamically allocates tape units to the above DDNAMES whenever they are needed, whether for output tapes being created or input tapes to be read. The types of units dynamically allocated are controlled by system parameters ARC0TYPE, MERPnTYP, ARCCTYPE and MERCnTYP. For information on these sysparms, turn to the topic “Specifying the Archive Medium” on page 59.

Supplying the tape dd statements in the JCL will override dynamic allocation. However, dynamic allocation does provide the following advantage. If the job will not actually use the tape units (for example, a simulate run, no data sets are selected to be archived, or the job is limited to reporting functions), the tape units will not be allocated, and unneeded waits are avoided. It also keeps the units free for use by other tasks.

Since dynamic allocation often calls for a different tape unit address for each allocation, you should also review the use of sysparm DYNUKEEP with a value of Y. This causes a tape unit to be allocated when needed, but then kept and reused throughout the remainder of the job step, rather than the system allocating and freeing a tape unit for each tape that is needed. Reusing the same tape unit can benefit the tape operators, but it also increases run time.

Before the next tape can be mounted, CA-Disk is forced to wait for the prior tape to be rewound. Dynamic allocation of a different unit allows some of the processing to be overlapped.

Alternatively, you can also supply the tape dd statements in the JCL (they exist as comment statements in the distributed JCL procedures). This also ensures that the needed tape units are available and allocated before the job starts. If the units are not available when a dynamic allocation request is made, the operator will be prompted for the necessary units.

Note: When supplying the tape dd statements in JCL or sysparms ARC0NAME, ARCCNAME, MERPnNAM, or MERCnNAM, make sure that the value you specify for the data set name does not exceed 22-characters. This will allow CA-Disk to append the required date/time at the end of the ARCHVOL data set. For related information, review Figure 9, “Generated 22 Character DSNAME,” on page 61 and the sysparm description for “ARCTNAMEd” on page 127.

Allocation Support for Masstor's M860 Device

Masstor Systems Corporation's M860 device is sysgen'd as either a 3420 or a 3480 device. The M860, however, treats the cartridge as either the 3420 or 3480 tape device as it was sysgen'd. Just as native 3420 and 3480 devices require different unit names to prevent requests for conventional tapes to be mounted on a 3480 drive, the M860 also requires this separation.

Providing the unit name of the type of device you want is very straightforward when new “tapes” are being created. CA-Disk saves this distinction so that RESTORE, RECOVER, XCOPY, and MERGE can dynamically request the proper type of device. If you use the M860 device, you should review and implement the support technique described in topic “Implementing Support for Masstor M860” on page 81.

Allocation Support for StorageTek's Redwood Device

CA-Disk acknowledges this device in quite the same way it did 3420 and 3480 devices, through an esoteric unitname. Once a unique esoteric is assigned to your Redwood devices, CA-Disk begins storing this name in the ARCHVOL records. This allows CA-Disk to dynamically request the proper device type during RESTORE, RECOVER, XCOPY, or MERGE processing.

For an overview description on the capabilities of the Redwood device, refer to “StorageTek Redwood Device” on page 16 in the User’s Guide.

For detail instructions on how to activate this support, refer to “Implementing Support for StorageTek Redwood” on page 82.

Allocation Support for IBM's Magstar Device

The Magstar device is accessed via dynamic allocation using a user-defined esoteric unitname. Once the esoteric is assigned to your Magstar devices, CA-Disk begins storing the esoteric name in each ARCHVOLS record created on those devices. This allows CA-Disk to dynamically request the proper input device during MERGE, RESTORE, RECOVER, or XCOPY processing.

For an overview description on the capabilities of the Magstar device, refer to “IBM's Magstar 3590 Tape Device” on page 28 in the User’s Guide.

For details on how to activate this support, refer to “Implementing Support for IBM's Magstar” on page 84.

Producing Duplicate Archive Tapes Concurrently

When you run archive or backup jobs, two copies of archive volumes are produced concurrently unless you nullify the allocation of the //ARCHIVEC dd statement. To suppress the duplicate copy, specify sysparm ARCCTYPE with a value of NULL, or provide the dd statement in your JCL as //ARCHIVEC DD DUMMY.

The volume serial number of a duplicate copy tape is carried in the archive volume record for the primary tape. The duplicate copy tape also has an archive volume record built for it. If another copy is made (the triplicate copy), the volume record for the duplicate will point to the triplicate. In each case, the volume record points to the copy record, which points to the next copy, and so on until the end of the chain is reached, indicated by blanks in the volume serial field for the next copy tape.

In the event that a primary archive tape is damaged or otherwise becomes unavailable, the RESET command can be used to flag it as disabled, and cause subsequent RESTORE or MERGE requests to automatically call for the duplicate copy backup tape.

If you produce an archive volume without a duplicate copy, it is possible to create a copy after the fact, just as you can produce the triplicate from the duplicate. See “Making Additional Copies of Archive Tapes” on page 337 in the User’s Guide.

Specifying Blocksizes for Archive/Backup Tapes

The following rules govern the blocksize value used for the archive data sets, whether they are on tape or disk.

1. If archive tapes are being dynamically allocated, a blocksize of 32,760 is used. To use a smaller blocksize, supply the dd statements in your JCL with the blocksize as desired.
2. When archiving to disk, the most efficient blocksize is assigned by CA-Disk based upon the device type being used. You can override this value by specifying sysparm ARCDSKBZ.
3. If either the primary copy or the duplicate copy is on disk, the blocksize for the copy on disk dictates the blocksize used for the tape.
4. The blocksize for the primary copy dictates that of the duplicate copy, if both are on tape.
5. When copying user data sets from disk to the archive medium, CA-Disk will split user data blocks across archive data blocks (similar to the concept of “spanned records”). This means that the blocksize of the archive medium can indeed be less than the blocksize of a data set being archived.

Creating Multivolume Tape Data Sets

If an extremely large data set on disk is to be archived (or backed up), or if very short tapes are mounted, several tapes can be required to hold the data set. CA-Disk permits a data set to span as many as 35 volumes, and automatically allows for this when dynamically allocating the tapes. But if you provide the tape dd statements in your JCL, you must also allow for up to 35 volumes. This is done by adding the volume count parameter, VOL=(,,,35), to your JCL.

IDRC Support in 3480/90 Cartridge Devices

IDRC (Improved Data Recording Capability) is a feature of the 3480/3490 cartridge subsystem. It provides data compression at the cartridge subsystem level. This feature can be controlled when creating output tapes with CA-Disk. Refer to the sysparm description for “IDRCUSEDx” on page 149 for more information. IDRC is activated during OPEN processing and can be used with either JCL or dynamic allocation processing.

When the IDRC feature is not available, two bits in the JFCTRTCH field of the JFCB are defined as zeros. When the IDRC feature is available, the two bits indicate that compaction is on (x'08') or off (x'04'). Just prior to open the bits are checked. If both bits are zero (IDRC not available), the IDRCUSED sysparm is ignored and nothing is done. If the bits are x'08' or x'04' (IDRC available), the IDRCUSED sysparm is checked and if it is specified Y or N the bits are altered to reflect the setting of the sysparm.

When an Archive tape is created with the IDRC feature active, the ARCHVOL record for this tape volume is flagged to indicate so. At restore time, no user action is required because compaction is automatically detected.

Considerations for the use of the IDRC Feature

- Do not use the IDRC Feature for any data targeted for a disaster recovery site unless the recovery site also has IDRC installed. If the recovery site does not have IDRC installed, the data is not recoverable.

- Examine the DCDATACP sysparm and run tests turning CA-Disk compression on and off, both with and without IDRC compaction. You might want to compare both the amount of compaction and the run time difference to determine the optimum settings for these sysparms.

- During duplex processing, if one copy is to a device without the IDRC feature you cannot get an efficient use of your cartridge, since primary and copy cartridges must switch simultaneously.

- If sysparm CARTCALC is specified with a value of 'Y', CA-Disk maintains compression statistics achieved by IDRC when creating ARCHVOLS, and will automatically compensate for it when predicting if a data set is to fit on the active ARCHVOL. However, to prevent CA-Disk from prematurely switching cartridges, CARTFEET needs to be specified with a value of 9999 if the following combination of system parameter settings are present:

  - CARTCALC — specified with a value of 'N'
  - IDRCUSED — specified with a value of 'Y'
  - DCDATACP — specified with a value of 'N'

  A detailed description of sysparm CARTCALC can be found in “CARTCALCn” on page 130.

  For related information, turn to the sysparm descriptions for “DYNUKEEPn” on page 144 and “DYNVREQDn” on page 144.

Running Archive/Backup in Simulate Mode

When you run archive or backup functions in the simulate mode, the tape drives do not need to be allocated. If you are using dynamic allocation options, the tapes will not be allocated. If you are supplying the archive dd statements in your JCL, you can dummy them out for simulate mode.

Running Multiple Archive/Backup Jobs Concurrently

It can be desirable to execute multiple archive or backup jobs concurrently on either the same or different CPUs in order to reduce the elapsed time needed for the process. If these jobs are to be run on the same CPU, or if you have a cross-system enqueue facility in place, the DSNAME parameter on the //ARCHIVEC and //ARCHIVE0 dd statements must be different for each concurrently executing job. This prevents the standard DSNAME enqueue lockout that will occur if the same name is used with DISP=NEW in more than one job.

By default, CA-Disk generates unique names (see "Naming the Archives" on page 61), and dynamically allocates them to these two ddnames, avoiding the enqueue lockout. If you select the option of using non-unique names during backup or archive processing, you must manually supply different DSNAMES for each job by either putting them in the JCL or by overriding sysparms ARC0NAME and ARCCNAME.

Each of your concurrent backup jobs should also process a list of volumes different from the other jobs. It is also advantageous if these volume lists achieve as much channel separation as possible in order to reduce channel contention among the jobs.

Cataloging the Tape Archives

As previously described, if you want to catalog your archive tapes, specify a value of C for sysparm ARCTNAME. If you also use a security system, please heed the authorization requirement documented in "ARCTNAMEd" on page 127.

To uncatalog tapes that are expired by MERGE or IXMAINT, specify sysparm UNCATARC with a value of Y.

Tape GDG Support (No Longer Recommended)

Prior to Release 7.2, users desiring to catalog archive tapes had to use GDG support. Although the GDG support is no longer needed, it can still be used by specifying sysparm ARCTNAME with a value of G. This option is not recommended; it is maintained strictly to provide compatibility with prior releases.

The following paragraphs document how this option works. This explanation is provided strictly for existing users who do not choose to change to the enhanced (default) method of generating tape archive data set names, and as such is not applicable to new users.

GDG support can be invoked by CA-Disk to generate unique tape archive data set names. This is done by appending the GDG name to the base name specified in sysparms ARC0NAME (for the primary copy) and ARCCNAME (for the duplicate copy). This generation name is also kept synchronized with the archive volume index. Two special sysparms are provided for this support:

1. ARCTNAMEG must be specified to synchronize the creation of index entries and catalog entries.
2. UNCATARCY is used to cause a tape whose index entry has been deleted in MERGE or IXMAINT to be uncataloged.

If ARCTNAMEG is specified, a generation index for the appropriate number of entries must first be built into the supporting catalog. Since CA-Disk creates a generation number for each tape created, the number of entries is equal to the maximum number of archive tapes at any given time. Once this generation index level is established, it must be specified as the data set name on all of the dd statements without an appended level number or absolute generation number. CA-Disk will build the correct level number and catalog the tape. The JCL disposition parameter for the archive tape(s) should always be coded as DISP=(NEW,KEEP,KEEP).

If a duplicate copy tape is being produced, a generation index must also be built for it. The default index is DMS.ARCHCOPY, but you can use any name you desire in sysparm ARCCNAME for dynamic allocation or supply it in the JCL.

Assigning Tape Expiration Dates

Expiration dates can be assigned to archive tapes either explicitly or implicitly. An explicit expiration date is created by hard-coding its value in sysparm DYNEXPDT for dynamic allocation or by specifying the EXPDT= or RETPD= parameter in the JCL for the //ARCHIVE0 dd statement.

CA-Disk will honor any non-zero expiration date and place it in both the tape label and in the archive volume index record. Hard-coding an expiration date, however, does create the possibility of expiring archive tape volumes prior to the expiration of all archived data sets on the archive volume.

We recommend using an explicit expiration date of “99365” on your archive tapes. This is considered a “never scratch” date by CA-Disk and protects against the previously mentioned case of a tape being written over before all of its data sets have expired. This is the technique called for by the distributed JCL procedures and system parameter default.

Caution: When using retention period via sysparm DYNEXPDTR, the value after the R is added to the current date to create the expiration date. When the value is greater than or equal to the date of 2155.365, problems can occur in some tape management systems when the tape is opened prior to midnight and closed after midnight. If 2155.365 is exceeded, message “ADSTH001 9999 EXPIRATION DATE EXCEEDED MAXIMUM (2155.365). MAXIMUM EXPIRATION DATE USED” will be issued.

For related information, please see “Year 2000 Considerations” on page 6 in the User’s Guide.

If you have implemented tape management support, it is recommended that all archive tape dd statements use the default EXPDT=99365. CA-Disk will notify the tape management interface as soon as any given tape can be returned to scratch status. See “Customizing the CA-Disk Tape Management Support” on page 85 in the Installation Guide for more information.

Tapepool Considerations

The CA-Disk tapepool support provides you with the ability to define up to 30 different pools of tapes. (You can define 30 pools in parmlib member POOLDEFS. However, this member is also used to define disk archive pools. The sum of tape and disk pool definitions cannot exceed 30.) Each pool name is associated with (mapped to) a data set name on an archive dd statement. Thus, if all archive tapes have the same name or prefix, one pool is

sufficient. However, if the duplicate copy tapes have different names than the primary tapes (as in the default), it is possible to define a separate pool of tapes for each. Similarly, backup tapes can be separated from archive tapes by giving them different names and associated pools.

The association (or mapping) of pool names to archive data set names is made by entries in member POOLDEFS in the parmlib data set. Each entry consists of an eight-character pool name (of your choice) and the tape data set name. (Remember, the data set names are generated from sysparms ARC0NAME and ARCCNAME or are provided in the JCL.) Each entry must be enclosed in quotes, with one or more blanks between the pool name and data set name. For example:

`'ARCHPRIM DMS.ARCHPRIM/'`	pool for primary archives
`'ARCHCOPY DMS.ARCHCOPY/'`	pool for duplicate archives
or perhaps	
`'ARCHPOOL DMS.ARCH/'`	pool for all archive tapes
`'BKUPPOOL DMS.BACK/'`	pool for all backup tapes

The first step in establishing tapepool support is to decide how many different data set names (categories of tapes) you have a real need for. One suggestion is to separate your archive from your backup tapes, and if you create duplicate copy tapes, to separate them from the primary tapes as well. Once decided, make sure that the names you have chosen are provided to CA-Disk via overriding sysparms for each job, or in the JCL for each job. Then create member POOLDEFS in your parmlib data set with the appropriate entries. You can copy in member SAMPDEFS as a starting point.

The next step is to place scratch tapes into the pools in preparation for use. This can be done most easily from the ISPF Online Tapepool Management facility. If you do not have this support, you must use the utility documented in “Tapepool Update Utility” on page 48. The ISPF facility and the ADD Command (described in “ADD Command” on page 48) also provide the means to do any maintenance functions that can be required, such as listing, adding or deleting entries.

At this point you are ready to activate the support. This is done by adding sysparm TAPEPOOLy to member SYSPARMS in the parmlib data set. Once this is done, new scratch tapes to be used for archive, backup or merge functions will be located in the appropriate pool, and their status updated to show that they are in-use. When subsequent merge or index maintenance functions determine that no unexpired data sets remain on a tape volume, its status is updated to indicate that it is available again for use as a scratch tape.

When a group of tapes are entered into a tapepool, they are entered at the bottom of the current list of scratch tapes for the named pool. Tapes that are scratched by MERGE or IXMAINT processing are also put at the bottom of the tapepool scratch list, in the order in which they are scratched. Scratch tapes needed by ARCHIVE, RETAIN, or MERGE are taken from the top of the scratch list. This ensures that the tapes are cycled, much like tape drives are cycled. Due to the complex logic that MERGE and IXMAINT use to scratch CA-Disk archive tapes, the tapepool scratch list quickly loses its initial VOLSER order. This causes no problems for CA-Disk, however. The tapepool LIST command described below can be used to inspect the status of the tapepools.

Tapepool support is also designed to maintain the integrity of the pools in the following manner:

1. The same volume cannot be placed into more than one pool.
2. Only the ISPF or batch utility program will add a volume into a pool; that is, scratch tapes mounted by operators or expired tapes being “returned to scratch status” by index maintenance are never added to a pool. The volume must already exist in the pool such that CA-Disk merely updates its status.
3. When creating new output tapes with the pool support active, a valid pool name to use must be found or processing terminates.

Tape volumes can be supplied by any combination of the following three techniques. It should be noted that these methods are progressive, in that processing begins with the first that is applicable and continues until either no more tapes are needed or the operator is unable to supply an acceptable volume serial.

From JCL:

If tapepool support is being used, from 1 to 5 volumes can be listed in the VOL=SER= parameter of the //ARCHIVE0 and //ARCHIVEC dd statements. These volumes are used before accessing the defined tapepool.

For non-tapepool support, up to five volumes can be supplied on the VOL=SER= parameter of the //ARCHIVE0 and //ARCHIVEC dd statements. If more than five volumes are required, the operator will be requested to mount scratch tapes.

From a tapepool:

If the tapepool support is being used, volume serial entries are taken from the pool when the VOL=SER= parameter is not specified in the JCL, or when the JCL list of volume serials has been exhausted.

From the operators:

If the tapepool support is being used but the pool is out of scratch tapes, a WTOR message is issued to allow the console operator to enter the volume serial of the next scratch tape to be used.

It is permissible to CANCEL an archive run when a volume serial cannot be supplied in response to the WTOR message, since the processing for any volume written up to that point is complete.

Tapepool Updates from ISPF

The Online Tapepool Management facility supplied with the CA-Disk ISPF support provides the easiest means of listing and maintaining your tapepools. In addition to being able to add or delete entries, sysparm TPOOLCHG is available to allow an “I” to be placed to the left of an entry to change it from “scratch” to “in-use”, or an “A” to change an “in-use” status to “available” (scratch).

These options should be used with extreme caution, however, since the tapepool must also be kept synchronized with the current ARCHVOLS listing of “in-use” tapes. For related information, please review “Repair Archvols (REPARCH) Utility” on page 289.

Tapepool Update Utility

If the Online Tapepool Management facility supplied with the CA-Disk ISPF support is not available, the following batch utility provides an easy means of listing and maintaining your tapepools. The available commands and there functions are:

- LIST - generate a volume status list for a requested pool.
- ADD - add one or more volumes to a pool.
- DELETE - delete a specific volume from a pool
- ERASE - remove all volumes from a specified pool.

```
//TAPEPOOL EXEC DMSPOOL
//SYSIN     DD  *
   LIST POOLNAME=,VOLUME=,STATUS=
   ADD POOLNAME=,VOLUME=,NUMBER=,STATUS=
   DELETE VOLUME=
   ERASE POOLNAME=
```

LIST Command

```
LIST POOLNAME=,VOLUME=,STATUS=
```

POOLNAME= - Specify one or more eight-character pool names. (A maximum of 25 entries supported.)

VOLUME= - This optional parameter specifies one or more volume serials in the tape pool. The status report is restricted to just the named volumes. (A maximum of 25 entries supported.)

STATUS= - Specify “S” if you want to list only scratch volumes, or “U” for in-use volumes. The default is to list both scratch and in-use volumes.

ADD Command

```
ADD POOLNAME=,VOLUME=,NUMBER=,STATUS=
```

POOLNAME= - Specify the eight-character name of the tape pool into which volumes are to be added.

VOLUME= - Specify the volume serial to be added to the named pool.

NUMBER= - Specify a value of from 1 to 32767 for this optional parameter to indicate the number of volumes to be added, beginning with the volume specified. The numeric portion of the volume serial is incremented by one, the new volume serial is added to the pool, and the process repeated until the requested number of tapes has been inserted. For example, to add volume serials DMS001 through DMS025, specify:

```
ADD POOLNAME=TAPEPOOL,VOLUME=DMS001,NUMBER=25,STATUS=S
```

STATUS= - This optional parameter specifies the status of the volume(s) to be added. Use “S” for scratch volumes and “U” for volumes in-use by CA-Disk. It defaults to S, and normally you should never have a need to insert in-use tapes.

DELETE Command

```
DELETE VOLUME=
```

VOLUME= - The specified volume will be deleted regardless of the pool it is in.

ERASE Command

```
ERASE POOLNAME=
```

POOLNAME= - The eight-character pool name specified will have all of its volumes deleted. The pool ceases to exist.

Tape and Cartridge Switching — Length Considerations

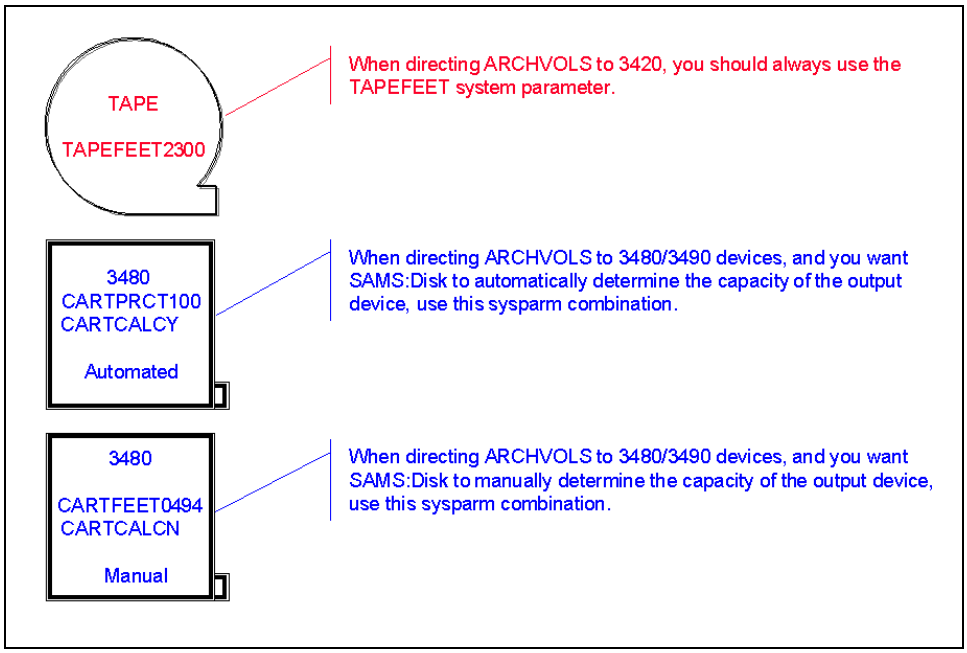
CA-Disk was designed to create single volume ARCHVOLS whenever possible. Before a data set is written to an ARCHVOL, CA-Disk attempts to predict whether or not the data set will fit on the current ARCHVOL. If the prediction indicates that the data set won't fit, the ARCHVOL is closed and a new one is opened. This design minimizes the need to mount more than one tape during subsequent RESTORE operations.

Note: The only exception is when the initial data set on an ARCHVOL exceeds the capacity of the ARCHVOL. Only then will CA-Disk create a multi-volume ARCHVOL, spanning up to 35 volumes if necessary.

The method in which CA-Disk calculates its predictions is determined by the output media used, and the values associated with four (4) system parameters. Also factored into the calculation is whether or not software and/or hardware compression is being utilized.

The following figure summarizes the media options available (with the exception of DISK), and which system parameters are in effect:

FIGURE 8. Predicting Output Device Capabilities



For TAPE (3420) Devices

When determining if the next data set will fit onto the current ARCHVOL, CA-Disk factors in the value specified for TAPEFEET, along with a number of other variables such as:

- Bytes
- Blocks
- Block Size
- Recording Density
- Size of Inter-Record Gaps

The result is then added to the number of bytes and blocks previously written to the ARCHVOL. If the data set will fit, CA-Disk begins transferring the data. If the data set will not fit, CA-Disk closes the ARCHVOL and begins transferring the data onto a new ARCHVOL.

For related information, see “TAPEFEETnnnn” on page 183.

For Cartridge (3480/3490) Devices

To determine if the next data set will fit onto the current ARCHVOL, CA-Disk uses one of two basic methods for cartridge devices:

1. Specifying a value of 'Y' to the CARTCALC system parameter instructs CA-Disk to automatically ascertain the capacity of the cartridge up front. This results in Utilization Statistics that assist CA-Disk in predicting whether or not a data set will fit onto the current ARCHVOL.
2. Specifying a value of 'N' to CARTCALC instructs CA-Disk to manually calculate cartridge capacities. CA-Disk factors in the value specified for CARTFEET, along with a number of other variables such as:
   - Bytes
   - Blocks
   - Block Size
   - Density (fixed value set at 38000 BPI)
   - Size of Inter-Record Gaps
   - IDRC reduction ratio
   - Hardware Compressed using DSS interface

First CA-Disk determines if the data set is small as defined by the IDRCSIZE sysparm. All small data sets are assumed to fit on the tape. If not small, data set byte count uncompressed by either IDRC or software (see “DCDATACPn” on page 136) is added to the number of bytes previously written to the ARCHVOL. If the data set will fit, CA-Disk begins transferring the data. If the data set will not fit, it is tested for Hardware Compressed.

Normally Hardware Compressed data sets using the DSS interface do not see any compression for IDRC or software. If it is, then CA-Disk closes the ARCHVOL and begins transferring the data onto a new ARCHVOL. If it is not, CA-Disk uses the previous compression ratio from the DSNINDEX record (see “IDRCRATE00” on page 148) to “compress” the data set and re-adds. If the data set will fit, CA-Disk begins transferring the data. If it still does not fit, CA-Disk checks for software compression and re-adds based on that. Again, if the data set will fit, CA-Disk begins transferring the data. If the data set will not fit, then CA-Disk closes the ARCHVOL and begins transferring the data onto a new ARCHVOL

For a list of special considerations and a detailed description of both the CARTCALC and CARTFEET system parameters, see “CARTCALCn” on page 130.

Tape Format

The data archived is stored on standard labeled tapes as a sequential data set in undefined record format. The specific CA-Disk record types are presented in the topic “Archive Tape Format” on page 468.

CHAPTER 3

Tailoring Options

General Information

CA-Disk provides two primary means of tailoring the system to more specific needs. You should become familiar with the capabilities each provides so that you can find relatively simple solutions to conditions that are unique to your installation.

1. CA-Disk provides for system parameters (sysparms) to be specified in member=SYSPARMS of the parmlib data set. The contents are retrieved at execution time and processing paths are varied accordingly. Default values for all sysparms are generated internally. You need to create member=SYSPARMS and specify a sysparm value only if you want to change a defined default. See the Table 24, “System Parameters,” on page 107 for a list of the sysparms available. There is no need to read and review every one of them until an actual need is made evident from your use of the various functions. Some of the more commonly used ones, or ones that are more important to consider in advance, are presented here to acquaint you with their use.
2. User exits are also supported. See the Table 45, “User Exits,” on page 203 for a complete list of the exits available.

In addition to sysparms and user exits, the following topics are highlighted in this section:

- “Performance Tips” on page 55
- “Archive/Backup Considerations” on page 57
  - “Activating Data Compression” on page 57
  - “Specifying the Archive Medium” on page 59
  - “Using Multiple Types of Media” on page 60
  - “Naming the Archives” on page 61
  - “JCL Overrides and DSNAME Usage” on page 62
  - “Assigning an Expiration Date to the Archive Data Set” on page 62

- “Archiving to Disk: Requirements and Recommendations” on page 62
- “Using Archive Functions to Manage Active Data” on page 71
- “Archiving the Archives is Not Allowed” on page 72
- “Archive Integrity and Security” on page 73
  - “Archive Tape Protection” on page 73
  - “Checks Against Overwriting Valid Tapes” on page 73
  - “Tape Status After Abends” on page 74
  - “Archive Disk Protection” on page 75
- “Processing PDSs that Contain Anomalies” on page 75
- “Implementing Support for Masstor M860” on page 81
- “Implementing Support for StorageTek Redwood” on page 82
- “Implementing Support for IBM's Magstar” on page 84
- “ISPF Custom Reports” on page 85
- “User-Specified Condition Codes” on page 89
- “DASD Threshold Manager” on page 91

Suggested System Parameters

System Parameters for Reports:

The Data Set Utilization report can be produced in three different sequences. By default, it is produced in sequence by date-of-last-use. The following options are available:

TABLE 3. DSUTIL Report Sorting Options

Sysparm	Description
DSUTILSQ	In data set name sequence
DSUTILCF	By date-of-last-use within index

Both the explicit and the implicit archiving functions can produce reports in one to three sequences simultaneously. By default they are produced in both data set name and data set name within volume sequences. See sysparm “ARCHSORTudv” on page 126 to select a different option.

The sequential migration to tape function has this same ability. See sysparm “MIGRSORTndv” on page 155 to adjust its defaults.

The SMF report requires its input from a sequential data set. If your SMF records reside in a VSAM cluster, add an IBM IDCAMS utility step to REPRO the records into a sequential data set to be passed to CA-Disk.

System Parameters for Messages

The following sysparms affect the number of informational messages that are printed for their respective functions, and should be kept in mind when initially testing and setting up CA-Disk runs.

TABLE 4. Message Suppression Sysparms

Sysparm	Description
DSCLMSGS	DSCL Data Sdata setet Selection Messages
RLSEDIAG	Idle Space Release messages
MIGBYPAS	Sequential Migration to Tape messages
VCBYPASS	Move/Copy messages

System Parameter for Lines Per Page

The number of lines per page on CA-Disk-generated output can easily be changed from its default of 58. For details, see the sysparm description for “RPTLINES” on page 172.

System Parameter for Separator Pages

The number of separator pages can be changed or suppressed using sysparm “FLYCOUNT2” on page 147.

User Exit for All Sysout Print Lines

For more extensive monitoring or modifications of printed output, such as special routing for certain messages, see user exit “SYSOUTEX — SYSOUT Exit” on page 247.

System Parameter for Files Data Set Logging

Three sysparms provide support for the files data set logging capability.

TABLE 5. Files Data Set LOGGING Activation Sysparms

Sysparm	Description
FILOGNAM	for specifying the name of the log data sets and to activate the feature.
FILEUNIT	for specifying the unit type allocated for the log data sets.
FILESPAC	for specifying the number of blocks allocated to the log data sets.

System Parameters for Archive/Backup

The following sysparms are commonly specified and should be reviewed for applicability in your installation:

TABLE 6. Command ARCHIVE/BACKUP Sysparms

Sysparm	Description
ARCBLKSI	for processing data sets that have BLKSIZE=0
ARCDSORG	for processing data sets with unknown DSORGs
ARCEMPTY	for processing empty data sets
ARCMODEL	for processing model DSCBs
ARCRETPD	change retention period for explicit archives
RETRETPD	change retention period for implicit archives

System Parameters for Restore/Recover

By default, CA-Disk issues a catalog locate for each data set being explicitly restored, unless a specific target volume is given. For volume recovery functions, these catalog locates are bypassed. Another locate is done in preparing the restore report. Both restore and recovery tasks normally skips any preallocated version of a data set. Sysparms to review in order to change these defaults are:

TABLE 7. Common RESTORE/RECOVER Sysparms

Sysparm	Description
RESCHCAT	for volume selection for non-VSAM during Restore
RECCHCAT	for volume selection for non-VSAM during Recover
PREALLOC	controls whether or not to overwrite preallocated data sets

Performance Tips

Report Processing

See the sysparms for the reporting function listed earlier that control the use of the LOCATE macro. This macro causes a high level of system overhead. Elapsed time values can be reduced significantly if the locate function is suppressed for runs that process a high number of data sets.

PDS Compress

Sysparm IOTRACKS also affects PDS directory processing. Tests have also shown that for directories in the range of one track or less (which is 46 directory blocks or less on a 3380), setting IOTRACKS to 1 can improve performance. If tests verify this in your environment, supply this value as a permanent sysparm override to the PDS Compress function.

Previous recommendations to exclude SMP/E data sets still being kept as PDSs (rather than VSAM data sets) are

no longer applicable. The revisions allow all partitioned data sets to be processed equally well.

Files Data Set

Because the files data set contains the archive indexes, it is an active and critical data set when doing implicit archiving, backup, or any index maintenance function. ENQ and RESERVE macros are also issued against the data set and its containing volume. Improperly selecting the containing volume can result in performance degradation of CA-Disk or other jobs trying to access the volume.

For important information, see the sysparm description for “RSUPPRESn” on page 172.

I/O Buffering and Memory Requirements

A sysparm that can have a dramatic effect on CA-Disk performance and the amount of memory that is required is IOTRACKS. IOTRACKS controls the maximum number of tracks read or written at one time by the CA-Disk EXCP access method, which is used for all but exception case handling of non-VSAM data sets. See the sysparm description for “IOTRACKSxx” on page 150.

Ideally, optimum performance is achieved when IOTRACKS allows a cylinder to be processed by a single I/O request; for example, 15 tracks for 3380 devices. The default value for IOTRACKS accomplishes this. Test different values of this sysparm to find the best to suit your environment. As with most performance issues, there can be several factors to consider, such as the memory constraints, I/O contention, and page and swap activity.

First consider the size of these buffers themselves. A 15-track buffer on a 3380 is (15)*(47,476) or 712,140 bytes. CA-Disk double buffers, which means 1,424,280 bytes (or roughly 1.4 megabytes). For the Move/Copy function, which is reading and writing to disk at the same time, this means 2.8 megabytes merely for basic I/O data buffers. Memory is required for many other items as well, such as space for the loaded programs themselves and buffers for reading VTOCs, PDS directories, and control information from the CA-Disk parameter library. Many of these memory tables reside above the 16 MB line. A five- to six-megabyte region is usually sufficient for all CA-Disk jobs, including Move/Copy. If real memory is very limited, however, this region requirement can be too high, and must be lowered by reducing the value of sysparm IOTRACKS.

Buffers are acquired based upon the values specified, but a single I/O request never exceeds a cylinder in size or crosses a cylinder boundary. If the number of tracks in a data set (or technically the number of tracks in each extent) is less than a cylinder in size, part of the buffer always remains unused (wasted). It follows that if the average data set size in tracks is less than the value specified for the appropriate sysparm, you can set the sysparm to a lower value to reduce memory constraints without noticeably increasing the number of I/Os (which degrades performance).

If the amount of available real memory is limited, having this sysparm set to a high value can cause jobs to be swapped frequently and/or paging rates to be high, which degrades performance instead of improves it.

If the “to” and “from” volumes are on the same channel when using the Move/Copy function to relocate data sets, the job can perform better by setting IOTRACKS to a much smaller value, perhaps even 1. This is due to the inherent channel contention in this particular case, combined with the paging and swapping problem described above. Again, testing within your environment demonstrates whether this case warrants special treatment.

Another sysparm to see when considering memory requirements is “IOMAXRECnnnnnn” on page 150. This sysparm sets the maximum allowable record size for data sets to be processed by CA-Disk. It is used to acquire buffer space to contain the records during backup and data compression/decompression. The higher this value is set, the greater the memory requirements. Set this sysparm to the lowest value possible that still allows CA-Disk to process. The default value is 65,000.

Archive/Backup Considerations

CA-Disk permits numerous tailoring options for archiving and backup functions. Some of the more important options to consider are discussed in this section.

Activating Data Compression

As an optional feature, CA-Disk can compress data as it is being written to the archive or backup data sets, whether they are on tape or disk.

If you activate data compression, the default technique is technique number 0. CA-Disk-supplied technique number 0 typically reduces the space used by 40 to 50 percent, which yields the same reduction in the amount of tape used. However, if you archive to disk instead of tape, the disk savings achieved can be closer to three or four to one, or even much higher. This is due to the optimized storage techniques in the archives, as well as wasted space in the original data sets.

CA-Disk users who are also licensed to use Sterling Software's CA-Compress data compression product, can specify technique number 1 to use CA-Compress as the compression method. CA-Disk supplies a formal interface to use this compression method. Instructions for implementation are listed in “Using CA-Compress Within The Archives” on page 58.

As a direct result of compression, the number of I/Os to the archive media is also reduced by the same percentage, which can provide a significant improvement if I/O contention is a problem in your installation.

To achieve these benefits, however, expect the CPU time to increase. For more information, see the Table 184, “Data Compression Techniques available,” on page 387.

The following sysparms are used by data compression. See them for applicability in your installation:

TABLE 8. Data Compression Sysparms

Sysparm	Description
DCDATACP	turns data compression off or on
DCCOMPTC	indicates the compression technique number
DCCMPEXn	exit for compression technique n
DCDCPEXn	exit for decompression technique n
DCDSNDEX	user exit for deciding if data compression is to be done for each data set, and what technique is to be used
DCEXCTBL	exclusion table name
DCINCTBL	inclusion table name (overrides DCDATACP”N”)
DCRTSTAT	turns compression statistics off, on, or on with details for each data set compressed.

Using CA-Compress Within The Archives

CA-Disk users who are licensed to use Sterling Software's CA-Compress data compression product can use CA-Compress to compress the CA-Disk archives. If CA-Compress is installed on your system, follow the following steps to implement this compression method.

1. Create a CA-Compress File Description Table (FDT) to use when archiving/backing up data. To do this, create a CA-Disk backup tape (NOT a disk data set) from a normal CA-Disk backup run, but without using data compression (that is, sysparm DCDATACP specified with a value of N). Use this backup tape as input to the CA-Compress File Prepass Utility. For specific instructions on creating the FDT, see the CA-Compress for MVS User's Guide.
2. Convert the FDT to load module format using the CA-Compress FDTLOADR Utility. Give the FDT an 8-character name (“DMSOSFDT” is recommended) and place it in the CA-Disk load library. For specific instructions on using the CA-Compress FDTLOADR Utility, review the CA-Compress for MVS User's Guide.
3. Specify sysparm SHRNKFDT with the value of the FDT name. Note: The default value for this sysparm is “DMSOSFDT”. If you have specified this name as the name of the CA-Compress FDT in the load library (step 2 above), it is not necessary to specify this sysparm.

   If you specified a different name, provide your name as the value for this sysparm in member SYSPARMS in the CA-Disk parmlib.)
4. Link the CA-Compress interface with CA-Disk using the JCL below. Note: If you install a new version of CA-Compress, you must relink this interface. The JCL and control statements are provided in member RELKSHRK of the installation library.

```
//JOBNAME  JOB (ACCT INFO)
//* ****************************************************************
//* * RELINK SAMS:DISK MODULES FOR SHRINK COMPRESSION              *
//* ****************************************************************
//S1LKED   EXEC PGM=IEWL,PARM=(LIST,LET),REGION=3072K
//SYSPRINT  DD  SYSOUT=*
//SYSLMOD   DD  DISP=SHR,DSN=SAMS.DISK.LOADLIB
//SYSLIB    DD  DISP=SHR,DSN=SAMS.DISK.ADMSLOAD
//SHRINK    DD  DISP=SHR,DSN=SAMS.COMPRESS.LOADLIB
//SYSUT1    DD  UNIT=SYSDA,SPACE=(1024,(50,20))
//SYSLIN    DD  *
 INCLUDE SYSLIB(ADSUT410)
 INCLUDE SHRINK(SHRKEXPD)
 ENTRY ADSUT410
 NAME ADSUT410(R)
 INCLUDE SYSLIB(ADSUT411)
 INCLUDE SHRINK(SHRKEXPD)
 ENTRY ADSUT411
 NAME ADSUT411(R)
/*
```

5. Concantenate the CA-Compress loadlib to the CA-Disk loadlib within the following PROCS:

DMS	DMSAR
FMS	RECOVER
RESTORE	

   This ensures the CA-Compress utilities are available to CA-Disk, if needed.

6. Exempt all CA-Compress-controlled data sets from compression during archival or backup processing. This can be accomplished either by placing the data set names or patterns in a data compression exclusion table as a parmlib member and specifying sysparm DCEXCTBL with the name of the parmlib member, or by using the CA-Disk user exit DCDSNDEX.

   The reason for excluding these data sets is that files already compressed are not compressed any further when archived or backed up. Excluding them from compression during archive or backup processing decreases CPU overhead and improve performance.

7. Specify the following required sysparms:

TABLE 9. Required Data Compression Sysparms

Sysparm	Description
DCDATACP	activate data compression
DCCOMPTC1	use CA-Compress as the data compression method for the CA-Disk archives

8. Optionally, you can specify the following sysparms:

TABLE 10. Optional Data Compression Sysparms

Sysparm	Description
SHRNKFDT	to specify the FDT name if other than the default of “DMSOSFDT”
DCDSNDEX	8 character user exit for deciding if data compression is to be done for each data set, and what technique to be used
DCEXCTBL	exclusion table name (cannot be used with DCINCTBL)
DCINCTBL	inclusion table name (cannot be used with DCEXCTBL)
DCRTSTAT	turns compression statistics off, on, or on with details for each data set compressed

Note: You cannot change or delete the FDT once you have compressed CA-Disk archive files using CA-Compress.

Specifying the Archive Medium

By default, CA-Disk dynamically allocates both the primary and duplicate copy to tape. The DD statement used for the primary is //ARCHIVE0 and for the duplicate copy is //ARCHIVEC. Sysparms ARC0TYPE and ARCCTYPE allow you to change the defaults from tape to either disk or 3480 cartridge units. Specify these sysparms with a value of DISK, TAPE or 3480, and place them in the SYSPARMS member of the parmlib data set. There are 2 ways to indicate that no duplicate copy is to be made:

1. Specify “NULL” to sysparm ARCCTYPE as “NULL”
2. In JCL, specify “DUMMY” to the //ARCHIVEC DD statement

For example, to direct the primary copy to disk and the duplicate copy to a 3480 cartridge unit, specify ARC0TYPEDISK and ARCCTYPE3480.

You can also specify the archive medium through JCL. If tape or 3480 devices are specified via JCL, the devices allocated at job initiation are used throughout the run; that is, the device are not deallocated at the end of each tape, but only at the end of the job. The dynamic allocation option can be made to behave in much the same way by specifying sysparm DYNUKEEP. The only exception is when more than five volumes are needed to contain a single archive data set, in which case dynamic allocation is used.

You can also specify archival to disk in JCL by allocating any disk device to the //ARCHIVE0 DD statement. For example:

```
//ARCHIVE0 DD UNIT=SYSDA,VOL=SER=anyvol,DISP=SHR
```

Notice that this only indicates to CA-Disk to use disk archive. The volume (or data set) named on the DD statement has no meaning to CA-Disk, and dynamic allocation of the ARCHVOL is still done.

Using Multiple Types of Media

This section describes how CA-Disk can utilize multiple types of media (i.e., 3480s, 3490Es, SILOs, etc.), sysparms to use, IDRC considerations, data compression, and MERGE processing.

Selecting the Output Media

The user selects the media CA-Disk writes to by specifying a value to the following sysparms, which are often entered as overrides in CA-Disk jobs, or kept in the SYSPARMS member in PARMLIB.

TABLE 11. Output Media Sysparms

Function	Primary Sysparm	Copy Sysparm	Notes
ARCHIVE	ARC0TYPE	ARCCTYPE	
BACKUP	ARC0TYPE	ARCCTYPE	
MERGE	MERPnTYP	MERCnTYP	For details, refer to “LIMITS=” on page 349 in the User’s Guide

The following table shows the media options that can be specified with the above sysparms. The primary and its copy can be directed to different media. These two output media types are synchronized so that if one is shorter than the other and fills up, both are closed and new tapes are mounted and/or new disk archive data sets are allocated. Other sysparms that are in effect when the respective media option is used are shown below:

TABLE 12. Available Values for Media Option Sysparms

Media Option	Type of Device	Unit for Writing or Reading	Unit for Auto-Restore	Tape Length
TAPE	“Round” tape (default)	DYNTUNITtape	ARESUNITtape	TAPEFEET2300
3480	3480/3490 cartridge	DYNCUNIT3480	ARESUNIC3480	CARTFEET0494
DYN1	Special device type 1	DYN1UNIT	ARESUNI1	TAPEFEET2300
DYN2	Special device type 2	DYN2UNIT	ARESUNI2	TAPEFEET2300
DYN3	Special device type 3	DYN3UNIT	ARESUNI3	TAPEFEET2300
DYN4	Special device type 4	DYN4UNIT	ARESUNI4	TAPEFEET2300
DYN5	Special device type 5	DYN5UNIT	ARESUNI5	TAPEFEET2300
DYN6	Special device type 6	DYN6UNIT	ARESUNI6	TAPEFEET2300
DYN7	Special device type 7	DYN7UNIT	ARESUNI7	TAPEFEET2300
DYN8	Special device type 8	DYN8UNIT	ARESUNI8	TAPEFEET2300
DYN9	Special device type 9	DYN9UNIT	ARESUNI9	TAPEFEET2300
DISK	DASD volumes	n/a	n/a	n/a
NULL	Disables Copy sysparms	n/a	n/a	n/a

Specifying the ESOTERIC Unit Name

The media options shown above relate to sysparms that provide the esoteric unit name CA-Disk uses to allocate the proper device for writing, reading and auto restoring. These sysparms are placed in the SYSPARMS member in PARMLIB and should not be changed unless the esoteric name changes, or a media type has been totally removed from CA-Disk control. The DYNn sysparm provides the flexibility of defining additional device types to CA-Disk that can be installed at various sites.

When a tape is created, CA-Disk stores the device type in the ARCHVOLS record along with a flag if the DYN1, through DYN9 option was used. With this information, CA-Disk retrieves the esoteric unit name from the DYN_UNIT or ARESUNI_ sysparm so the proper device can be allocated.

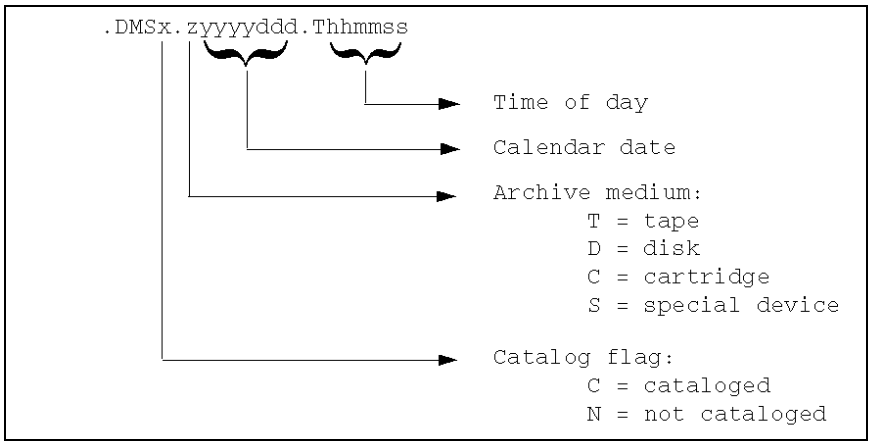
Naming the Archives

In theory, the names you assign to archive data sets are of no consequence to CA-Disk. CA-Disk keeps track of the archive data sets regardless of their names. However, other issues dictate that naming conventions be observed, including the IBM operating system and tape management concerns. CA-Disk generates unique archive data set names, providing flexibility in managing the archives and allowing them to be cataloged. Unique data set names also eliminates contention during dynamic allocation that would occur if non-unique data sets were used.

CA-Disk sysparm defaults generate unique archive data set names. The base names are taken from either sysparm ARC0NAME or ARCHIVE0 DD statement (for the primary copy) and sysparm ARCCNAME or ARCHIVEC DD statement (for the duplicate copy). If both copies are being produced, the base names should be different — such as DMS.ARCHPRIM and DMS.ARCHCOPY.

Specify up to 22 characters for the base name and CA-Disk generates a 22-character appendage as follows:

FIGURE 9. Generated 22 Character DSNAME



The calendar date above is the current date in Julian format, and the time of day is in hours/minutes/seconds. Therefore, the archive data set name provides the date and time it was created, and is always unique.

For disk archive data sets, this generated name is always used and is always cataloged. For tape and cartridge archive data sets, the default is to use the generated name but not to catalog it. You can tell CA-Disk to catalog the tape or cartridge archive data set name by specifying a value of C for sysparm ARCTNAME.

JCL Overrides and DSNAME Usage

As just explained, CA-Disk defaults to using dynamic allocation for the archive data sets. For tape and 3480 devices, however, you can override dynamic allocation by supplying the needed information on the appropriate DD statements in the JCL. CA-Disk creates the data set name from the DSNAME you provide in the JCL unless you omitted it completely or specified a temporary name (DSN=&&anyname). In both of these cases, CA-Disk uses the appropriate sysparm (ARC0NAME, ARCCNAME, MERPnNAME or MERCnNAME) to create the data set name.

Assigning an Expiration Date to the Archive Data Set

By default, CA-Disk assigns the “never expire” date of 99365 to each of the archive volumes. This guarantees that an archive volume is never expired before all of the data sets that it contains. Therefore your only concern is the retention period being assigned to each data set being archived. This is recommended if you are using the EDM interface of your Tape Management System.

When CA-Disk determines that all data sets on an archive volume have expired, it automatically expires the volume as well. You can, however, specify a different expiration date through dynamic allocation with sysparm DYNEXPDT, or in JCL through the LABEL= parameter. Append either Eyyddd or Rddddd, where E indicates a Julian date follows and R indicates a 5-digit retention period follows.

For related information, please see “Year 2000 Considerations” on page 6 in the User’s Guide and “Customizing the CA-Disk Tape Management Support” on page 85 in the Installation Guide.

Archiving to Disk: Requirements and Recommendations

Traditionally, backup and archive copies of data sets have been directed to tape. CA-Disk also provides the option of writing backup and archive data sets to disk. Using disk devices rather than tape or cartridge units for archive data sets eliminates the need for an operator to mount a tape when the backup or archive task is performed or when restoring data. This is particularly attractive for users implementing the auto-restore capability of CA-Disk.

Note: When archiving to disk, you should consider activating software compression to save on DASD space. To activate software compression, refer to the sysparm description of “DCDATACPn” on page 136.

The following summary outlines the steps necessary to implement archival to disk. Where necessary, an expanded explanation is provided for some of the summary items.

1. Specify archival to disk by one of the following methods:
   - If your ARCHIVE and DMS JCL procedures do not contain //ARCHIVE0 or //ARCHIVEC DD statements, specify sysparm ARC0TYPE with a value of DISK for the primary, and/or ARCCTYPE with a value of DISK for the copy, to allocate them dynamically.
   - Supply JCL statements for the ddnames above that causes allocation to any disk device. CA-Disk deallocates and then allocate the device as needed to perform the archive. For example, the following DD statement activates archival to disk for the primary copy:
     ```
     //ARCHIVE0 DD UNIT=3380,VOL=SER=anyvol,DISP=SHR
     ```
2. Specify the base name for the disk archive data sets using the following sysparms. CA-Disk appends a unique name to the end of the base name, based on the current date and time.
   ```
   ARC0NAME – for the name of the primary
   ARCCNAME – for the name of the copy
   ```

3. Map the names you gave above to proper diskpools (members in your parmlib data set) from which target volumes are selected. This mapping is specified with entries in member POOLDEFS of the parmlib data set. The general form of the entries is:

   ```
   'VOLU3380 DMS/'
   ```

4. Place target volumes in your defined pools by creating the parmlib member by the same name as your pool, in this case “VOLU3380”, and inserting statements such as:

   ```
   VOL=(vol001,vol002,vol/)
   ```

5. Specify an expiration date using sysparm DYNEXPDT, or by JCL. CA-Disk assigns the default expiration date of 99365 if none is specified.

   For related information, please see “Year 2000 Considerations” on page 6 in the User’s Guide.

6. Determine if the default values blocksize is appropriate. If not, override the default with the sysparm ARCDSKBZ.

7. Determine if the default manner of calculating primary and secondary space allocation is appropriate. If not, override the default by specifying sysparms:

   ```
   SPACEPRImmm – megabytes to allocate for primary
   SPACESECmmm – megabytes to allocate for secondary
   ```

8. Determine if default values for index maintenance, merge and rebuild functions are appropriate, and if not, specify sysparm to override the defaults.

9. Exclude Disk ARCHVOLS from CA-Allocate EOV processing or any other product designed to add a new DASD volume to an allocation to avoid x37 type abends.

Step 1 — Specifying Archival to Disk

CA-Disk is distributed with default sysparms that cause archive and backup copies of data sets to be written to tape, creating both a primary and a duplicate copy tape concurrently. You can tell CA-Disk to write archive and backup copies of archive data sets to disk instead. You can write both the primary and duplicate copies to disk, or assign them to different media. You can also “dummy out” the duplicate copy. Indicate your choice of archive media either through JCL or CA-Disk sysparms ARC0TYPE and ARCCTYPE.

Step 2 — Naming the Disk Archive Data Set

For further discussion of this step, see “Naming the Archives” on page 61.

Step 3 — Map Data Set Names to Diskpools

Candidate volumes for archiving to disk must be provided via “diskpools”, which reside as members in the parmlib data set. The name of the diskpool is simply the member name. CA-Disk determines which diskpool (member) to use in the same manner in which tapepool names are determined. That is, the POOLDEFS member of parmlib is used to associate (map) a data set name to a poolname. For example:

FIGURE 10. Sample Source of POOLDEFS Member in PARMLIB

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT          .DMS90.PARMLIB(POOLDEFS) - 01.02          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'ARCPOOL0 DMS.ARCHPRIM/'
 000002 'ARCPOOLC DMS.ARCHCOPY/'
 000003 'BKPPOOL0 DMS.BKUPPRIM/'
 000004 'BKPPOOLC DMS.BKUPCOPY/'
 000005 'ARCPOOL0 SBDLM./'
 ****** ***************************** Bottom of Data ***************************

 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT          .DMS90.PARMLIB(ARCPOOL0) - 01.01          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 VOL=(VOL001,VOL002,VOL003,VOL004,VOL005,VOL006,VOL007,VOL008,VOL009)
 000002 VOL=(VOL010,VOL011)
 000003 VOL=(VOLA/,VOLB/,VOLC/,VOLD/,VOLE/,VOLF/,VOLG/)
 000004 VOL=(/)
 ****** ***************************** Bottom of Data ***************************

 4 ■                        Aa   A TCPIP        R 17 C 15        15:08   3/11/98
```

Step 4 — Place Target Volumes in Defined Pools

Next, designate your candidate volumes in diskpool=ARCPOOL0 by creating a member in the parmlib data set with that name and inserting entries. For an example, examine the lower portion of Figure 10 above.

Either explicit volumes or volume patterns can be used. However, the total number of volumes in the pool cannot exceed 300. Do not try to continue the volume list on multiple lines (that is, continuation lines are not supported). Instead, enter multiple VOL= statements.

If both the primary and the duplicate copy are being directed to disk, each pool must contain volumes of the same device type(s) as the other pool; that is, if pool 1 has both 3380s and 3350s, pool 2 must also have some 3380s and 3350s. This is required because a volume from pool 1 is selected first, and then pool 2 is searched for a volume of the same device type.

By default, CA-Disk automatically separates the copy from the primary; that is, if a volume appears in both pools, CA-Disk does not select the same volume for the copy as is used for the primary. Obviously, this is to prevent the loss of both copies if the disk pack is destroyed, whether it be due to oxidation, head crashes, or other reasons. To allow both copies to go to the same volume, specify sysparm ARCSEPC0 with a value of N.

Step 5 — Specifying an Expiration Date

For further information about this step, see “Assigning an Expiration Date to the Archive Data Set” on page 62.

Step 6 — Specifying Blocksize

When archiving to disk, CA-Disk automatically adjusts the optimal blocksize value for the device type of the volume; that is, half track blocking on 3375s and 3380s, and full track blocking on other devices. You can override the defaults by specifying sysparm ARCDSKBZ. The blocksize selected for a disk device is also used for any tape or 3480 copy that is being created concurrently.

If you have more than one device type in your archive pool, use sysparm ARCDSKBZ to specify a blocksize appropriate for all devices; that is, one that provides optimal performance and space usage on all the devices.

Step 7 — Specifying Space Allocation

CA-Disk determines the amount of space to allocate (the amount of space it needs on a diskpool volume) in the following manner. It assumes that quite frequently the primary copy is on disk and the duplicate is on tape or a 3480 cartridge. As stated earlier, the target volume leads CA-Disk to the device type, which in turn dictates the blocksize to be used. This, coupled with the density and length of tape being used, allows an easy and very accurate calculation of the tape capacity. (Since the ends of 3480 tapes cannot be clipped off, the most common variable in the calculation has been removed!)

The optimal disk allocation is the exact same capacity as its backup medium, such that when one is filled and both must be closed, nothing is wasted on the other medium. To do this on an exact equivalence, however, would require very large amounts of free space to be available. Therefore CA-Disk selects 1/16th of that value as default primary and secondary space allocations. In many cases, an archive run does not need a full tape, and one or two extents are more than enough space. However, when a large archive run is made and the diskpool volumes contain plenty of free space, CA-Disk obtains as many as possible of the 16 extents before closing the archive data set. This maintains high tape usage as well. Any excess (unused) disk space is, of course, released immediately when the data set is closed.

CA-Disk also considers the size of the data sets being archived when determining how much space to allocate. If the first data set to be archived is larger than the default primary space, the primary space is reset to the size of the data set. If free space equal to this new primary value is not available but the default value is, CA-Disk attempts to archive the input data set to a multivolume output data set. This is done only as a last resort and you should not use a Storage Class that is defined with guaranteed space since unused space is only released in the active volume at CLOSE time (there is a possibility that all preallocated volumes would not be used because the data is written in compressed format).

Once the target archive data set is allocated and the first data set to be archived is copied to it, CA-Disk continues to copy additional data sets (the second through nth) as long as there is sufficient space to hold them. To determine whether sufficient space is available, CA-Disk examines the unused space in the current extent, and checks to see if additional extents can be obtained. If there is not sufficient space, the current disk archive data set is closed and a new one is allocated (like swapping to a new tape).

If your analysis or practical experience indicates that the default method of calculating space allocations should be changed, specify sysparms SPACEPRI and SPACESEC with a three-digit number representing the number of megabytes of disk space to allocate for each.

When archiving to disk, you normally want the archive data sets to be large, as restoring from a large data set is as fast -- or faster -- than restoring from several small data sets. CA-Disk provides options for balancing the need to conserve storage space with the need to keep overhead to a minimum. If you set the primary space value (sysparm SPACEPRI) to the size you want to use for staging, then set the secondary space value (sysparm SPACESEC) to 000, CA-Disk creates smaller archive data sets, but more of them.

Step 8 — Exclude Disk ARCHVOLS from CA-Allocate EOV processing

Disk ARCHVOLS should be excluded from CA-Allocate EOV processing or any other product designed to add a new dasd volume to an allocation to avoid x37 type abends.

Step 9 — IXMAINT, MERGE and REBUILD

To accommodate archival to disk, the following options are available within the functions listed below.

- IXMAINT

  Parameter SCRATCH=YES/NO is available on the DSNDELETE, VOLDELETE, and PURGE commands. DSNDELETE and VOLDELETE default to SCRATCH=YES, which means to scratch and uncatalog the disk archive data set when the archvols record is being deleted. PURGE defaults to SCRATCH=NO, because it is frequently used just prior to running REBUILD. If a disk archive data set is scratched, there is nothing from which to rebuild. If you do not intend to follow PURGE with a REBUILD command, specify SCRATCH=YES. Otherwise you are creating “disconnected” archive data sets on disk that are never used or deleted by any other means.

- MERGE

  The following parameters have been added.

  TYPE=(DISK,TAPE,3480) — specify any of the three types listed to restrict the merge to archvols of those types.

  CREDT=date — only those archvols created on or before the specified date are merged forward.

  DAYSOLD=dddd — an “alternative form” of the CREDT parameter that limits the merge to those archvols that were created dddd or more days ago.

- REBUILD

  Rebuild accepts the archive data set to be read as input from either tape or disk. The supplied JCL procedure requires that you provide the correct //ARCHIVES DD statement pointing at the tape or disk data set to be rebuilt. No input cards are required. If they are provided, only the expiration date parameter is used.

Creating and Using Dual Primary Volumes

Differences of Primary/Copy Volumes vs. Dual Primary Volumes

The typical CA-Disk backup/archive process creates duplicate outputs on tape or disk referred to as the primary and copy archive volumes (or, ARCHVOLS).

An optional process that can be utilized during backup/archive is to create Dual Primary Volumes. Using this option, the backup/archive process creates PRIMARY1 and PRIMARY2 (P1 and P2) volumes on tape or disk.

Here are some distinctions between these two processes:

	Primary/Copy	Dual Primary Volumes
1	The primary and copy volumes contain exactly the same data.	The two primary volumes contain exactly the same data.
2	The primary/copy volume relationship does not change for the life of the volumes.	The P1 and P2 volumes are not connected or related in any way.
3	For each data set processed, one DSNINDEX record is created.	For each data set processed, a P1 DSNINDEX record and a P2 DSNINDEX record is created.
4	A primary ARCHVOLS and a copy ARCHVOLS record are created for the output volumes.	A P1 ARCHVOLS record is created for the P1 output volume and a P2 ARCHVOLS record is created for the P2 output volume.
5	The DSNINDEX record contains the volser (or key) of the primary ARCHVOLS record.	The P1 DSNINDEX record contains the volser (or key) of the P1 ARCHVOLS record and, the P2 DSNINDEX record contains the volser (or key) of the P2 ARCHVOLS record.
6	The primary ARCHVOLS record contains the key of the copy ARCHVOLS record.	The P1 ARCHVOLS record has no field pointing to the P2 ARCHVOLS record, nor does the P2 ARCHVOLS record point to the P1 ARCHVOLS record.
7	The primary and copy archvol can <u>not</u> be managed separately.	The P1 DSNINDEX and P1 ARCHVOLS records can be managed independently of the corresponding P2 records.
8	The copy ARCHVOL is kept until the primary ARCHVOL expires, at which time the copy ARCHVOL is also expired.	The P2 DSNINDEX and P2 ARCHVOLS records can expire before, after or at the same time as the P1 records.

Important Considerations of the Dual Primary Volume Process

In an execution of CA-Disk, the option to create Dual Primary Volumes can be utilized or Primary/Copy volumes can be created. These are mutually exclusive processes in the job step being executed. The

Copy Utility can be used to create copies of the Dual Primary Volumes.

- All of the P1 and P2 DSNINDEX and ARCHVOLS records are maintained in the FILES data set, along with the entries for the primary/copy ARCHVOLS and DSNINDEX records.
- The DSNINDEX records and the primary ARCHVOLS records created in the primary/copy process are considered as P1 records (e.g., the “P2” indicator flag is *off*).
- There is no distinction - and no method to distinguish - between primary DSNINDEX records and ARCHVOLS created in the primary/copy process and the *“P1”* DSNINDEX records and ARCHVOLS created in the Dual Primary Volume process.

Although the Dual Primary Volume feature provides very flexible options in managing dual output images and volumes, the user must be aware that there could be a significant increase in the number of DSNINDEX records maintained in the FILES data set. Before using this feature, these factors must be considered:

- Ensure that the physical size (number of cylinders allocated) of the FILES data set is adequate to contain the expected increase of DSNINDEX records.
- Ensure that the record (or, indexing) capacity for the DSNINDEX subfile is adequate (specified in member FILEDEFN in the CA-Disk parmlib).
- The runtime of IXMAINT can be longer because of the additional records created by this feature.

Dual Primary Volume Feature Overview

The following are the functions where the P1 and/or P2 records can be selected or viewed:

- **RESTORE/RECOVER** option selects P1 and/or P2 DSNINDEX records.
- **IXMAINT** option selects P1 and/or P2 records for maintenance processing.
- **MERGE** option selects P1 or P2 volumes.
- **LISTD** option displays both P1 and P2 DSNINDEX records.
- **LISTD** and **LISTV** shows the setting of the P2 flag field.
- **REBUILD** option restores the P2 indicator for the DSNINDEX and ARCHVOLS records when the P2 information is available.
- Merging of FILES Data Sets includes the P2 indicator for the sort sequence.

Except for the functions mentioned above, all other CA-Disk functions operate without consideration for the P2 record indicator.

Dual Primary Volume Details by Function

Archive/Backup

Dual Primary Volumes are created when specifying the following statement in the DSCL:

```
SET  PRIMARY2
```

When SET PRIMARY2 is specified, a P1 and P2 DSNINDEX record is created for each data set processed, and a P1 and P2 ARCHVOLS record is created for each respective output volume.

The Primary2 Volume data set name and unit information are taken from the ARCCNAME and ARCCTYPE sysparms.

Different expiration dates can be applied to the P1 and P2 DSNINDEX and ARCHVOLS records. The P1 DSNINDEX and ARCHVOLS records can have expiration dates that are greater than, less than or equal to the P2 records expiration dates.

- Parameters RETPD and EXPDT apply to the P1 DSNINDEX records, except as noted below.
- Parameters **RETPD2** and **EXPDT2** apply to the P2 DSNINDEX records as they are created.
- If P2 DSNINDEX records are being created and RETPD2 or EXPDT2 is not specified, the expiration date is taken from the RETPD or EXPDT parm, if specified.
- If P2 DSNINDEX records are being created and neither RETPD/EXPDT or RETPD2/EXPDT2 parameter is specified, the retention period for both the P1 and P2 DSNINDEX records is taken from sysparm RETRETPD.

For SMS data sets, if the Management Class is present and not overridden (e.g., sysparm SMSMCBYP is set to N or B), the RETPD/EXPDT and RETPD2/EXPDT2 parameters are ignored and the expiration date of the P1 and P2 DSNINDEX records is set to 99365. Those P1 and P2 DSNINDEX records continue to be managed using DFSMS and can not be managed separately.

The P1 and P2 ARCHVOLS records can be assigned different expiration dates:

- Sysparm DYNEXPDT applies to the P1 ARCHVOLS records, except as noted below.
- Sysparm DYNEXPD2 applies to the P2 ARCHVOLS records as they are created.
- If P2 ARCHVOLS records are being created and DYNEXPD2 is not overridden in the SYSPARMS member or in the job, the expiration date for both the P1 and P2 ARCHVOLS records is taken from sysparm DYNEXPDT.

Here is an example of using the new parameters and the results:

```
SET PRIMARY2
SCAN REALVOLS
BACKUP RETPD=30,RETPD2=15
```

Sysparms:

```
DYNEXPDTE99365
```

The P1 DSNINDEX records expires in 30 days.
The P1 ARCHVOLS record does not expire until all of its DSNINDEX records have been deleted.
The P2 DSNINDEX records expires in 15 days.
The P2 ARCHVOLS record does not expire until all of its DSNINDEX records have been deleted.

Here are more examples of creating Dual Primary Volumes and retention periods:

Command (Today=2000.325, RETRETPD=30)	P1 DSN EXPDT	P2 DSN EXPDT
BACKUP	2000.355	2000.355
BACKUP RETPD=5	2000.330	2000.330
BACKUP RETPD2=5	2000.355	2000.330
BACKUP RETPD=5,RETPD2=10	2000.330	2000.335
BACKUP EXPDT=2000.360	2000.360	2000.360
BACKUP EXPDT2=2000.364	2000.355	2000.364
BACKUP EXPDT=2000.360,EXPDT2=2000.364	2000.360	2000.364

Restore/Recover/FMS Recover/Auto Restore

Sysparm RESPRISQ indicates which DSNINDEX records are to be searched, and in which order, for use in the Restore/Recover functions. The values that can be specified are:

- **1** – Search only the P1 DSNINDEX records
- **2** – Search only the P2 DSNINDEX records.
- **12** – Search for P1 DSNINDEX records first but allow use of the P2 record if no P1 is found. (default).
- **21** – Search for P2 DSNINDEX records first but allow use of the P1 record if no P2 is found.

These values can affect the version number of the DSNINDEX records selected. Refer to the description of sysparm RESPRISQ in the Systems Guide for more complete information.

IXMAINT

A parameter PRIMARY2 for the DSNDELETE command indicates whether the P2 records (both DSNINDEX and ARCHVOLS) are to be processed. The values that can be specified are:

- YES – The P2 records are processed as individual records (default).
- NO – The P2 records are not processed.
- ONLY – Only the P2 records are processed.

  By default, IXMAINT treats P2 DSNINDEX records as individual records (e.g., data sets can be expired from P2 volumes without affecting any P1 data sets or volumes) except in COPY processing. In COPY processing, if IXMAINT deletes the P1 record, the P2 record is also deleted. COPY processes are: the COPIES parameter on the DSNDELETE command; SMS backup copies and SMS GDG copies.

Using the following example:

```
DSNDELETE COPIES=5
```

If there are six pairs of P1 and P2 DSNINDEX records, the oldest pair is deleted. If there are only four pairs, none are deleted.

MERGE

A simple parameter **PRIMARY2** for the Merge command indicates that only P2 volumes are selected. The default is that only P1 volumes are selected.

LISTD (in ISPF or batch)

A simple parameter **both** for the LISTD command allows the display of the most current P1 DSNINDEX record and its corresponding P2 DSNINDEX record, if both are available. The default is that the most current record is displayed, which could be a P1 record or it could be a P2 record if there is no corresponding P1 record. For example, if using the commands:

LISTD DSN=A.B.C./,BOTH	The most current P1 record and corresponding P2 record are displayed.
LISTD DSN=A.B.C./	The most current record is displayed, whether P1 or P2.

The P2 flag field is added to the reports and is an option in the FIELDS parameter on the LISTD command.

LISTV (in ISPF or batch)

The P2 flag field is added to the reports.

REBUILD

A simple parameter PRIMARY2 indicates that the DSNINDEX and ARCHVOLS records are rebuilt as P2 records. The P2 flag is set and the appropriate expiration dates are used.

VBACKUP

Dual Primary Volumes cannot be created in the VBACKUP function at this time.

XCOPY

XCOPY cannot process PRIMARY2 volumes at this time.

Using Archive Functions to Manage Active Data

Traditionally, most archived data sets have been truly “inactive” data, and a very small percentage of archived data sets were ever restored. As users begin to use archive functions to manage active as well as inactive data, the number of data sets being restored increases dramatically.

Conceptually, it is probably more accurate to think of these archived active data sets as just temporarily compressed versions, not truly archived versions. In general, users do not want to keep the compressed versions archived very long, if at all, after the data set has been restored, as this wastes disk space.

CA-Disk provides sysparms that allow you to balance the need to manage active data sets with the need to use disk space most efficiently. This is accomplished by creating one archive data set for each data set processed, then allowing the disk space to be reclaimed quickly after the data set is restored from its compressed archive copy.

The process is as follows: Sysparm RESIXRPD allows you to determine whether to keep the archived version of a data set after it has been restored, and if so, for how many days, by resetting its expiration date. The default value is 99, indicating no change is to be made to the expiration date. Specifying a value of 1 causes the DSNINDEX entry to be kept one day after the restore. A value of zero causes it to be deleted immediately. Note that this affects only the DSNINDEX entry. An exception to this rule is that no change is made to the expiration date, nor is the DSNINDEX record deleted, if the NEWNAME, NOLOAD, MEMBER or AIXNAME parameters are specified on a RESTORE command.

Sysparm RESIXDLB can be used in conjunction with RESIXRPD to indicate if DSNINDEX records are also to be deleted (or the expiration date reset) whenever a backup copy is restored. A backup copy does not have the scratch flag turned on in its DSNINDEX record - the data set was not scratched at archive time. The default value N tells CA-Disk not to reset or delete backup DSNINDEX records; that is, only the restore of an archived data set deletes the index record. A value of Y processes all DSNINDEX records alike; that is, the value set in RESIXRPD also controls processing of DSNINDEX records for backup copies.

Then sysparm ARCHPACK can be specified with a value of N during the archive run, telling CA-Disk that “data set packing” is not to be used. The default value Y tells CA-Disk to pack all of the data sets being archived into a single data set. This packing feature alone saves, on the average, one half track of space for every data set processed. When N is specified, packing is not used (that is, CA-Disk creates one archive data set for each data set processed), and disk space is wasted. However, the benefit is that the disk space can be reclaimed quickly after the data set is restored.

Note: This sysparm should be specified as N only for archiving or merging to disk. If used when archiving or merging to tapes or cartridges, each tape contains only one data set!

As an alternative to turning off the archive packing function as just described, consider using the disk-to-disk ability of the merge function to reclaim space occupied by expired data sets in the disk archives. A job to merge only the disk archives does not require any operator intervention, and can therefore be scheduled to run on a regular basis. This implementation maintains the advantages of data set packing, keeps the number of ARCHVOLS entries to a much smaller number, and still reclaims the expired “dead” space.

Archiving the Archives is Not Allowed

Traditional concepts naturally lead you to use retention control to create more backups of the archive data sets being placed on disk -- or to “archive the archives.” If this were permitted, it could lead to mass confusion in trying to manage the archives. For example, you need to restore the archive copy of the archives in order to restore an archived data set. And if the archive copy of the archives was given a different retention period than the original archives, then it can have expired. You can't get the original archives back, unless maybe you took a backup copy that was given a different expiration date, etc. If you need additional backup copies, use the MERGCOPY PROC command.

Instead, CA-Disk allows you to create a duplicate backup copy at the time of the original archive run. Then, if the primary copy is lost, the ARCHVOLS record for it is disabled, and the backup copy is used. If you need additional backup copies, use the archive copy utility as documented...

Note: Creating the duplicate backup copy during the original archive run also requires less processing. The input (user) data is read once and then written to two different output devices -- read, write, write. Using ARCHIVE or RETAIN to create another copy means an additional read -- read and write for the first copy, then read and write it again, increasing I/O by one third.

For the above reasons, CA-Disk does not allow archive and retention control to process the archive data sets written to disk. They are automatically detected and bypassed. The Move/Copy function can be used to copy them, however, since it does not introduce confusion to the archive/restore process. Moving the archives to other disk volumes can be good if you are installing new devices, or perhaps just changing volumes within the archive diskpools.

Archive Integrity and Security

Archive Tape Protection

You can set your security package to assign WRITE (UPDATE) authority to users who must run backup, archive or merge jobs to CA-Disk archive data sets on tape. You can then assign an authority of NONE to all other users. This allows only authorized users to read, modify or delete these archives through utilities outside of CA-Disk.

CA-Disk Restore processing takes care of its own access to the archives, if the CA-Disk sysparm TAPEPRTR is specified with the default Y. To determine the data set names of the CA-Disk archives, see the discussion under "Naming the Archives" on page 61.

CA-Disk password-indicates each archive tape, but does not create any passwords for them, if you specify the CA-Disk sysparm TAPEPROT with a value of Y. Also, this password-indication does not allow users to read, modify or delete these archives through utilities outside of CA-Disk. This protection is selectable, and supplements any protection provided by your security package.

Checks Against Overwriting Valid Tapes

When a new archive or backup tape is to be created, CA-Disk always checks the archive index to see if it is already listed as having valid archive data. This can happen under the following conditions:

- An operator mounts the wrong tape in response to a request to mount a scratch tape.
- You have placed tapes into a pool for the first time, but they have already been used.
- The status of the CA-Disk archive index has not been kept synchronized with your tape management control data set. Your tape management system can have indicated that the tape is an available scratch, but CA-Disk maintenance jobs to indicate it as such have not yet been run.

CA-Disk rejects a tape that is mounted but found to be already in use (it still has valid data on it), and calls for a new tape. Additionally, if short tapes are mounted and tapepools are being used, the next tape is selected from the appropriate pool.

For non-DFP systems (which exist only in the MVS/370 environment) there are two different cases:

- If tapepools are being used, the tapes are validated just as in DFP systems.
- When tapepool support is not being used, however, CA-Disk does not know the volume serial of the tape until after the operator has mounted it in response to the “mount scratch tape” request. If CA-Disk sees that it is already in the archives at this point, data that is possibly very valuable is on the verge of being overwritten. In fact, since the tape label has already been written, the first data set on the tape can be unrecoverable. To prevent any further damage, CA-Disk issues appropriate messages for the tape that can need to be recovered, and then shuts down, keeping the remaining data sets on the tape intact.

Tape Status After Abends

The CA-Disk archive index is updated immediately after each disk data set has been successfully copied to a tape. With this implementation, every archive or backup tape created by CA-Disk is a good tape, even though a system failure or job cancel can have occurred.

If you are using a tape management system, its default action can be the opposite, that is, it can return a tape in use at abend time to scratch status. Consult with your tape management personnel to either change this action for tapes created by CA-Disk or take other precautions to ensure that a partial archive or backup tape is not marked as a scratch tape, and therefore possibly written over by some other job.

If you are using RMM as your tape management system, review APARs OW17304 and OW17377 and consider using the optional JOBNAME Control with program name ADSMI002 and (for Sequential Migrate and Unload) ADSMI000. This functionality allows specific tapes to be kept if they are left in an Open status or were being created when an Abend occurred. Direct any questions on these APARS to RMM support at IBM.

Using the External Data Manager (EDM) interface of a tape management system with CA-Disk is highly recommended. It provides the best method of securing CA-Disk tapes, including those being created as a result of an abend.

If CA1 release 4.8 or higher is installed and the CA-Disk/CA1 EDM interface is used, a tape created by an abending CA-Disk job is controlled by CA-Disk and released only when the data sets on the tape have expired. See “Customizing the CA-Disk Tape Management Support” on page 85 in the Installation Guide.

Note: Contact Customer Support for information about other EDM interfaces that can be available.

If CA1 release 4.7 or earlier is installed and the CA-Disk/CA1 direct interface is used, a CA1 exit (member SLI035 in the USERMODS library) is available for the proper treatment of CA-Disk abend tapes. See “TMSCTLEX — Tape Management System Control Interface Exit” on page 249.

Note: If CA1 release 5.0 or later is installed, do not use the CA-Disk/CA1 direct interface. CA1 macro changes make this interface incompatible.

If the CA-Disk tapepool support is used, tapes are usually indicated to the tape management system as being assigned permanently to CA-Disk and physically separated form the general scratch pool available to non-CA-Disk jobs. Since CA-Disk knows that the partial tape was written, it is not called from the pool until its expiration date has passed. See “Tapepool Considerations” on page 45 for more details.

Archive Disk Protection

You can set your security package to assign all users the authority of NONE to all CA-Disk archive data sets on disk. This prevents utilities other than CA-Disk from reading, modifying or deleting these archives. CA-Disk takes care of its own access to the archives. To determine the data set names of the CA-Disk archives, see the discussion under "Naming the Archives" on page 61.

Unless you set CA-Disk sysparm DISKPROT to a value of N, CA-Disk also password-indicates each archive data set on disk, but without creating any passwords for these archive data sets on disk. This password-indication also prevents utilities other than CA-Disk from reading, modifying or deleting these archives. CA-Disk takes care of its own access to the archives. This protection is selectable, and supplements any protection from your security package.

Processing PDSs that Contain Anomalies

To maintain data integrity for partitioned data sets, CA-Disk performs extensive validity checking against the VTOC entry, the directory and the member data itself. In practice, several inconsistencies can exist at the same time, or a single error can produce multiple error symptoms. In some of these circumstances, it can be impossible to distinguish actual causes from the one or more effects, or to determine a "proper" course of action if an attempt to proceed is to be made. PDS anomaly processing does not apply to PDSE in any CA-Disk function.

If anomalies are found that can affect processing, CA-Disk, by default, issues appropriate diagnostic messages and bypasses the data set, relying on the assumption that someone reviews the messages and corrects the data set. However, provisions have also been made to allow processing to continue for certain types of errors, as defined below.

For each specific PDS error that CA-Disk detects, a 3-digit code has been assigned. The first digit is used to group similar errors together into a common "class" (which is associated with a particular stage of CA-Disk processing), and the last two digits are the specific anomaly code. As each stage of CA-Disk processing is completed, the decision to proceed to the next stage is determined by an error threshold associated with each specific anomaly, and another threshold associated with the class as a whole.

The threshold value is a percentage from 000 to 100, which represents the number of items in error compared to the total number of items in the class. Class "0" items are directory blocks, so threshold values refer to the percent of directory blocks that contain errors. Class "1" and "2" items are the members, so threshold values refer to the percent of members that contain errors.

The default threshold of 000 for each anomaly code causes CA-Disk to bypass any PDS in which an error is detected. Specifying 100 (or any value up to 999) causes CA-Disk to "accept" all errors of that type and continue processing as defined for that error. Threshold percentages between 000 and 100 therefore represent a "degree of error" which you are willing to accept before causing the data set to be bypassed. Closely examine every bypassed data set to determine what corrective action is needed.

The first code in each class does not represent a specific type of error itself. Instead, it represents a summary or composite entry for the entire class of errors, for which you can provide a different threshold. For example, you might want to allow processing to continue if a PDS has I/O errors on 5 percent or fewer of its members (threshold 005 for code 201), and with no more than 2 percent of its members diagnosed as disconnected aliases (threshold

002 for code 207). If there is no overlap between these two error groups, 7 percent of the members could be in error. A threshold of 007 for summary code 200 would allow processing to continue for such a data set. A threshold of 005 for code 200 allows your “worst case” for either code, but not for different members in the same data set.

If you set the summary threshold below any of the specific error thresholds, you effectively lower them to the same value. A special case is that of leaving the summary threshold at 000, which nullifies all thresholds for its class.

Specifying your Options for PDS Error Handling

For some specific error types, CA-Disk provides more than one processing option. You specify both the processing options and the error thresholds in a member of the parmlib data set. This member can be given any name, and is identified to CA-Disk by sysparm PDSANMLYmmmmmmmm.

To change the default PDS anomaly options, create a new member with a name of your choice, and copy SAMPPDSA into it. This sample anomaly table contains all of the error codes and default options in the required format. It looks like the following:

FIGURE 11. PDS Anomaly Table

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT       SBDLM.DMS90.PARMLIB(SAMPPDSA) - 01.02          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 '*---PARTITION DATA SET ANOMALY CODE AND OPTION SPECIFICATIONS---'
 000002 '000       000              DIRECTORY SUMMARY                    '
 000003 '001       000             *DIRECTORY I/O ERROR                  '
 000004 '002       000             *ILLEGAL DIRECTORY FORMAT             '
 000005 '003       000              DIRECTORY KEY 8X"FF" MISSING         '
 000006 '004       000              DIRECTORY BLOCK KEY SEQUENCE         '
 000007 '005       000              DIRECTORY KEY NOT IN BLOCK           '
 000008 '006       000              DIRECTORY MEMBER(S) SEQUENCE         '
 000009 '007       000              DIRECTORY BLOCK ACTIVE BYTES         '
 000010 '100       000              EXTENT VIOLATION SUMMARY             '
 000011 '101       000              MEMBER BEYOND EXTENTS                '
 000012 '102       000     0        MEMBER BEYOND LSTAR                  '
 000013 '*                 0/A      .EXCLUDE MEMBER                      '
 000014 '*                 1/A      .ATTEMPT TO PROCESS MEMBER           '
 000015 '103       000              MEMBER IN DIRECTORY                  '
 000016 '200       000              MEMBER SUMMARY                       '
 000017 '201       000     0        MEMBER I/O ERROR                     '
 000018 '*                 0/R      .RESTORE MEMBER                      '
 000019 '*                 1/R      .NO MEMBER RESTORE                   '
 4 ■                        Aa   A TCPIP        R 4  C 15        9:30   3/11/98
```

PDS Anomaly Code Descriptions and Notes

For each anomaly code, the threshold must be raised above 000 if processing for the data set is to continue after encountering the associated error. For anomaly codes 102, 201, 202, 207 and 208, CA-Disk provides a choice of processing options as described by the comments beneath each. The /A, /R and /A,R indicate whether the option applies to archive, restore, or both. The digit for the selected option must be placed on the line with the anomaly code and threshold.

Anomalies 001 and 002 always result in the data set being bypassed, regardless of the threshold value. The threshold value for anomaly 208 is always ignored (forced to a value of 100), since data integrity is maintained in all cases. Its processing option simply gives control over the number of messages that are potentially issued.

An important consideration for recovery purposes is the possibility for CA-Disk to, by default, never backup a PDS because of anomalies. Therefore, correct any PDS anomalies discovered in a timely fashion.

Password protection is in effect only when not overridden by RACF, CA-Top Secret, or other security systems. Also, it is not in effect on DFSMS volumes.

Processing Defined for Each Anomaly

For each anomaly with a threshold above 000, CA-Disk continues processing as defined below.

TABLE 13. PDS Anomaly Codes and their Descriptions

Code	Processing Description
000	**Percentage of directory blocks with errors** This code does not represent a specific error type, but allows a threshold to be specified representing an acceptable (to you) percentage of directory blocks that can be found to contain errors, regardless of the type of error, and still allow processing to continue. Specific error types and a threshold for each one are defined below. The processing that takes place for each error that is “accepted” is also defined.
001	**Directory I/O error** Logical processing for the data set is terminated whenever an I/O error occurs processing the directory, regardless of the threshold value.
002	**Illegal directory format** A PDS directory is defined as containing one or more records with a key length of 8 bytes and data length of 256 bytes followed by an end-of-file record. It begins at TTR=0000.01, the first record in the data set, and ends within the first extent of the data set. Logical processing for the data set is terminated whenever the above conditions are not found, regardless of the threshold value.
003	**Directory key of 8X’FF’ is missing** The end of the active directory is signified by a directory block with a key length of 8 bytes with all bits on, 8X’FF’. A non-zero threshold allows processing to continue even though this key is not found.
004	**Directory block is out of sequence** PDS member names should be unique and in ascending sequence. The key of each directory block should be the value of the last member name in the block. Therefore, directory block keys should also be unique and in ascending sequence. If a non-zero threshold is specified, a directory block whose key is out of sequence is skipped.

TABLE 13. PDS Anomaly Codes and their Descriptions

Code	Processing Description
005	**Directory block key is not the high member name** Each directory block key should be the value of the last member name in the block. If it is not but a non-zero threshold is specified, processing continues but member names greater than the block key are bypassed, effectively deleting them from the copy being taken.
006	**Directory member names are out of sequence** Directory member names should be unique and in ascending sequence. If they are not but the threshold value is non-zero, duplicate and descending member names are skipped.
007	**Directory block active byte count is wrong** The directory block active byte count is located in the first two bytes of each directory block. Its value is the sum of the lengths of all member entries, plus two for the length of the active byte count field itself. For all active directory blocks, the valid range for this field is 14 to 256. As each entry is processed, a residual byte count is computed by subtracting the length of the entry from the active byte count. Processing of the entries in the block stops when the residual count goes to zero or becomes negative. If all is well, the count is zero. If it is negative, the active byte count was wrong and processing is terminated for the data set unless a non-zero threshold has been specified.
100	**Summary of extent violations** The TTR value of each member entry is tested to see that it points within the data set, within the used space, and not within the directory itself. Each error is logged against its specific code below, but all are logged against this summary entry as well. This permits a threshold value to be specified for this code that represents the percent of members in each data set that can have extent violations, regardless of type, before processing for the data set should be terminated.
101	**Member is beyond the extents** If a member's TTRC points outside the data set's allocated space, a non-zero threshold causes the member entry to be ignored (it does not appear in the copy being taken).
102	**Member is beyond the LSTAR address** If a member's TTRC points outside the data set's used space and a non-zero threshold is specified, a 0 in the option field causes the member entry to be ignored (it does not appear in the copy being taken). A 1 causes CA-Disk to attempt to process it, for which the results are unpredictable.
103	**Member points within the directory itself** PDS member data should start immediately after the end-of-file record for the directory. Any TTR that points into the directory itself is invalid. (One exception is permitted, which is that alias entries can have a TTR=0000.01; that is, they can point to the very first directory block.) If a non-zero threshold is specified, the invalid entries are bypassed, effectively deleting them from the copy.
200	**Percentage of members with internal errors** As the data for each real member (and its associated alias entries) is processed, one or more of the errors as described below can be detected. The error threshold for this code represents the total percent of members that can have internal errors of any type before processing for the data set is bypassed.
201	**Member I/O error** If an I/O error occurs while processing member data and a non-zero threshold is specified, processing for the member continues on the next track. At restore time, processing occurs in one of two ways. If a 0 is specified as the processing option, whatever member data was read and saved successfully, if any, are restored. It should be noted that some records (and potentially all of them) can have been lost. If a 1 is specified, no attempt is made to restore the member.

TABLE 13. PDS Anomaly Codes and their Descriptions

Code	Processing Description
202	**Member not found** Each directory entry specifies a TTR location where the first block of its member data should be found. This error is indicated when an attempt to read the specified block results in a “record not found” condition. (A special case in which this condition is not regarded as an error is described under the topic “Anomalous Conditions Not Regarded as Errors” on page 80.) Two options are available when a non-zero threshold permits processing to continue. A 0 causes the member entry to be maintained, but as a null entry (no data, just an end of file). A 1 causes the member entry to be bypassed, effectively deleting it from the copy.
203	**Duplicate real members detected** Real members should have unique TTR values to identify their locations. In this case, two or more have been found to point to the identical area. An examination of the data is required to determine which are correct and which are in error. If a non-zero threshold permits execution to continue, the duplicate entries are maintained.
204	**Note list size error detected** A user TTRN in the directory entry identifies a note list block at location TTR if the N field has a value not equal to zero. This value also indicates the size of the note list block; that is, the data length is the quantity N times four. An error is detected when the block at that location is read but does not have the proper length. If a non-zero threshold permits processing to continue, the block in question is treated as a standard data block; that is, no attempt is made to treat it as a note list block and resolve internal TTRXs.
205	**Unresolved note list TTRXs** If a note list block of the proper length is found, it should contain TTRXs pointing internal to the member data itself. CA-Disk validates these TTR values such that they can be updated when the member is moved. CA-Disk also permits TTR values less than or equal to 0000.01 to remain unchanged. An error is detected when the TTR fails to satisfy the above tests. If a non-zero threshold is specified, these invalid TTRs are left unresolved when the member is moved, which can lead to unpredictable results in the future.
206	**User TTRN can’t be resolved** A directory entry for a member can contain up to three user TTRs that point to records within the associated member data. Unlike the member’s TTRC, these TTRs must always point to the exact location of a data record. (CA-Disk also permits TTR values less than or equal to 0000.01 to remain unchanged.) An error is detected when the TTR fails to satisfy the above tests. If a non-zero threshold is specified, these invalid TTRs are left unresolved when the member is moved, which can lead to unpredictable results in the future.
207	**Disconnected alias detected** An alias entry has a TTRC that points within its associated real member. (For load modules, the TTRC of an alias entry is the same as the real member’s, but the entry point addresses can differ.) An error is detected when the TTR value of an alias entry fails to point internal to any real member data. (These are commonly caused by improper specification of alias names when using the linkage editor or IEBCOPY to process a member that has aliases. The “old” alias entries were left unresolved, and therefore point to what probably should be considered “down-level” copies of the original member data.) If a non-zero threshold permits processing to continue, there are three options that can be taken, all of which apply at the time the member is restored. A 0 causes the entry to be maintained as a disconnected alias. A 1 causes it to be converted to a real member. And a 2 causes the member to be omitted, effectively deleting it from the new data set.

TABLE 13. PDS Anomaly Codes and their Descriptions

Code	Processing Description
208	**Member record greater than the blocksize** The F1-DSCB specifies the maximum size of all member records in the PDS. An error is detected when a data block is read whose length exceeds the stated value. CA-Disk maintains data integrity by processing the entire data block, regardless of its length. Other applications can, however, truncate these records, which can cause unpredictable results. For the above reasons, CA-Disk always continues processing for this error type, regardless of the threshold value. A processing option of 0 causes a warning message to be issued, indicating the blocksize of the largest block for the data set in error. Option 1 provides this information, plus messages indicating the blocksize for information, plus messages indicating the blocksize for the largest block for each individual member in error.
209	**Note list block not found** A user TTRN in the directory entry identifies a note list block at location TTR if the N field has a value not equal to zero. This error is detected when an attempt to read the block at that location results in a “record not found”. If a non-zero threshold is specified, processing continues but the invalid user TTRN is left unresolved, which can lead to unpredictable results in the future.
210	**Overlapping members** When processing members in their ascending TTR sequence, the end of file for each member should occur prior to the TTRC for the beginning of any subsequent members. An error is detected when this is not the case. The member that has just been read is considered good. All subsequent members whose beginning TTRs have been overlapped are considered invalid. If a non-zero threshold is specified, processing continues but the invalid members are not restored.
211	**Inconsistent user TTR count** More user TTRs are indicated than can be contained within the user data field of this member. When the member is read, all user TTRs outside of the range of the user data fields are ignored. When the member is restored, three processing options are available. If processing option 0 is specified, the directory entry for the member is written back unchanged. If option 1 is specified, the directory entry for the member is restored with the user TTR count adjusted to reflect only the TTR count that can exist in that member’s user data field length. If option 2 is specified, the user TTR count is set to zero when the directory entry is restored.
212	**End of extents encountered prior to end of file** The end of every PDS member is indicated by an end of file record. This error is detected when the end of the data set is reached before encountering the end of file. If a non-zero threshold is specified, the member is processed as if the end of file had been reached instead of the end of the data set; that is, the member is assumed to consist of the data processed up to that point.

Anomalous Conditions Not Regarded as Errors

The following anomalous conditions can be found in PDSs, but their interpretation and processing is common among IBM and many vendors, and are therefore not regarded as errors.

1. Member TTRC not found on track

   A member's TTRC should point to the relative location within the data set of the member's first record. Occasionally this TTRC points incorrectly to the end of a track (to a record that does not exist) instead of pointing to the beginning of the next track where it was actually written.

   This is usually caused by a PDS update program thinking it can start a member at the end of a track but then discovering that the data block is too large to fit in the remaining space on the track. It therefore writes on the next track, but “forgets” to go back and update its starting location, resulting in the TTRC value being technically in error.

Knowing how this condition arises, read routines automatically switch to the next track and return the first record, without indicating an error condition.

2. Alias entries with TTR=0000.01

   Although technically invalid for a standard PDS, this TTR value can be left unchanged without any integrity exposure, and is therefore not regarded as an error. Even when moved to a different device type and location, the TTR value never needs to be resolved -- record one is still record one. Even if the size of the directory is reduced, its minimum size is at least the one block.

   Other vendor products (for example, STROBE) are known to create and use these entries intentionally.

Error Messages and Associated Processing for Anomalies

When an anomaly is detected and a message issued, the error is identified by its associated code and an appended letter. An appended A indicates that the error was detected at archive time, reading the input data set. An R indicates that the error was detected at restore time in the target PDS. An F indicates that the error was in the data contained on the archive medium itself, which means that the errors were allowed previously when the data set was being archived.

When anomalies are detected but your threshold options have allowed the data set to be archived anyway, a “logical error flag” in the index record is set for the data set and “logs” the errors with the archived data set. This permits restore processing to inform you of those facts whenever you restore a copy that contained errors.

Implementing Support for Masstor M860

Masstor Systems Corporation's M860 device is sysgen'd as either a 3420 or a 3480 device. The M860, however, treats the cartridge as either the 3420 or 3480 tape device as it was sysgen'd. Just as native 3420 and 3480 devices require different unit names to prevent requests for conventional tapes to be mounted on a 3480 drive, the M860 also requires this separation.

Providing the unit name of the type of device you want to use is very straightforward when new “tapes” are being created. CA-Disk needs to save this distinction, such that the restore, recover, or merge functions can dynamically request the proper type of device. If this is not done, CA-Disk issues requests to mount M860 cartridges on either 3420 or 3480 tape drives!

Distinguishing the M860 Device

CA-Disk supports the M860 as a distinct device only through the use of dynamic allocation. If archives are written to the M860 by referencing it on a DD statement in your JCL, it appears that it is a 3420 or 3480 and is recorded as such in the ARCHVOLS record (which is subsequently used by restore, recover, and merge processing to allocate what is thought to be the proper type of device).

You inform CA-Disk that an M860 device is being used and should be tracked in the ARCHVOLS as requiring special allocation by providing the proper value for sysparms ARC0TYPE, ARCCTYPE, MERPnTYP, and MERCnTYP. The standard values for these sysparms are TAPE, 3480, or DISK. For the M860, a value of DYN1 must be specified. This causes the “DYN1 special unit flag bit” to be set in the ARCHVOLS record. Subsequent

CA-Disk dynamic allocation processing examines this bit and, if on, uses the unit name provided by sysparm ARESUNI1 for auto-restore, and sysparm DYN1UNIT for all other functions. (DYN2 through DYN9 options are also provided, but are explained later.)

The following table indicates which sysparms you use to provide the unit name for each device type value of sysparm ARC0TYPE. Unit names for the auto-restore function are provided by sysparms beginning with “ARES”, while all other functions take their unit names from sysparms beginning with “DYN”.

TABLE 14. Possible M860 Values for ARC0TYPE

Value	Sysparms for providing your unit names		
TAPE	DYNTUNITuuuu	ARESUNITuuuu	— for your 3420s
3480	DYNCUNITvvvv	ARESUNICvvvv	— for your 3480s
DYNn	DYNnUNITxxxx	ARESUNInxxxx	— for special unit 1

Additional Implementation Needs for the M860

CA-Disk tapepools should be created that consist solely of M860 cartridges, and all cartridges within each tapepool must physically be within the same M860 unit.

If you have multiple M860 units and a unique unit name or unit address can be used to reference each of them, you should use DYN1 to map allocations to tapepools defined in one unit, DYN2 for the second, DYN3 for the third, and so on. DYN2 through DYN9 works in exactly the same manner as described for DYN1, and together provide the ability to identify and access up to three different groups of devices with special allocation requirements.

M860 Unit 1 - Use DYN1 when archiving, and use archive data set names that map to tapepools in this unit. Sysparm DYN1UNITuuuu provides its unit name/address.

M860 Unit 2 - Use DYN2 when archiving, and use archive data set names that map to tapepools in this unit. Sysparm DYN2UNITvvvv provides its unit name/address.

M860 Unit 3 - Use DYN3 when archiving, and use archive data set names that map to tapepools in this unit. Sysparm DYN3UNITwwww provides its unit name/address.

Implementing Support for StorageTek Redwood

The Redwood device supports cartridge capacities of 10GB, 25GB, or 50GB of uncompressed data. The cartridge used by this device has the same size and same characteristics as existing 3480/3490 cartridges, with two obvious differences:

1. The actual tape exits the cartridge from a different corner
2. To prevent these cartridges from being mounted into a true 3480/3490 device, a notch in the corner of each cartridge is flipped on

CA-Disk acknowledges this device in quite the same way it did 3420 and 3480 devices, through an esoteric unitname. Once a unique esoteric is assigned to your Redwood devices, CA-Disk begins storing this name in the ARCHVOL records. This allows CA-Disk to dynamically request the proper device type during RESTORE, RECOVER, XCOPY, or MERGE processing.

For an overview description on the capacities the Redwood device has, refer to "StorageTek Redwood Device" on page 16 in the User's Guide.

Distinguishing the Redwood Device

CA-Disk supports the Redwood as a distinct device only through the use of dynamic allocation. If archives are written to the Redwood by referencing it on a DD statement in your JCL, it appears that it is a 3420 or 3480 and is recorded as such in the ARCHVOLS record (which is subsequently used by restore, recover, and merge processing to allocate what is thought to be the proper type of device).

You inform CA-Disk that a Redwood device is being used and should be tracked in the ARCHVOLS as requiring special allocation by providing the proper value for sysparms ARC0TYPE, ARCCTYPE, MERPnTYP, and MERCnTYP. The standard values for these sysparms are TAPE, 3480, or DISK. For Redwood, a value of DYN1 must be specified. This causes the "DYN1 special unit flag bit" to be set in the ARCHVOLS record. Subsequent CA-Disk dynamic allocation processing examines this bit and, if on, uses the unit name provided by sysparm ARESUNI1 for auto-restore, and sysparm DYN1UNIT for all other functions. (DYN2 through DYN9 options are also provided, but are explained later.)

The following table indicates which sysparms you use to provide the unit name for each device type value of sysparm ARC0TYPE. Unit names for the auto-restore function are provided by sysparms beginning with "ARES", while all other functions take their unit names from sysparms beginning with "DYN".

TABLE 15. Possible REDWOOD Values for ARC0TYPE

Value	Sysparms for providing your unit names		
TAPE	DYNTUNITuuuu	ARESUNITuuuu	— for your 3420s
3480	DYNCUNITvvvv	ARESUNICvvvv	— for your 3480s
DYNn	DYNnUNITxxxx	ARESUNInxxxx	— for special unit 1

Additional Implementation Needs for Redwood

CA-Disk tapepools should be created that consist solely of Redwood cartridges, and all cartridges within each tapepool must physically be within the same Redwood unit.

If you have multiple Redwood units and a unique unit name or unit address can be used to reference each of them, you should use DYN1 to map allocations to tapepools defined in one unit, DYN2 for the second, DYN3 for the third, and so on. DYN2 through DYN9 work in exactly the same manner as described for DYN1, and together provide the ability to identify and access up to nine different groups of devices with special allocation requirements.

Redwood Unit 1 - Use DYN1 when archiving, and use archive data set names that map to tapepools in this unit. Sysparm DYN1UNITuuuu provides its unit name/address.

Redwood Unit 2 - Use DYN2 when archiving, and use archive data set names that map to tapepools in this unit. Sysparm DYN2UNITvvvv provides its unit name/address.

Redwood Unit 3 - Use DYN3 when archiving, and use archive data set names that map to tapepools in this unit. Sysparm DYN3UNITwwww provides its unit name/address.

Implementing Support for IBM's Magstar

Magstar supports cartridge capacities of 10GB and 20GB of uncompressed data. The cartridge used by this device is unique and cannot be used by any other IBM tape device.

CA-Disk acknowledges this device via the use of esoteric unit names. Once a unique esoteric is assigned to these devices, CA-Disk begins storing this esoteric name in each ARCHVOLS record created on the device. This procedure allows CA-Disk to dynamically request the proper input device during XCOPY, RESTORE, RECOVER, or XCOPY processing.

For an overview description on the capacities the Magstar device has, refer to the topic "IBM's Magstar 3590 Tape Device" on page 28 in the User's Guide.

Distinguishing the Magstar Device

CA-Disk supports the Magstar device as a distinct device that can only be accessed through dynamic allocation. If archives are written to a Magstar device by referencing the device on a DD statement in JCL, the Magstar device appears as a 3420 or 3480 and is recorded as such in the ARCHVOLS record (which is subsequently used by MERGE, RESTORE, RECOVER, and XCOPY processing as input to the proper device type).

To inform CA-Disk that a Magstar device is being used is accomplished by assigning the proper value to the following system parameters:

- ARC0TYPE
- ARCCTYPE
- MERPnTYP
- MERCnTYP

The standard values for these sysparms are TAPE, 3480, CART, or DISK. To use a Magstar device, a value of DYN1 must be specified. This causes the "DYN1 special unit flag bit" to be set in the ARCHVOLS record. Subsequent CA-Disk dynamic allocation processing examines this bit and, if on, uses the unit name provided by one of two system parameters:

1. ARESUNI1 — for auto-restore
2. DYN1UNIT — for all other functions

Additionally, sysparm CARTCALC must be specified with a value of “Y” in order for the full length of the tape to be used. As an example, if you were to GEN into your MVS system the esoteric “3590” to represent Magstar, the following sysparm settings would direct BACKUP/ARCHIVE processing to your Magstar devices:

```
//SYSPARMS DD *
DYN1UNIT3590                    <= POINT TO MAGSTAR
ARC0TYPEDYN1                    <= USE DYN1
ARCCTYPEDYN1                    <= USE DYN1
ARC0NAMEyour.primary.name       <= PRIM TAPE NAME
ARCCNAMEyour.copy.name          <= COPY TAPE NAME
CARTCALCY                       <= USE WHOLE TAPE
/*
```

Additional Implementation Needs and Notes for Magstar

Magstar tape cartridges are unique in that they cannot be used in any other IBM or compatible tape system. For this reason, though not required, CA-Disk tapepools should be created that consist solely of Magstar cartridges.

ISPF Custom Reports

The following discussion assumes that the user has a working knowledge of the ISPF online reporting feature of CA-Disk. You should be familiar with “ISPF” on page 417 in the User’s Guide, which covers the online report facility, and also have had hands-on experience with the product. Although the online reporting facility of CA-Disk provides a lot of flexibility in defining user reports, it doesn't allow much tailoring as far as how individual fields are printed (column headings, numeric editing patterns, etc.). You can tailor these values on an installation-wide basis, however, by modifying the data dictionary in parmlib.

Each dictionary element requires two text lines in the parmlib member. These follows the standard rules for continuing information in a CA-Disk parmlib member. The starting column number listed is relative to the first character past the quote. The fields that should not be modified are so documented. No fields should be deleted from this member -- there are other options that allow the suppression of fields from selection lists.

TABLE 16. ISPF Custom Report Field Names

Columns	Name of Field	Description
1-8	FIELD NAME	This is the 8-character field name that appears on the ISPF panels. This field should not be changed unless it is done prior to any reports being defined. If you want to use a different field name
9-16	SOURCE FIELD IN FMTxFLDS	This is the name of the source field in the lower level dictionary. It should never be modified.
17-18	WIDTH OF PRINT DATA	This is a 2-digit number that tells the report writer how wide the field is on the output report. The heading that is specified can cause the actual width of the field to be enlarged on the report.

TABLE 16. ISPF Custom Report Field Names

Columns	Name of Field	Description		
19-32	DATA CONVERSION METHOD	This field tells the report writer how to interpret the source field to get it into a printable format. There are several techniques that are used. The first character gives the method and the next 13 characters define modifiers where applicable. The conversion methods are:		
		*	No data conversion required	
		M	A special module is required to convert the data into its proper format. Details covered later.	
		N	The field is to be printed with the numeric pattern specified (such as ZZ,ZZ9). The edit pattern should be the same length as the field width specified.	
		X	X Print the field as a hexadecimal value.	
		G	Gregorian date. The format is influenced by the width of the field. For example:	
			Width	Format
			6	MMDDYY
			7	DDMMMYY — where MMM is alphabetic (i.e., 12JAN99)
			8	MM/DD/YY
			9	DDMMMYYYY
			10	MM/DD/YYYY
			12	MMM DD, YYYY
		J	Julian date format. The format is influenced by the width of the field. For example:	
			Width	Format
			5	YYDDD
			6	YY.DDD
			7	YYYYDDD
			8	YYYY.DDD
			12	MMM DD, YYYY
		E	European format. The format is influenced by the width of the field. For example:	
			Width	Format
			8	DD/MM/YY
			10	DD/MM/YYYY
33-40	SORT CONVERSION MODULE	If the source field does not sort properly in its natural state, a special conversion module is called to convert the data as needed. If no module is specified, the sort data is extracted directly from the source field.		
41-42	SORT DATA WIDTH	Used only when a sort module is specified. It gives the length of the converted sort data. This field should not be changed for established fields because CA-Disk-supplied routines require the length specified.		
43	PRINT JUSTIFICATION	How should the printed text under the column heading be justified? Choices are:		
		L	left justified	
		R	right justified	
		C	centered	
		A	as is	
		Can be changed to suit your tastes. If “A” is used for numeric fields, the editing pattern specified determines where the field gets printed (Hint: to make the numbers line up on the right column, make the editing pattern as wide as the heading).		

TABLE 16. ISPF Custom Report Field Names

Columns	Name of Field	Description
44-45	SUBGROUP OCCURRENC	Not currently used.
46	FLAG -DISPLAY ON REPORT SCREEN?	Should the field be displayed on the field selection panel where the user selects the fields to appear on the report? (Y/yes N/no) You cannot want the user to have access to all fields (such as RACF flags, catalog flags, etc.) Change this flag to an N and the user no longer sees it on the selection panel, and therefore cannot print it.
47	DISPLAY ON SORT SELECT SCREEN?	This field governs which fields get displayed on the ISPF selection panel for sort fields. A=always
48	DISPLAY ON TOTALS SCREEN?	This field governs which fields get displayed on the ISPF selection panels for sub and grand total fields. A=always, N=never, S=only if field was selected to be reported on. If you change an N flag to A or S, be sure that the source field is defined as numeric.
49-68 and 1-10	COLUMN HEADING FOR REPORT	The heading to use over the column where the value is printed on the report. The heading starts on the first definition line and continues on the next line. Up to 30 characters can be specified. The special character “]” indicates the heading should be split to another print line. There can be up to two of the “]” characters in any given title. Leading and trailing blanks are honored and help to govern the width used for printing the heading. Hint: If you want the headings to appear on the report in a language other than English, you can change them here and all reports automatically reflect the new headings when they are run the next time.
11-52	FIELD DESCRIPTION	A free-form description of the field. This description appears on all ISPF panels where the field is referenced. Again, these descriptions can be changed to a language other than English for the convenience of your users. Any change implemented here takes effect for all users immediately. The description does not appear on any reports that are printed by the system.

Coding ISPF Report Data Conversion Modules

The following documentation provides the minimal guidelines required to code a data conversion routine. Because of the narrow appeal of this type of facility, it is not dealt with in an exhaustive manner.

There can be occasions when the standard methods of data conversion for printing purposes do not cover a user's requirement. The module exit facility allows the user to code special conversion routines and includes them in the data dictionary. There are a few exits that are supplied with the CA-Disk system because of special editing requirements. These include the RACF indicator flags, catalog locate routine, and such fields as DSORG, RECFM, etc.

The module coded in the dictionary is passed the following parameters when called upon to format a field for output:

TABLE 17. Parameters Passed to Dictionary Module

Parameter Description	Type	Length
field name (from dictionary)	INPUT	CL8
source field’s length	INPUT	XL1
source data	INPUT	variable
converted data’s expected width	INPUT	XL1
converted printable data	OUTPUT	variable

The field name is the source field name that is coded in the dictionary definition. This parameter can usually be ignored unless the same conversion routine is called to convert more than one field name.

The source field's length is extracted directly from the dictionary entry. This is used to tell the program how wide the source data (passed as parameter 3) is. The source data is the raw data that is found in the F1 or F4 DSCB.

The final two parameters govern the width and content of the formatted print data. The fourth parameter is used to tell the called program how wide the print area is for this field. The called program under no circumstances should return data that exceeds the length specified. If you need a larger print area, change the value in the dictionary to reflect the required size. Please note that the numeric fields passed are one-byte unsigned binary numbers.

Suppose you need the contents of more than one source field in the DSCB to determine what to print. This was a requirement of a couple of the standard print fields (for example, to print the catalog flag, both the data set name and volume were required to issue a catalog locate). The source data being passed actually resides in a contiguous area of storage that contains both the DSCB and appendage areas.

What happens in the call to the conversion module is that the offset to this field is calculated and then the called program only sees from the source field on in this storage area. So the way to allow addressing to all parts of this area is to specify the field data set name as the source field (since it is the first field in the F1-DSCB). Then you can use the following mapping macros to provide addressability to the entire area: ADSDMDS1, F1APPEND, and S1APPEND.

Coding ISPF Report Sort Conversion Modules

The following documentation provides the minimal guidelines required to code a sort conversion routine. Because of the narrow appeal of this type of facility, it is not dealt with in an exhaustive manner.

The method used to provide a user-written sort conversion routine is very similar to that used for printing purposes. The biggest difference is in the parameter list to the conversion module; there is no parameter describing the length of the source field for the sort. The other notable difference is that the output of the conversion routine does not get printed anywhere.

The module coded in the dictionary is passed the following parameters when called upon to format a field for sorting:

TABLE 18. Parameters Passed to Dictionary Module for Sorting

Parameter Description	Type	Length
field name (from dictionary)	INPUT	CL8
source data	INPUT	variable
converted data’s expected width	INPUT	XL1
converted sort data	OUTPUT	variable

The field name is the source field name that is coded in the dictionary definition. This parameter can usually be ignored unless the same conversion routine is called to convert more than one field name.

The final two parameters govern the width and content of the formatted sort data. The third parameter is used to tell the called program how wide the sort area is for this field. The called program under no circumstances should return data that exceeds the length specified. If you need a larger sort area, change the value in the dictionary to reflect the required size. Please note that the numeric fields passed are one-byte unsigned binary numbers.

User-Specified Condition Codes

CA-Disk assigns a step completion condition code to every step. The code accompanies a message and indicates the circumstances under which the step finished. In most cases, this CA-Disk-generated condition code is the best value for your installation. However, you can override these generated condition codes if your installation has special requirements. This override allows you to select the code that accompanies a message upon the completion of a step. These user-specified condition codes are optional.

Note: A similar option is available for those instances where “NO CATALOG ENTRIES WERE FOUND” exists when DSCL is searching for candidate data sets. See “CCDRESETn” on page 133 for details.

To specify user condition codes, you must create a member in PARMLIB called CCSET. This member is not provided with the system, although a member called SAMPCSET is provided as a sample, and can be copied to create your initial CCSET. User condition codes are then specified by an entry in this member.

When using this option, several technical points must be kept in mind with respect to user condition code processing.

1. Once an N (no-override) is encountered during processing, CA-Disk determines, sets and assigns condition codes for all the message number entries that follow without regard for the overrides specified in member CCSET.
2. Whenever a message is issued during step processing, the highest condition code is set without regard to its origin.
3. Before CA-Disk returns control to the operating system, a check of the CA-Disk-generated return code is made. If the CA-Disk return code is odd and the user condition code option is in effect, then the user override condition code is incremented by one. For example, if the user specified an override condition code of 4 and the CA-Disk generated condition code is odd, then the user override condition code of 4 is incremented to 5. This is needed to produce sorted report listings.
4. No duplicate message number entries are allowed in member CCSET. If duplicates are detected at table initialization, the program abends with a user 200.

Each CCSET entry is composed of a message number, a state flag (O=override or N=no-override), and one of five possible condition codes. The message number is the first four positions followed by a blank. The next two positions consist of the state flag and a condition code respectively.

To add, change or delete entries in the CCSET member, use your online editor or other conventional means for updating a PDS member in the following manner:

1. Create a member named CCSET in the parmlib data set by copying and renaming the sample member SAMPCSET.

2. Specify the message display control entry in CCSET. This must be the FIRST entry in CCSET and is required. This entry controls the display of the message which notifies users that the user condition code option is in effect. It is recommended that you try the default first which displays the message in BATCH and TSO. The format for this entry is:

TABLE 19. Format of $FLGXyz

Field	Meaning		
$FLG	$FLG		
X	BLANK		
y	BATCH FLAG	B	display message under BATCH
		BLANK	do not display message
z	TSO FLAG	T	display message under TSO
		BLANK	do not display message

In order to suppress the display of the notify message under BATCH or TSO processing, simply blank out the respective flag. If you want to suppress the display of the notify message under BATCH and TSO processing, blank out both flags.

3. Specify the message number, state flag and condition code enclosed by apostrophes in the CCSET member. An example could be:

```
'3179 OE'
```

The format for message number, state code and condition code is:

TABLE 20. Format of MMMMBsc

Field	Meaning
MMMM	message number
B	blank
s	state flag
c	condition code

The valid entries for these are:

TABLE 21. Valid entries for MMMMBsc

Field	Possible Values	
MMMM	any valid four-digit CA-Disk message number	
B	must be blank	
s	O	override
	N	no-override
c	G	0, good, successful completion
	I	4, informational message
	R	8, resource error
	E	12, error occurred during CA-Disk processing
	C	16, command error, DSCL command error detected

DASD Threshold Manager

Overview

The CA-Disk DASD Threshold Manager is a program that can help you maintain an acceptable level of freespace on DASD volumes or within a pool of DASD volumes. This program enhances the usefulness of CA-Disk without reducing or altering its functionality.

Most implicit CA-Disk storage management tasks scan VTOCS or catalogs to determine what data sets are archived or moved, based on data set attributes. Data sets are archived or moved with the assumption that this action achieves desirable storage management objectives.

The Threshold Manager begins with the aim of achieving a stated storage management objective: to maintain a specified percentage of freespace balance across a volume or a pool of volumes. Instead of examining data set attributes, the program examines volume (or volume pool) freespace. If the data on the volume increases beyond a specified threshold, causing the freespace to decrease below the desirable percentage, the program determines what data sets could be archived or moved to achieve the balance objective across the volume or pool.

Based on user-specified criteria, the program then generates explicit DSCL Archive or Move Migrate commands for input into CA-Disk. It does not invoke CA-Disk to perform the Archives and Moves directly, but acts as a front-end to CA-Disk.

The Threshold Manager determines the least amount of data to archive or move in order to achieve the balance. If you are processing a pool of volumes, the program archives or moves the least required amount of data. The remaining data sets are distributed within the pool. Each volume finishes with an optimum balance of data and freespace.

The Threshold Manager procedure (THRSHMGR) dynamically allocates two sequential data sets for the DSCL Archive and Move commands it generates. These sequential data sets are specified on the //ARCHOUT and // MIGROUT DD cards in the procedure.

These files become the SYSIN data sets for the DMS procedure. They have an LRECL of 80 and are large enough to handle three records for each complete Archive or Move command generated.

The Threshold Manager is driven by keywords and parameters supplied through the execution of the THRSHMGR procedure and the SYSIN DD statement. The keywords and parameters define the scope of the processing and the criteria used to determine the DSCL commands generated.

The Threshold Manager keywords, parameters and subparameters used in the job control statements use a different syntax than standard CA-Disk processing. This syntax is as follows:

Threshold Manager Control Statement Syntax

- A line is considered to begin at the first non-blank character.
- The line is considered a comment if it begins with an asterisk or contains only blanks through column 72.
- A line is scanned from column 1 through column 72 only.
- After the beginning character, each line is terminated either by a blank character or column 72.
- A line is continued if and only if its end character is a comma.
- A line and all of its continuation lines are considered a statement. Only a keyword can begin a statement.
- Keywords and parameters can be abbreviated to their smallest unique leading characters.
- Keywords, parameters or values cannot contain any blank characters. The only exception to this is when the USEREXIT parameter is present, its null value requests that an IEFBR14 look-alike be used.
- Any list of parameters or parameter values must be separated by commas and enclosed in parentheses. Parentheses are not required when using a single parameter or value.
- Keywords and parameters must be delimited with an equal sign which must be followed with its appropriate value or list of values.

TABLE 22. Threshold Manager Syntax Error Messages

Error Number	Message
100	’CONTINUATION EXPECTED’
101	’KEYWORD DELIMITER “=” NOT FOUND’
102	’MISSING KEYWORD’
103	’INVALID NUMERIC FIELD’
104	’MISSING VALUE’
105	’DUPLICATE POOLNAME’
106	’UNPAIRED PARENTHESIS’
107	’INVALID DSN’
108	’INVALID MEMBER NAME’
109	’DYNAMIC ALLOCATION FAILURE’
110	’INVALID PARAMETER FOR KEYWORD -keyword-’
111	’INVALID DATE’
112	’DSN PARAMETER IS REQUIRED’
113	’MISSING VALUE FOR PARAMETER’
114	’"/” CAN ONLY BE LAST CHAR OF A DSN PATTERN’
115	’DSN NAME EXCEEDS 44 CHARS’
116	’INVALID DSN VALUE’
117	’CONTINUATION NOT ALLOWED’
118	’INVALID KEYWORD’

TABLE 22. Threshold Manager Syntax Error Messages

Error Number	Message
119	’"YES” OR “NO” EXPECTED’
120	’DUPLICATE OR MUTUALLY EXCLUSIVE KEYWORD’
121	’DSNAME REQUIRED’
122	’MULTIPLE VALUES NOT ALLOWED’
123	’LINE CANNOT TERMINATE WITH “(“’
124	’"GLOBAL” CANNOT FOLLOW A “POOLDEF”’
125	’DDNAME STATEMENT DOES NOT EXIST’
126	’DDNAME MUST BE 1 TO 8 CHARS LONG’
127	’DUPLICATE OR CONFLICTING KEYWORDS’
128	’KEYWORD TRUNCATION IS NOT UNIQUE’
129	’INVALID CONTINUATION OF KEYWORD LINE’
130	’POOLNAME MUST BE 1 TO 8 CHARS LONG’
131	’THRESHOLD VALUE MUST BE IN THE RANGE 1-99’
132	’IS PRECEDED BY VOLSER DEFINITION’
133	’EXIT NAME MUST BE FROM 1 TO 8 CHAR’
134	’EXIT MODULE -madmen- FAILED TO LOAD’
135	’INVALID CHANNEL ADDRESS VALUE’
136	’INVALID DEVICE’
137	’INVALID MOUNT STATUS’
138	’INVALID VOLSER PATTERN’
139	’INVALID SUBPARAMETER FOR PARAMETER -parmname’
140	’EXPDT MUST BE AFTER TODAY’
141	’LAST REFERENCE MUST BE BEFORE TODAY’
142	’NEITHER “MIGROUT” NOR “ARCHOUT” DD IS PRESENT’
143	’MIGRATE=YES BUT “MIGROUT” DD IS NOT PRESENT’
144	’ARCHIVE=YES BUT “ARCHOUT” DD IS NOT PRESENT’

Threshold Manager Control Hierarchy

Below is a summary of the keywords, parameters and subparameter hierarchy, and their default values. This is followed by sample JCL to execute the program. A more complete description of the keywords, parameters and subparameters is provided following the JCL.

```
GLOBAL=parameter/parameter list
   ARCHIVE=y/n
   EXPDT=+180/date format
   LASTREF=-7/date format
   VSAM=n/y
   ISAM=n/y
   THRESHOLD=80/1-99
   RECATALOG=y/n
   DELETE=n/y
   MIGRATE=y/n
   USEREXIT=user exit module name
EXCLUDE=DSN=dsn/dsn list/dsn pattern/dsn pattern list
INCLUDE=parameter
   DDNAME=ddname/ddname list
   FILENAME=file name/file name list
MULTIVOL=n/y
POOLDEF=parameter/parameter list
   ARCHIVE=y/n
   EXPDT=date
   LASTREF=date
   VSAM=n/y
   ISAM=n/y
   THRESHOLD=80/1-99
   RECATALOG=y/n
   DELETE=n/y
   MIGRATE=y/n
   USEREXIT=user exit module name
   NAME=pool name
   CHANADDR=channel address/channel address list
   DEVICE=device type/device type list
   MOUNT=pu/pr/s (mount status - public, private, storage)
   VOLSER=volser number/volume pattern/threshold values
```

JCL - Threshold Manager

The following figure illustrates the JCL required for a successful execution of the Threshold Manager:

```
//S2       EXEC THRSHMGR
//S2.SYSIN  DD  *
GLOBAL=(ARCHIVE=,MIGRATE=,VSAM=,ISAM=,EXPDT=,LASTREF=,RECATALOG=,DELETE=,
   THRESHOLD=,USEREXIT=)
EXCLUDE=(DSN=a.b.c/,
      DSN=d.e.f/)
INCLUDE=DDNAME= OR INCLUDE=FILENAME=
MULTIVOL=
POOLDEF=(threshold=,name=,vol=(,),chanaddr=,device=,mount=,archive=,migrate=,
   recatalog=,delete=,vsam=,isam=,expdt=,lastref=,userexit=)
```

Keyword, Parameter, and Subparameter Definitions

The Threshold Manager procedure uses five keywords. Each control statement must begin with a keyword. Although none of the keywords are required, at least one keyword must be supplied.

Each keyword is followed by a parameter, parameter list or value, as follows:

GLOBAL=

The GLOBAL= keyword is used to specify a series of parameters that set global default values. These default values are honored unless they are overridden by a different value supplied on the POOLDEF= keyword. The GLOBAL= keyword can be specified only once, and must precede any POOLDEF keywords.

ARCHIVE= - The default value of this parameter is NO, causing the program to bypass generation of Archive commands. A value of YES causes the program to generate DSCL Archive commands.

MIGRATE= - The default value of this parameter is YES, causing the program to generate DSCL MOVE commands. Specify NO to bypass generating MOVE commands.

EXPDT= - Specify a retention period or date that is applied to the generated Archive commands. If this parameter is not specified, a default retention period value of +180 is used.

- Valid retention period format is “+nnnn”, indicating today plus nnnn days. The date calculated from the retention period cannot exceed the year 2155.

- Valid date formats are:

  `ccYYDDD`[1]
  `ccYY/MM/DD`
  `SYYDDD`[2]
  `SYY.DDD`
  `SYYYYDDD`
  `SYYYY.DDD`

Note: Julian dates that otherwise would be invalid must be prefixed with "S" in order for Threshold Manager to accept them. For example, 99000 would be invalid because a year does not have a day 000. Specifying this value as S99000 instructs Threshold Manager to bypass some of its date checking logic.

1. CC = Century (optional)
2. S= Special Julian Dates

LASTREF= - Specify a time period or a date to cause data sets to become candidates for Archive that have not been referenced during this time or since this date. To specify a time period, use a negative number to indicate the number of days in the past.

- If this parameter is not specified, the default time period value of -7 is used, indicating the data set has not been referenced for seven days.
- If you specify a date, it must be in the past. Valid date formats are:

  ccYYDDD[1]
  ccYY.DDD
  ccYY/MM/DD

RECATALOG= - The default value of YES causes archived data sets to be recataloged to the CA-Disk pseudo-volume. If you specify a value of NO, no recatalog action is taken at archive time. This parameter is mutually exclusive with DELETE. For related information, review "DISP=" on page 220 in the User's Guide.

DELETE= - The default value of NO prevents any catalog action or scratch from occurring during archive time. If you specify a value of YES, archived data sets are scratched and uncataloged. This parameter is mutually exclusive with RECATALOG. For related information, review "DISP=" on page 220 in the User's Guide.

THRESHOLD= - The default value of this parameter is 80, which indicates that the desired freespace threshold is 80 percent (that is, 80 percent of the volume's tracks are allocated). You can specify any value from 1 to 99.

VSAM= - The default value of this parameter is NO, which excludes ICF VSAM clusters from being candidates for Archive or Move processing. If you specify a value of YES, ICF VSAM clusters can be considered candidates for Archive.

ISAM= - The default value of this parameter is NO, which excludes ISAM data sets from being candidates for Archive or Move processing. If you specify a value of YES, ISAM clusters can be considered candidates for both Archive and Move.

USEREXIT= - Specify the name of a user exit module that exists and can be programmatically loaded. The default value is null and a routine equivalent to IEFBR14 is linked as the user exit.

EXCLUDE=

The EXCLUDE= keyword is used to exempt specified data sets from processing. It must precede any POOLDEF= keywords.

DSN= - The DSN= parameter is the only valid parameter for the EXCLUDE= keyword. It cannot be truncated. Specify for this parameter a data set name or pattern or a list of data set names or patterns that are to be exempted from processing. Some examples of valid data set patterns are:

```
prefix/   — exclude all data sets that begin with "prefix"
*.prod/   — exclude production data sets
!abcd!    — exclude data sets that end with "abcd"
```

You can supply one DSN= parameter on an EXCLUDE= keyword, or a list of DSN= parameters.

1. CC= Century (Optional)

INCLUDE=

The INCLUDE= keyword is used to insert a sequential file or DD statement into the SYSIN. The file to be inserted is substituted for the INCLUDE= statement, and must have a logical record length of 80. The include file's syntax must obey all syntax rules, as it is interpreted in exactly the same manner as the instream control statements. Any number of INCLUDE= statements can be used or nested.

DDNAME= - This parameter is used with the INCLUDE= keyword. Specify the DDNAME of the DD statement to be inserted into the SYSIN.

FILENAME= - This parameter is used with the INCLUDE= keyword. Specify the data set name for a sequential file, or the PDS (member) name of the file to be inserted into the SYSIN.

MULTIVOL=

The default value of this keyword is NO, which excludes multivolume data sets from being considered as candidates for Archive processing.

If this keyword is specified with a value of YES, only those multivolume data sets with all of their volumes defined in the specified pool w are considered as candidates for Archive processing. If a multivolume data set is contained in part on a volume not defined within the specified pool, the data set is not considered a candidate.

POOLDEF=

The POOLDEF= keyword is used to define a pool of volumes, limit the Threshold Manager's processing to volumes within the pool (or pools if multiple POOLDEF= keywords are supplied), and to indicate the processing criteria to use in selecting data sets on volumes in the pool to Archive or Move. By specifying a VOLSER= parameter, you can also define subpools within a pool of volumes and specify a THRESHOLD= subparameter to set a different threshold limit for volumes in the subpool. If this keyword is omitted, all volumes defined in the system's UCB table are candidates for processing.

The POOLDEF=keyword must be specified following any GLOBAL= or MULTIVOL=keywords. It can be repeated as necessary to define multiple pools.

Volumes are selected and assigned to pools (and subpools, if a VOLSER= parameter is specified) in the order of definition. Once an online volume is assigned, it is flagged so that it is not reselected.

The 10 parameters defined above for the GLOBAL= keyword can be specified on the POOLDEF= keyword. If specified, they override the global parameters for volumes in this pool. In addition to the 10 global parameters, six other parameters can be specified for the POOLDEF= keyword.

NAME= - Specify a 1- to 8-character name to be assigned to the defined pool. If not supplied, a unique default name is assigned to the pool.

CHANADDR= - This parameter restricts the volumes that can be assigned to this pool by supplying valid volume channel addresses. If this parameter is not supplied, then all channel addresses are candidates for selection.

- Specify one or a list of channel addresses, as defined in the UCB table. Each entry must be four characters in length. Each character must be a valid hexadecimal character (0-F) or the wildcard character “?”.

DEVICE= - This parameter restricts the volumes that can be assigned to this pool by supplying valid device types. If this parameter is not supplied, all device types are candidates for selection.

- Specify one or a list of device types. You can also specify the devices with the following character device codes:

TABLE 23. Device Type Character Code Values

Device Type	Code
3350	B
3375	C
3330-1	D
3380, Any capacity	E
3380, Single density	3380-1
3380, Double density	3380-2
3380, Triple density	3380-3
3390, Any capacity	F
3390, Single density	3390-1
3390, Double density	3390-2
3390, Triple density	3390-3

Note: These character device codes can be used only in specifying this parameter. They are not applicable elsewhere in CA-Disk processing.

MOUNT= - This parameter restricts the volumes that can be assigned to this pool by supplying valid mount status types. If this parameter is not supplied, all mount status types are candidates for selection. Supply PUBLIC, PRIVATE, STORAGE in any combination for mount status selection.

VOLSER= - Supply a 6-character volume serial or pattern, or combinations of both, to specify a subpool of volumes within this pool definition. Pattern characters are the wildcard "?" and the pad to the right “/”. If less than six characters are specified, the pad is assumed.

THRESHOLD= - You can specify one THRESHOLD= subparameter on a VOLSER= parameter, provided it follows the volume serial name or pattern, to set a different threshold for volumes in the subpool.

Implementation Considerations

Although the EXCLUDE= keyword can specify data sets to be excluded from processing, you can also want to exempt certain volumes from processing by the Threshold Manager. Assign these volumes to a pool, then specify ARCHIVE=NO and MIGRATE=NO. This in effect prevents volumes from being selected by a subsequent pool definition.

When a pool is defined with ARCHIVE=NO and MIGRATE=YES, the DSCL Move commands are generated with the objective of distributing the data across the volumes in the same proportion as the relative thresholds. This is the default setting.

When a pool is defined with ARCHIVE=YES and MIGRATE=N0, data sets are selected as archive candidates based only on the threshold limit defined for the volume on which they reside. If MULTIVOL=YES, then the decision to archive is based on the threshold limit of each containing volume. If archive is required to bring any volume to its threshold, the multivolume data set is selected, regardless of whether or not archive is required for the other containing volumes.

When a pool is defined with ARCHIVE=YES and MIGRATE=YES, the Threshold Manager processes volumes in the pool in two phases. During the first phase, candidates for archiving are selected. The threshold limit used is the average threshold of the pool (the sum of the individual volume thresholds, divided by the number of volumes in the pool). Archive candidates are selected from any of the volumes to achieve the pool's average threshold limit. If the volume pool is already at its average threshold limit, no archive commands are generated.

The second phase of processing begins once sufficient archive commands have been generated to bring the pool down to or below its threshold. Using the individual volume threshold values, data sets are selected for migrating, such that each volume ends up at or below its individual threshold.

Threshold Manager User Exit

The Threshold Manager program calls a user exit after a data set or VSAM cluster is determined to be a candidate for archive or move processing. A logical 32-bit preference weight has already been assigned as an archive preference weight. This number is the logical inverse of the right justified 3-byte last reference date. Candidates are placed on a list in ascending order of this preference weight.

At this point the user exit can make the following decisions.

1. Reject the candidate for archiving.
2. Reject the candidate for move.
3. Reject the candidate for both archiving and moving.
4. Change the archive weight, thereby changing the placement of the candidate on the archive list. (Archive candidates are used in the order in which they appear on the candidate list until the desired threshold is attained or the list is exhausted.)

A user exit can be implemented for all processing by specifying the unique member name in the USEREXIT= parameter of the GLOBAL= keyword. At the pool level, this value can be overridden by specifying the unique member name in the USEREXIT= specify and call any user exit load module by supplying its parameter of the POOLDEF= keyword.

Any number of pools can use the same user exit module, but different pools can used different exits. If no user exit module is specified for a pool, it uses the equivalent of an IEFBR14.

The exit module is made available to the processor by a standard load. This means it must be located in the library specified by the JOBLIB or STEPLIB DD statement.

A skeleton user exit module is provided by the name THRESEXT and is included in the installation library. It can be compiled by any assembler compiler and needs no special macros.

This skeleton contains the layout for a candidate record and provides addressability. Your decision can be based on attributes of the data set name or cluster name that is provided in the record. After you have made your decision, you need only to branch to the label in the exit program that affects your decision.

Creating and Using Dual Primary Volumes

Differences of Primary/Copy Volumes vs. Dual Primary Volumes

The typical CA-Disk backup/archive process creates duplicate outputs on tape or disk referred to as the *primary* and *copy* archive volumes (or, ARCHVOLS).

An optional process that can be utilized during backup/archive is to create **Dual Primary Volumes**. Using this option, the backup/archive process will create PRIMARY1 and PRIMARY2 (P1 and P2) volumes on tape or disk.

Here are some distinctions between these two processes:

	Primary/Copy	**Dual Primary Volumes**
1	The primary and copy volumes contain exactly the same data.	The two primary volumes contain exactly the same data.
2	The primary/copy volume relationship does not change for the life of the volumes.	The P1 and P2 volumes are not connected or related in any way.
3	For each data set processed, one DSNINDEX record is created.	For each data set processed, a P1 DSNINDEX record and a P2 DSNINDEX record is created.
4	A primary ARCHVOLS and a copy ARCHVOLS record are created for the output volumes.	A P1 ARCHVOLS record is created for the P1 output volume and a P2 ARCHVOLS record is created for the P2 output volume.
5	The DSNINDEX record contains the volser (or key) of the primary ARCHVOLS record.	The P1 DSNINDEX record contains the volser (or key) of the P1 ARCHVOLS record and, the P2 DSNINDEX record contains the volser (or key) of the P2 ARCHVOLS record.
6	The primary ARCHVOLS record contains the key of the copy ARCHVOLS record.	The P1 ARCHVOLS record has no field pointing to the P2 ARCHVOLS record, nor does the P2 ARCHVOLS record point to the P1 ARCHVOLS record.
7	The primary and copy archvol can <u>not</u> be managed separately.	The P1 DSNINDEX and P1 ARCHVOLS records can be managed independently of the corresponding P2 records.
8	The copy ARCHVOL is kept until the primary ARCHVOL expires, at which time the copy ARCHVOL is also expired.	The P2 DSNINDEX and P2 ARCHVOLS records can expire before, after or at the same time as the P1 records.

Important Considerations of the Dual Primary Volume Process

In an execution of CA-Disk, the option to create Dual Primary Volumes can be utilized or Primary/Copy volumes can be created. These are mutually exclusive processes in the job step being executed. The Copy Utility can be used to create copies of the Dual Primary Volumes.

All of the P1 and P2 DSNINDEX and ARCHVOLS records are maintained in the FILES data set, along with the entries for the primary/copy ARCHVOLS and DSNINDEX records.

The DSNINDEX records and the primary ARCHVOLS records created in the primary/copy process are considered as P1 records (e.g., the “P2” indicator flag is *off*).

There is no distinction - and no method to distinguish - between primary DSNINDEX records and ARCHVOLS created in the primary/copy process and the *"P1"* DSNINDEX records and ARCHVOLS created in the Dual Primary Volume process.

Although the Dual Primary Volume feature provides very flexible options in managing dual output images and volumes, the user must be aware that there could be a significant increase in the number of DSNINDEX records maintained in the FILES data set. Before using this feature, these factors must be considered:

- Insure that the physical size (number of cylinders allocated) of the FILES data set is adequate to contain the expected increase of DSNINDEX records.
- Insure that the record (or, indexing) capacity for the DSNINDEX subfile is adequate (specified in member FILEDEFN in the CA-Disk parmlib).
- The runtime of IXMAINT may be longer because of the additional records created by this feature.

Dual Primary Volume Feature Overview

The following are the functions where the P1 and/or P2 records can be selected or viewed:

- **RESTORE/RECOVER** option will select P1 and/or P2 DSNINDEX records.
- **IXMAINT** option will select P1 and/or P2 records for maintenance processing.
- **MERGE** option will select P1 or P2 volumes.
- **LISTD** option will display both P1 and P2 DSNINDEX records.
- **LISTD** and **LISTV** will show the setting of the P2 flag field.
- **REBUILD** option will restore the P2 indicator for the DSNINDEX and ARCHVOLS records when the P2 information is available.
- Merging of FILES Data Sets includes the P2 indicator for the sort sequence.

Except for the functions mentioned above, all other CA-Disk functions operate without consideration for the P2 record indicator.

Dual Primary Volume Details by Function

Archive/Backup

Dual Primary Volumes will be created when specifying the following statement in the DSCL:

```
SET  PRIMARY2
```

When SET PRIMARY2 is specified, a P1 and P2 DSNINDEX record is created for each data set processed, and a P1 and P2 ARCHVOLS record is created for each respective output volume.

The Primary2 Volume data set name and unit information are taken from the ARCCNAME and ARCCTYPE sysparms.

Different expiration dates can be applied to the P1 and P2 DSNINDEX and ARCHVOLS records. The P1 DSNINDEX and ARCHVOLS records can have expiration dates that are greater than, less than or equal to the P2 records expiration dates.

- Parameters RETPD and EXPDT apply to the P1 DSNINDEX records, except as noted below.
- Parameters **RETPD2** and **EXPDT2** apply to the P2 DSNINDEX records as they are created.
- If P2 DSNINDEX records are being created and RETPD2 or EXPDT2 is not specified, the expiration date will be taken from the RETPD or EXPDT parm, if specified.
- If P2 DSNINDEX records are being created and neither RETPD/EXPDT or RETPD2/EXPDT2 parameter is specified, the retention period for both the P1 and P2 DSNINDEX records is taken from sysparm RETRETPD.

For SMS data sets, if the Management Class is present and not overridden (e.g., sysparm SMSMCBYP is set to N or B), the RETPD/EXPDT and RETPD2/EXPDT2 parameters are ignored and the expiration date of the P1 and P2 DSNINDEX records is set to 99365. Those P1 and P2 DSNINDEX records continue to be managed using DFSMS and can not be managed separately.

The P1 and P2 ARCHVOLS records can be assigned different expiration dates:

- Sysparm DYNEXPDT will apply to the P1 ARCHVOLS records, except as noted below.
- Sysparm **DYNEXPD2** will apply to the P2 ARCHVOLS records as they are created.
- If P2 ARCHVOLS records are being created and DYNEXPD2 is not overridden in the SYSPARMS member or in the job, the expiration date for both the P1 and P2 ARCHVOLS records is taken from sysparm DYNEXPDT.

Here is an example of using the new parameters and the results:

```
SET PRIMARY2
SCAN REALVOLS
BACKUP RETPD=30,RETPD2=15
```

Sysparms:

```
DYNEXPDTE99365
```

The P1 DSNINDEX records will expire in 30 days.
The P1 ARCHVOLS record will not expire until all of its DSNINDEX records have been deleted.
The P2 DSNINDEX records will expire in 15 days.
The P2 ARCHVOLS record will not expire until all of its DSNINDEX records have been deleted.

Here are more examples of creating Dual Primary Volumes and retention periods:

Command (Today=2000.325, RETRETPD=30)	P1 DSN EXPDT	P2 DSN EXPDT
BACKUP	2000.355	2000.355
BACKUP RETPD=5	2000.330	2000.330
BACKUP RETPD2=5	2000.355	2000.330
BACKUP RETPD=5,RETPD2=10	2000.330	2000.335
BACKUP EXPDT=2000.360	2000.360	2000.360
BACKUP EXPDT2=2000.364	2000.355	2000.364
BACKUP EXPDT=2000.360,EXPDT2=2000.364	2000.360	2000.364

Restore/Recover/FMS Recover/Auto Restore

Sysparm **RESPRISQ** will indicate which DSNINDEX records are to be searched, and in which order, for use in the Restore/Recover functions. The values that can be specified are:

- **1** – Search only the P1 DSNINDEX records
- **2** – Search only the P2 DSNINDEX records.
- **12** – Search for P1 DSNINDEX records first but allow use of the P2 record if no P1 is found. (default).
- **21** – Search for P2 DSNINDEX records first but allow use of the P1 record if no P2 is found.

These values may affect the version number of the DSNINDEX records selected. Please refer to the description of sysparm RESPRISQ in the Systems Guide for more complete information.

IXMAINT

A parameter **PRIMARY2** for the DSNDELETE command will indicate whether the P2 records (both DSNINDEX and ARCHVOLS) are to be processed. The values that can be specified are:

- YES – The P2 records are processed as individual records (default).
- NO – The P2 records are not processed.
- ONLY – Only the P2 records will be processed.

By default, IXMAINT will treat P2 DSNINDEX records as individual records (e.g., data sets may be expired from P2 volumes without affecting any P1 data sets or volumes) except in COPY processing. In COPY processing, if IXMAINT deletes the P1 record, the P2 record will also be deleted. (*COPY processes* are: the COPIES parameter on the DSNDELETE command; SMS backup copies and SMS GDG copies.)

Using the following example:

```
DSNDELETE COPIES=5
```

If there are six pairs of P1 and P2 DSNINDEX records, the oldest pair will be deleted. If there are only four pairs, none will be deleted.

MERGE

A simple parameter **PRIMARY2** for the Merge command will indicate that only P2 volumes are to be selected. The default is that only P1 volumes will be selected.

LISTD (in ISPF or batch)

A simple parameter **BOTH** for the LISTD command will allow the display of the most current P1 DSNINDEX record and its corresponding P2 DSNINDEX record, if both are available. The default is that the most current record will be displayed, which could be a P1 record or it could be a P2 record if there is no corresponding P1 record. For example, if using the commands:

`LISTD DSN=A.B.C./,BOTH`	The most current P1 record and corresponding P2 record are displayed.
`LISTD DSN=A.B.C./`	The most current record is displayed, whether P1 or P2.

The P2 flag field will be added to the reports and will be an option in the FIELDS parameter on the LISTD command.

LISTV (in ISPF or batch)

The P2 flag field will be added to the reports.

REBUILD

A simple parameter **PRIMARY2** indicates that the DSNINDEX and ARCHVOLS records will be rebuilt as P2 records. The P2 flag will be set and the appropriate expiration dates will be used.

VBACKUP

Dual Primary Volumes can not be created in the VBACKUP function at this time.

XCOPY

XCOPY can not process PRIMARY2 volumes at this time.

CHAPTER 4

Sysparms

Each installation using CA-Disk to manage data storage has unique needs, requiring a variety of techniques for implementing CA-Disk to provide compatibility with each environment. To meet these various needs, CA-Disk was developed to use system parameters (sysparms) to control various aspects of each function. The default value of each sysparm can be altered by users to meet the operating requirements of each installation. Most system parameters can also be specified temporarily, through overriding dd statements, to accommodate any special circumstances of a particular job run.

Each function in CA-Disk has logic to access a sysparm when it is necessary to determine the value in effect. Once the value has been retrieved, program logic will invoke the code to perform the function according to the sysparm value specified.

CA-Disk assigns a default value to every system parameter and will execute according to this default value unless a different value is assigned through an entry in the parmlib data set member SYSPARMS or through an overriding dd statement.

The CA-Disk default value of a sysparm is assigned, based on the “typical” installation needs. In many cases, the default value will be the best value for your installation and no adjustments are required. Only if the default value does not meet your installation's needs will you need to make any changes.

To change a sysparm to a value other than its default value, first create a member in the parmlib data set called SYSPARMS. This member is not provided with the system. However, a member called SAMPARMS is provided as a sample, and can be copied in to create your initial SYSPARMS member. System parameters are then specified in the SYSPARMS member, one parameter per line. If more than one entry is entered on a line, all but the first is treated as comments.

Each system parameter is composed of a parameter name and a parameter value. The name is the first eight characters and the value is appended immediately after the name. All parameters have unique names; the length and type of data required in the value differ. The default value and the acceptable user-specified values for each sysparm are documented later in this section under each sysparm name.

To add, change or delete entries in the SYSPARMS member, use any online text editor.

System Parameter Names Reserved for Users

Occasionally you can need to create your own system parameters, perhaps in conjunction with a user exit or for user-supplied applications using CA-Disk modules. See "Modifying CA-Disk" on page 299. In order to ensure that sysparms you create will not conflict with CA-Disk-supplied sysparms in a future release, specify sysparms you create with a name beginning with a dollar sign ($). By definition, all sysparm names beginning with a $ are created and specified by the user or installation.

Overriding Sysparms Instream

To provide maximum flexibility, CA-Disk also permits you to supply an input stream of overriding system parameters through the //SYSPARMS dd statement. Sysparms thus supplied either supplement those specified in your SYSPARMS member of parmlib or override them, if already present.

The dd statement must point to a sequential file with DCB attributes of LRECL=80 and RECFM=F or FB. It can be a dd * type input stream. Sysparms start in column 1 of the record image, as shown in the example below:

```
//SYSPARMS DD *
* COMMENTS HAVE ASTERISK IN COLUMN ONE.
ADPCTPDS025
FLYCOUNT0
PRIEXMODMYEXIT         ADD A USER EXIT
'DSDISPEX '            BLANK OUT USER EXIT
/*
```

The capability to override sysparms through an input stream is enabled by specifying sysparm SYSPARMO with a value of Y or S in the SYSPARMS member of parmlib. See "SYSPARMOn" on page 181 for details. The default value of N disables this feature.

Note that the sysparm values specified in the input stream are temporary overrides. They do not update the contents of the SYSPARMS member of the parmlib data set. To protect against unauthorized use of restricted functions or operations, certain sysparms cannot be overridden by this technique.

System Parameters

TABLE 24. System Parameters

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
ACF2FORC	•	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	•	•	•	
ACF2SUPP	•	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	•	•		
ADPCTOTH				•	•		•											•				
ADPCTPDS				•	•		•											•				
ADTRACKS				•	•		•											•				
ALLOCPRI	•	•	•													•	•	•				
ALLOEXIT				•	•		•															
ALLOVBLK				•	•		•											•				
ALOCATEX				•	•		•										•	•				
ALOPOSEX				•	•												•	•				
ALTTRKAS				•	•	•												•				
ARBACKUP				•																•		•
ARC0TYPE		•	•			•					•											
ARCATOPT		•	•																			
ARCBLKSI		•	•																			
ARCCNAME		•	•			•																
ARCCPYDV		•								•	•											
ACRRPYDV																						
ARCCTYPE		•	•			•					•											
ARCDSKBZ		•	•								•											
ARCDSNEX		•	•			•					•											
ARCDSORG		•	•																			
ARCEMPTY		•	•																			
ARCEODEX		•	•		•																	
ARCFDEFR		•	•																			
ARCHIVEX		•	•																			
ARCHPACK		•	•	•	•		•		•	•	•											
ARCHSORT		•	•			•																
ARCMODEL		•																				
ARCMULTI		•	•																			

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
ARCNOSTD		•	•																			
ARCPLENQ		•	•																			
ARCRETPD		•	•																	•		
ARCSEPC0		•	•								•											
ARCTNAME		•	•			•			•	•	•											
ARCVOLID		•	•								•											
ARC0NAME		•	•			•																
ARDDNEXC				•							•											
ARDIAGNM				•																		
ARDSNPOL				•																		
ARESCRDT				•																		
ARESPREX				•																		
ARESTENQ				•																		
ARESUNIC				•																		
ARESUNIT				•																		
ARESUNIn				•																		
ARJOBEXC				•																		
ARPOOLEX				•																		
ARRESDELn				•																		
ARSECURE				•															•			
ARTAPEOK				•																		
ARTASKNM				•																		
ARVOLPOL				•																		
CARTCALC		•	•							•	•					•						
CARTFEET		•	•							•	•					•						
CARTPRCT		•	•							•	•					•						
CATBKDEL					•		•															
CATBKNAM					•		•															
CATBKSPC					•		•															
CATBKUNT					•		•															
CATLOCWA																						•
CATPURGE		•	•																			
CCDRESET	•	•	•												•			•			•	•

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
CCINITEX		•	•															•				
CNVATCYL				•	•		•															
DARCHIEX		•	•																	•		•
DARCKEEP		•	•																	•		
DASDPMAX				•	•																	
DASDVMAX				•	•		•															
DATECUTO	•	•	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	•	•	•
DBDSNCFD																					•	
DBDSNINX																					•	
DBRCDBSZ		•	•	•	•		•									•		•				
DBRCICSZ		•	•	•	•		•									•		•				
DBRCPREX		•	•	•	•		•									•		•				
DBRCRLIB		•	•	•	•		•									•		•				
DBRCRLSE		•	•	•	•		•									•		•				
DBRCSMUT		•	•	•	•		•									•		•				
DBRCSUPP		•	•	•	•		•									•		•				
DBRCVCUT		•	•	•	•		•									•		•				
DCAVGCMP		•	•																			
DCCMPEX0		•	•			•	•															
DCCMPEX1		•	•																			
DCCMPEXn		•	•																			
DCCOMPTC		•	•			•																
DCDATACP		•	•		•	•	•															
DCDCPEXn				•	•	•	•															
DCDSNDEX		•	•																			
DCEXCTBL		•	•			•																
DCINCTBL		•	•			•																
DCRTSTAT		•	•																			
DEFARCEX		•	•																			
DELETEOK																						
DERASEEX		•	•	•																•		
DFLTPOOL				•	•		•															
DIAGALOC				•	•																	

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
DIAGAUTH	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
DIAGEXCP		•	•	•	•		•				•							•				
DIAGVSRL		•	•	•	•		•															
DINXUFEX		•	•	•	•	•	•															
DISKPROT		•	•			•				•												
DLVOLMSG	•	•	•		•																	
DRESARCH				•																		
DRESKEEP				•																•		
DRESTOEX				•																•		•
DSBALTRT																					•	
DSBCATSZ																					•	
DSBSELVS																					•	
DSBTRKSZ																					•	
DSBUNITS																					•	
DSBUSREX																					•	
DSBVOLCT																					•	
DSCBCVEX				•	•		•				•						•	•				
DSCBJBNM	•	•	•	•							•											
DSCBLMOD	•										•											
DSCBLUSD	•										•							•				
DSCBOPCD	•										•											
DSCBSVMD	•			•	•						•						•	•				•
DSCBUSRD																					•	
DSCLMSGS	•	•	•			•									•			•				
DSCLOWRC		•	•			•									•			•				
DSCLRPLY	•	•	•			•									•			•				
DSCLSCEX	•	•	•	•		•									•							
DSDISPEX		•	•																			
DSNDELEX									•													
DSNDLPEX									•													
DSNDUPVL				•	•				•					•								
DSNSELRE	•	•	•			•									•			•			•	
DSSEIOTR			•																•			

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
DSSMSGDD			•																•			
DSUCATLG	•																					
DSUTILCF	•																					
DSUTILSQ	•										•											
DYNALDSN										•												
DYNCUNIT		•	•	•	•		•			•	•											
DYNEXPDT		•	•			•				•	•											
DYNTUNIT		•	•	•	•		•			•	•											
DYNUKEEP		•	•	•	•	•	•			•	•											
DYNVREQD		•	•							•	•											
DYNnUNIT		•	•	•	•		•			•	•											
ENQADDTL		•	•	•	•	•	•								•	•	•	•				
ENQDSSIM																	•					
ENQGDGBY		•	•	•	•	•	•												•			
ENQJES3T		•	•	•	•	•	•			•					•		•	•	•			
ENQMSGES		•	•	•	•										•	•	•	•				
ENQSTPOK	•	•	•	•	•	•	•								•	•	•	•		•		•
ENQVOLWT		•	•	•	•		•			•	•	•										
ENQWAITD	•	•	•	•	•	•	•								•	•	•	•		•		•
EXPARSOP		•	•																			
EXPASCOP		•	•																			
EXPDTUSE		•	•																			
FASTCOMP																	•					
FILESCNT														•								
FILESPAC		•	•	•	•	•			•	•	•	•	•	•		•			•	•	•	•
FILEUNIT		•	•	•	•	•			•	•	•	•	•	•		•			•	•	•	•
FILOGNAM		•	•	•	•	•			•	•	•	•	•	•		•			•	•	•	•
FLYCOUNT	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•		•	•	•
FMSSAFEX				•	•																	
FMSSPREX				•	•																	
FORCPOOL		•	•						•	•	•		•									
GDGCATEX				•	•												•	•				
HONORTTR				•	•		•											•				

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
ICHEKAEX		•	•	•	•		•				•							•	•	•		
ICHEKPEX		•	•	•	•		•				•							•	•	•		
IDRCRATE		•	•			•				•												
IDRCSIZE		•	•			•				•												
IDRCUSED		•	•			•				•	•											
IGNLSTAR		•	•																•			
IOCHNBLKn		•	•			•				•	•											
IOERRLIM						•																
IOMAXREC		•	•	•	•	•	•															
IOTRACKS		•	•	•	•		•												•			
ISAMRECO		•	•															•				
IXCTLGEX									•					•								
IXMCLEAN									•					•								
IXMDUPDT									•					•								
IXMQTIME		•	•	•	•	•			•	•		•	•	•								
IXMREVEX									•													
IXMVOLON									•	•												•
IXUSRNOT									•													
LISTDMEX																				•		
LISTDVEX								•												•		
LISTREEX																				•		
LISTSORT				•																		
LISTVTOC					•																	
LUSEDATE				•	•		•															
MERCnNAM										•	•											
MERCnTYP										•	•											
MERDISVL										•												
MERDSNEX										•												
MERFASTS										•												
MERPnNAM										•	•											
MERPnTYP										•	•											
MERRELTM				•						•												
MERREPLY										•	•											

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
MERVOLEX										•												
MIGBYPAS																•						
MIGCATLG																•						
MIGCREDT																•						
MIGCTGDG																•						
MIGDISEX																•						
MIGDSMAX																•						
MIGDSORG																•						
MIGEXPDT																•						
MIGFXPDT																•						
MIGGDGEX																•						
MIGMODEL																•						
MIGNOEXP																•						
MIGNXPDT																•						
MIGPSCOP																•						
MIGPSDEN																•						
MIGREXIT																•						
MIGRSORT																•						
MIGYRCHG																•						
MVDCATLG	•																					
NEWPATHN				•	•																	
NOBLKMSG																		•				
OFFLDASD				•																		
ONCCFAIL		•	•															•				
PASSNEWN				•							•							•				
PASSWORD		•	•	•	•	•	•				•				•	•	•	•		•		
PDSANMLY		•	•	•	•		•															
PDSELOAD		•		•	•																	
PDSENQSH																	•					
PDSEXSIM																	•					
PDSMAXPR																	•					
PDSMAXSE																	•					
PDSMINDB																	•					

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
PDSMINPR																	•					
PDSMINSE																	•					
PDSRRMAX																		•				
PDSSETBS																	•					
POINTABS				•	•		•			•	•											
PREALLOC				•	•		•															
PRIALLOC				•	•												•	•				
PRTSORTF	•																					•
PS2LSTARn		•																				
PSUDEVTP		•	•																			
RAADDAEX																		•	•	•		
RAADDPEX																		•	•	•		
RACFALLO				•	•		•				•								•	•		
RACFALWZ	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•	•	•		•
RACFBITS				•	•													•	•	•		•
RACFBKUP																			•	•		
RACFDAEX											•								•	•		•
RACFDPEX											•								•	•		•
RACFDVL2																	•					
RACFDVOL		•	•	•	•	•	•	•	•	•	•								•	•		•
RACFEAEX		•	•	•	•	•	•	•	•	•	•								•			•
RACFENTY		•	•	•	•	•	•		•	•	•								•	•		•
RACFEPEX		•	•	•	•	•	•	•	•	•	•								•			
RACFMDSN				•	•		•		•										•			
RACFMODL				•	•		•		•		•								•			
RACFMVOL				•	•		•		•		•								•			
RACFNEWN		•	•	•		•					•		•					•	•	•		•
RACFPDSW																	•		•			
RACFPRED				•	•		•											•	•			
RACFPROC	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•	•		•	
RACFSEQM																•			•			
RACFSEQT																•			•			
RACFSMEX																•			•			

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
RACFSUPP	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•	•		•	
RACFUSID	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•	•			
RACFVCAV																		•	•			
RACGDGVL	•	•	•	•	•	•					•				•	•	•	•	•	•		•
RACHKAEX	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•	•			
RACHKPEX	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•	•			
RADEFAEX	•	•	•	•	•	•	•	•		•	•			•			•	•	•	•		
RADEFPEX	•	•	•	•	•	•	•	•		•	•			•			•	•	•	•		
RADELAEX									•	•									•			•
RADELPEX									•	•									•			•
RADELSVC																			•			•
REBLIMIT																		•				
REBLKBIT				•	•													•				
REBYPASS																		•				
RECATPSU		•	•																			
RECATVOL		•	•	•	•		•		•													
RECCHCAT					•		•															
RECPRIEX					•																	
REDEVDEF		•	•	•	•		•															
RESAFTEX				•	•		•															
RESCHCAT				•																		
RESCOPYS				•	•		•				•											
RESCOPYT				•	•		•				•											
RESCRCAT				•	•																	
RESCRNEX				•	•															•		
RESIXDLB				•	•		•		•	•												
RESIXRPD				•	•		•		•	•												
RESPRIEX				•	•		•															
RESRETPD				•	•		•															
RESTCTLG				•	•		•															
RETAFTEX				•	•		•															
RETDSNCD				•	•		•															
RETDSNCL				•	•		•															

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
RETDSNDF				•	•		•															
RETDSNLU				•	•		•															
RETEXCLD				•	•		•															
RETEXCMD				•	•		•															
RETGDGEX				•	•		•															
RETGDGNM				•	•		•															
RETMULTI				•	•		•															
RETNODSP				•	•		•															
RETNOMIG				•	•		•															
RETPREFT				•	•		•															
RETPRIEX				•	•		•															
RETRETPD				•	•		•															
RLSEAFEX															•							
RLSEDIAG															•							
RLSEMAXS															•							
RLSEMINS															•							
RLSESECR															•							
RLSESYS1															•							
RLSEUNMV															•							
RPTCLSEL	•																					
RPTDSBEX	•																					
RPTDSNEX	•																					
RPTDSUEX	•																					
RPTLINES	•																					
RPTMVDEX	•																					
RPTNMTSZ	•																					
RSCHGBIT				•	•																	
RSUPPRES									•													
SAMSYSNM				•																		
SCRASYS1		•	•	•	•		•									•	•	•		•		
SCRTCHEX		•	•	•	•		•									•	•	•		•		
SECARVOL		•	•	•	•	•	•	•	•	•									•	•		
SECFRVOL				•	•		•												•		•	•

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
SECURAEX	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	
SECURCMD	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•		•	
SECURDRS				•																		
SECURFIL		•	•	•	•	•	•	•	•	•	•	•	•	•		•			•	•		•
SECURLIM	•	•	•	•	•	•	•	•	•	•	•	•	•	•		•			•	•		•
SECURLOG	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•		•	
SECURAEX	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	
SECURPEX	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	
SECURRPT	•							•			•								•	•		
SECURVOL	•	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	•	•		
SECUSUPP	•	•	•	•	•	•	•				•				•	•	•	•		•		•
SELECTAM		•																		•		•
SELECTDA		•																		•		•
SELECTPO		•																		•		•
SELECTPS		•																		•		•
SELECTRE	•	•	•			•									•			•			•	
SHRNKFDT		•	•	•	•	•	•															
SHRNKSYS		•	•	•	•																	
SMFCDATD	•																					
SMFCTIMD	•																					
SMFDFLAG	•																					
SMFDSNMD	•																					
SMFJDATD	•																					
SMFJTIMD	•																					
SMFTYP14	•																					
SMFTYP15	•																					
SMFVLSRD	•																					
SMSACSDF																			•			
SMSALLOC				•	•													•				
SMSBYBKV									•													
SMSCACHE		•		•	•				•									•				
SMSCPROC	•	•	•	•	•				•	•					•		•	•				
SMSDPROC	•	•	•		•				•	•					•			•				

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
SMSFORCE	•	•	•	•	•	•			•	•					•		•					
SMSGBRST		•	•			•			•	•												
SMSIPROC	•	•	•		•				•	•					•			•				
SMSLOWRC		•	•																			
SMSMCBYP		•	•						•	•					•					•		
SMSSGBYP		•	•			•														•		
SMSSTGAD		•	•																			
SMSUPDAT				•	•																	
SORTPADD	•										•											
SPACECEX				•	•		•				•							•				
SPACEPRI		•	•							•	•											
SPACESEC		•	•							•	•											
SPFAUNIT																						•
SPFDSNAM	•																					•
SPFENQQN	•																					•
SPFENQTY	•																					•
SPFENQWT		•	•	•	•		•															
SPFHDGEX	•																					
SPFHILVN																						•
SPFORDEX																						•
SPFORSEX																						•
SPFORSPC																						•
SPFORSPE																						•
SPFRPTDT																						•
SPFSYSIN																						•
SPFUSRID																						•
SVCNODMS		•	•	•	•		•					•			•	•	•	•		•	•	
SVDCATLG	•																					
SYSOUTEX	•	•	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	•	•	•
SYSPARMO	•	•	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	•	•	•
SYSPARMS	•	•	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	•	•	•
TABLESDD		•	•		•																	
TAOPPOEX		•	•			•					•							•				

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
TAOPPREX		•	•			•					•							•				
TAPCTLEX									•	•	•	•	•									
TAPEDLEX									•	•												
TAPEFEET		•	•							•	•											
TAPEFSCR		•	•			•				•	•											
TAPEPOOL		•	•			•			•	•	•									•		
TAPEPROT		•	•			•				•	•											
TAPEPRTR		•	•							•												
TAPEVVER		•	•			•				•												
TAPMFEET																•						
TERMCLER																				•		
TMSCTLEX									•	•	•		•									
TOPSALWZ	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•	•		•	•
TOPSFORC	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•	•		•	•
TOPSSUPP	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•	•		•	•
TPMGMTEX									•	•	•	•	•									
TPOOLCHG																						•
TPOOLSEQ		•	•						•	•	•											•
TRKOVCHG																						
TSCHKAEX	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•	•		•		•
TSCHKPEX	•	•	•	•	•	•	•	•	•	•	•			•	•	•	•		•		•	•
TSODACAT																				•		•
TSODARES																				•		•
TSODASCR																				•		•
TSOTULMT																				•		
TSOTUNIT																				•		
TSOUSRID																				•		
TSOVOLOF																				•		
TTRULMOV																		•				
UCBALLOC	•	•	•	•	•		•			•	•	•			•	•	•	•		•	•	•
UCC1BYPS									•	•		•	•									
UCC1EXPR									•	•		•	•									
UCC1EXTN									•	•		•	•									

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
UCC1SCRT									•	•		•	•									
UCC6CPDS	•	•	•														•	•				
UNCATARC									•	•												
UNCATDEL		•	•			•		•	•	•												
UNCATFDR									•	•												
UNCATPSU				•					•	•												
UNLOADEX											•			•								
UNMOVEAR		•	•	•	•		•															
UNMOVEVC																		•				
USERSENQ		•	•	•	•	•	•								•	•	•	•				
USERVENQ						•																
USRCDLEN																					•	
USRCODEX																					•	
USEDSSIO		•			•	•																
VBRCHCAT		•	•		•																	
VBRIVSAM		•	•		•																	
VCBYPASS																		•				
VCCHGBIT																		•				
VCCOPYEX																		•				
VCDMIGEX																		•				
VCUDSORG																		•				
VOLSELRE	•	•	•			•									•			•			•	
VSACCESS		•	•																			
VSACESEX		•	•																			
VSAIXDEF				•	•		•				•											
VSALLOEX				•	•		•				•											
VSALOSHR		•	•	•	•		•				•											
VSAMSUPP		•	•																			
VSAPPEND				•	•		•				•											
VSARCAIX		•	•																			
VSARCFMT		•	•	•	•		•				•											
VSATTREX				•	•		•				•											
VSATTRIB				•	•		•				•											

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
VSBIXCAT				•	•		•				•											
VSBIXPSW				•	•		•				•											
VSBLDINX				•	•		•				•											
VSCATMAX																						
VSCATPTN																						
VSCIACAT	•																					
VSCIADET	•																					
VSCOMPNM	•																					
VSCRIMAX																						
VSDATEFT				•	•		•															
VSDATSUF				•	•		•											•				
VSDEFCAT				•	•		•															
VSDSPACE	•	•	•															•				
VSIDCUTP				•	•		•															
VSIDCUTV				•	•		•															
VSINXSUF				•	•		•															
VSMAFTEX	•																					
VSMPRIEX	•																					
VSMPURGE		•	•	•	•		•															
VSMSGEMP		•	•																			
VSMSGTOT		•	•																			
VSMSGWRN		•	•																			
VSMSPWEX	•	•	•	•	•		•															
VSMSTATS																						
VSNEWNEX				•	•		•															
VSNOERAS				•	•		•															
VSNOSCRO				•	•		•															
VSONLINE																						
VSPREDEF				•	•		•															
VSRESCLD				•	•		•															
VSSEQCNT				•	•		•															
VSSEQERR		•	•																			
VSSPEEDO				•	•		•															

TABLE 24. System Parameters (Continued)

Sysparm	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	XCOPY	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF
VSSPUSED				•	•		•											•				
VSSTUMPL	•	•	•	•	•		•															
VSSYSVER		•	•																			
VSUNOPEN	•	•			•	•													•			
VSUSEBLK	•																					
VSUSEMOD	•																					
VSVERIFY		•	•												•							
VTOCREAD	•						•								•	•				•	•	
WTORESRL																•						
XCPARCEX											•											
XCPDSNEX											•											

System Parameter Definitions

To place in effect a value other than the default value for any of the following system parameters, create a one-line entry in member SYSPARMS of the parmlib data set, enter the name of the sysparm and append the appropriate value.

ACF2FORCn - This sysparm is active only for users who have specified sysparm ACF2SUPP with a value of Y. The default value is N, which will cause CA-Disk to abend if CA-ACF2 is not active. Specify a value of Y to cause CA-Disk to function even if CA-ACF2 is not active.

ACF2SUPPn - This sysparm allows you to activate the CA-Disk ACF2 security interface. The default value is N, which causes CA-Disk to not query CA-ACF2 for any reason. Specify this sysparm with a value of Y to activate the security interface. When the ACF2 Security Interface is activated, CA-Disk will query the ACF2 security system before it processes certain resources. The return code that ACF2 sets will cause CA-Disk to either accept or reject that processing.

See the topic "CA-Disk Authority Level Translation Tables for ACF2" on page 319 for information on which resources are protected by the interface. See the topic "Installing CA-Disk Under CA-ACF2" on page 26 in the Installation Guide for other ACF2 options within CA-Disk.

ADPCTOTHnnn - Specify a numeric value from 000 to 100 to represent the percent increase to be added to the CA-Disk-computed space value when allocating a non-PDS data set to a different device type than it was archived from. This percentage value is added after the CA-Disk-computed value, allowing for device type changes and any other command parameters that apply to space conversion. The default value is 000.

Note: Not available when using the DFSMSdss Data Mover interface.

ADPCTPDSnnn - Specify a numeric value from 000 to 100 to represent the percent increase to be added to the CA-Disk-computed space value when allocating a PDS data set to a different device type than it was archived from. This percentage value is added after the CA-Disk-computed value, allowing for device type changes and any other command parameters that apply to space conversion. The default value is 000.

Note: Not available when using the DFSMSdss Data Mover interface.

ADTRACKSnnnn - Specify a numeric value from 0000 to 9999 to represent the number of tracks to be added to the CA-Disk-computed space value when allocating both VSAM and non-VSAM data sets to a different device type than they were archived from. For a VSAM KSDS, the tracks will be added to both the data and index components. The default value is 0000.

Note: Not available when using the DFSMSdss Data Mover interface.

ALLOCPRIy - When a data set is archived (or backed up), a “space allocated” value is saved in the DSNINDEX record and can subsequently be displayed. This sysparm indicates whether the size of the first extent (primary allocation value) or the sum of all extents (total allocation) is to be saved. The default value for this sysparm is Y (save the primary allocation). A value of N causes the total allocation to be saved.

Several of the CA-Disk reports have both the PRIM field (first extent value) and an ALLOC field (total allocation value). Setting this sysparm to N will cause the PRIM field to be equivalent to the ALLOC field, which is probably not very useful. Therefore it is recommended that a value of N be used only for backup or archive processing.

ALLOEXITmmmmmmmm - Specifying this sysparm causes CA-Disk to call a user exit. The module named by “mmmmmmmm” is given control at data set allocation time during restore or recover processing. The format-1 DSCB of the data set to be allocated is passed to the module to allow the CREDT, EXPDT, last used date, last MOD date, or open count fields to be reset. See the user exit description for “ALLOEXIT — Restore/Recover Data Set Allocation” on page 206 for more information.

ALLOVBLKn - Specify this sysparm with a value of Y to allow data sets with no track overflow and a blocksize larger than the track capacity to be allocated on the volume. The default is N, which causes CA-Disk to not allocate the data sets.

Note: Not available when using the DFSMSdss Data Mover interface.

ALOCATEXmmmmmmmm - Specify this sysparm with the module name to be invoked for the ALOCATEX user exit. See the user exit description for “ALOCATEX — Data Set Allocation Preprocessing” on page 207 for more information.

ALOPOSEXmmmmmmmm - Specify this sysparm with the module name to be invoked for the ALOPOSEX user exit. See the user exit description for “ALOPOSEX — Data Set Allocation Post-Processing” on page 207 for more information.

ALTTRKASy - This sysparm controls the assignment of alternate tracks if an error is encountered during a write to disk operation. The default value for this sysparm is Y, which allows alternate tracks to be assigned. Specify a value of N to prevent assignment of alternate tracks.

Note: Not available when using the DFSMSdss Data Mover interface.

ARBACKUPy - This sysparm allows you to determine which DSNINDEX records can be candidates for autorestore. This sysparm can be useful in 2 situations:

1. If you use the MFILES DD statement and considerations regarding order of concatenation is important
2. Whether backup and archive copies are kept in separate files data sets

There are 3 possible values for this sysparm:

TABLE 25. Possible Values for ARBACKUP

Value	Definition
Y	This default value allows the first DSNINDEX record found to be used. Users who ensure the latest copy of a data set will be found first in the MFILES concatenation will normally use this setting. Users who use the DSCL DELETE DISP=RECAT command can also see benefit in this setting.
N	Will only allow archive copies to be considered, and will ignore or skip any backup DSNINDEX records as MFILES are searched.
1	Will scan for the first DSNINDEX record found, but will insist that first record ALSO be an archive copy. If it is not, the restore will fail. This ensures that only the latest archive copy will be used when MFILES are expected to have archive copies found before any backups. This might be the case in shops emulating DFHSM or similar processing.

Note: This sysparm will add a TIMEDATE= parameter to the generated autorestore command, and this TIMEDATE will be used for Immediate or Deferred requests. In previous releases of CA-Disk, DISPOSE would use the first available DSNINDEX at the time the DISPOSE job executed.

ARCATOPTu - The default value of this sysparm is U, which causes catalog action in batch explicit archive (not retention control or VSAM implicit processing) to uncatalog the data set. Specify a value of R to recatalog to the pseudo-volume serial ARCIVE. Specify a value of N to use no-catalog action. Any of these options can be overridden by the RECATALOG, UNCATALOG or NOCATALOG parameters on the ARCHIVE command.

ARCBLKSIn - The default value of this sysparm is N, which causes CA-Disk to bypass processing for data sets with zero blocksize. Specify a value of Y to allow data sets with zero blocksize to be archived. They will be processed as physical sequential data sets. For data sets with no extents (model DSCBs), sysparm ARCMODEL will take precedence, even when the blocksize is zero.

ARCCNAMEnnn - If MVS dynamic allocation is used to allocate the //ARCHIVEC dd statement (for the duplicate copy being written to the archives), this sysparm provides the data set name to be used. The nnn value is the 1- to 22-character base name, the complete name or the tape GDG index, depending on installation JCL and other sysparm options. The default value is DMS.ARCHCOPY. See “Archive/Backup Considerations” on page 57 for more information on the generation of this data set name.

ARCCPYDVnnnn - Use this SYSPARM when making copies of Archive Tapes to determine how ARCFLAG2 and ARCFLAG4 will be copied from the original ARCHVOL. The options for SYSPARM ARCCPYDV are as follows:

1. If "N" (the default) is used , the low order five bits will be copied from ARCFLAG2, the high order three bits zeroed. If it is a mass storage device the ARCDYN1 bit is set. ARCFLAG4 will have its 4 low order bits set to the binary value of the DYN number determined by the existing logic.
2. If "Y" is used, the copied ARCHVOL has the same ARCFLAG2 and ARCFLAG4 as the original.

3. If "DYN1 " to "DYN9" is used, the low order five bits are copied from ARCFLAG2, the high order three bits zeroed, and the corresponding ARCDYN1, ARCDYN2, or ARCDYN3 bit are set. ARCFLAG4 has its 4 low order bits set to the binary value of the DYN number selected.

ARCCTYPEtape - If the //ARCHIVEC dd statement is not provided in the JCL, this sysparm indicates the type of device to be dynamically allocated. It must specify one of the following: TAPE, 3480, DISK, DYN1 through DYN9 or NULL. TAPE is the default. This feature is for MVS users only.

ARCDSKBZbbbbb - CA-Disk assigns an optimal blocksize to an archive data set being written to disk, based upon the disk device type. To override the computed value, specify the desired blocksize for the bbbbb value. The default of 00000 indicates that the computed value is to be used.

ARCDSNEXmmmmmmmm - This sysparm indicates the module to be given control after a CA-Disk archive data set name is generated. See the user exit description for "ARCDSNEX — Archive Data Set Name Generation Exit" on page 208 for more information.

ARCDSORGn - Specify this sysparm to allow data sets that have no DSORG specified in the format-1 DSCB to be eligible for processing by either archive or backup functions. The default is N, which causes CA-Disk to bypass data sets that have no valid DSORG specification.

ARCEMPTYy - Specify this sysparm with a value of N to suppress the archival of empty data sets. The default is Y, which causes CA-Disk to archive empty data sets if they are selected for processing.

ARCEODEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the ARCEODEX user exit. See the user exit description for "ARCEODEX — Archive Tape End Of File (CLOSE) Exit" on page 208 for more information.

ARCFDEFRn - Specify this sysparm with a value of Y to queue a deferred archive request for a data set that would otherwise be bypassed when an enqueue cannot be obtained. The default value is N, which causes CA-Disk to bypass the data set when an enqueue cannot be obtained. This sysparm is not intended to requeue requests during dispose processing.

ARCHIVEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the ARCHIVEX user exit. See the user exit description for "ARCHIVEX — Process Archive Request" on page 209 for more information.

ARCHPACKy - Specify this sysparm to control archive data set packing. The default value is Y, which causes each archive data set to contain multiple data sets. A value of N causes CA-Disk to create one archive data set for each data set processed. This sysparm controls data set packing for both the merge and archive functions.

Warning: For archival to tape or cartridge, do not specify a value of N, or every archive tape or cartridge will contain only one data set.

For further discussion of this sysparm, see "Using Archive Functions to Manage Active Data" on page 71.

ARCHSORTudv - Specify this sysparm to change the selection of archive reports produced. The default value of UDV will cause archive reports to be sorted by different fields as described below:

TABLE 26. Possible Values Available for ARCHSORT

Value	Description
U	print unsorted archive report
D	print archive report in dsn/volume sequence
V	print archive report in volume/dsn sequence
N	Specify N for any report not desired

ARCMODELn - Model DSCBs can be archived (and restored) during archive processing if this sysparm is specified with a value of Y. The default value is N.

ARCMULTIn - Specify a value of Y for this sysparm if multivolume PS and DA data sets are to be processed by explicit archive requests processed by the ARCHIVE proc. The default value is N, not to process multivolume data sets.

ARCNOSTDn - This sysparm eliminates messages from MSGPRINT. During BACKUP or ARCHIVE, default processing issues messages 0435 and/or 3990 for each non-standard PS data set processed. Specifying a value of Y will override this process and suppress these messages from appearing in MSGPRINT. The default value of N deactivates this feature.

For related information, see "Physical Sequential Data Sets that have Non-Standard Formats" on page 207 in the User's Guide.

ARCPLENQe - This sysparm allows the DASD Administrator to control what type of enqueue CA-Disk is to issue while searching for available space during an archive to disk job. The default value of `E' will issue an EXCLUSIVE enqueue on Qname=DMSOS, Rname=ARCHPOOL and ensures that disk archive processes will complete primary and copy allocation before allowing other concurrent archives to do space checks or allocations.

Specifying a value of `S' will issue a SHARED enqueue and will not prevent concurrent tasks from seeing the same volume as the `best' choice. Choosing this value could cause X37 abends in concurrent archive jobs. However, specifying 'S' can be useful as a trade-off if a large number of small archive requests is causing wait contention on the 'archpool' resource name.

ARCRETPDnnnnn - This sysparm controls the number of days that an explicitly archived data set is to be retained in the archives. The default value is 00030. You can specify any 5-digit value. If the user has specified the RETPD parameter on the ARCHIVE command, its value will override the one specified here.

ARCSEPC0y - When both the primary and the duplicate copy archive data sets are being written to disk, the default value for this sysparm is Y, which instructs CA-Disk to always separate the copy from the primary; that is, never put the backup copy on the same disk volume as the primary. Specifying N (no separation) permits allocation to the same volume.

ARCTNAMEd - By specifying one of the 4 values for this parameter, you can tell CA-Disk how to generate archive tape data set names, and whether or not they are to be cataloged.

TABLE 27. Possible Values Available for ARCTNAME

Value	Meaning
D	generates a unique tape data set name but not catalog it (default value)
C	generates a unique tape data set name and catalog it
G	will use GDG support to generate a unique tape data set name and catalog it
U	uses data set name specified in JCL, or the exact value of sysparm ARCxNAME

Options D (default value) or C are preferable to G or U and should be used wherever possible. These options generate unique data set names for each tape data set created. They provide significantly enhanced flexibility if you are using tape management programs such as CA1, and eliminate data set name contention during allocation. Option C also catalogs the tape data set names. For more information about the benefits of using these options, see "Cataloging the Tape Archives" on page 44 and "Naming the Archives" on page 61.

Setting this sysparm to a value of G provides the benefits of generating unique tape data set names, requires building of GDG indexes, and is limited to 255 generations per index. It is maintained strictly to provide compatibility with prior releases.

Setting this sysparm to a value of U forces CA-Disk to create output archvols based on either:

1. The value specified for DSN= in JCL
2. The value specified for ARCxNAME

The value of U is not allowed in Merge due to allocation contention if the same dsname were to be mounted as both input and output. Its also not permitted in Archive/Backup if the same dsname is specified for both ARCxNAMEs.

If you have specified either a value of C or G, you should also specify sysparm UNCATARC with a value of Y to tell CA-Disk to uncatalog the archive data set when it expires the volume.

If your security system protects tape volumes and you want to specify options C or G for this sysparm, you must also ensure that all users running backup or archive jobs have access to tape volume DMSTMP for the following reason.

To serialize tape names, CA-Disk will temporarily catalog the data set name to the fictitious volume DMSTMP. CA-Disk will then open the tape, determine the actual tape volume that was mounted, and then add it to the catalog entry, replacing the temporary value of DMSTMP. For this catalog action to take place correctly, the user running CA-Disk must have authority to tape volume DMSTMP.

ARCVOLID@#$+-%&[- When archival to disk is being used, the key to the ARCHVOLS record is generated by CA-Disk. Each key is built by taking the first character of this sysparm and appending 99999 to it. Each successive key is decremented by 1 to make it unique. When 00000 is reached, the second character from this sysparm is used and the process started over.

The default value for this parameter is @#$+-%&[. You can change the order, or specify other special characters. However, you must not specify any characters that are used in volume serial numbers at your data center, or any of the CA-Disk pattern characters (*,?,/,!). Valid characters are:

```
[ + ] & $ ; ^ - | % _ : # @ ~ \
```

ARC0NAMEnnn - If MVS dynamic allocation is used to allocate the //ARCHIVE0 dd statement (for the primary copy being written to the archives), this sysparm provides the data set name to be used. The nnn value is either the 1- to 22-character base name, the complete name or the tape GDG index, depending on installation JCL and other sysparm options. The default value is DMS.ARCHPRIM. See "Archive/Backup Considerations" on page 57 for more information on the generation of this data set name.

ARC0TYPEtape - If the //ARCHIVE0 dd statement is not provided in the JCL, this sysparm indicates the type of device to be dynamically allocated. It must specify one of the following: TAPE, 3480, DISK, DYN1 through DYN9. TAPE is the default. This feature is for MVS users only.

ARDDNEXC_ddname - This sysparm is intended to assist with the IVP process of the test auto-restore function (TSTAR). The default value is _DDNAME. Specify the DD statement name you want to invoke TSTAR. Jobs displaying this DD statement will have explicit access to the TSTAR started task. For related information, see the sysparm description for "ARJOBEXC_jobname" on page 129.

ARDIAGNM_jobname - Debugging purposes of the auto-restore function and is inactivated by default, which provides no diagnostic information. The default value is _JOBNAME. Specify the jobname in which you want to invoke these diagnostics.

ARDSNPOLxxxxxxxx - Use this sysparm to identify to CA-Disk the name of the member in your parmlib that contains the data set name pool for auto-restores. If you choose to use pool support for Auto-Restores, we recommend that you use the name ARDSNPOL for your member. To specify this, code the sysparm as ARDSNPOLARDSNPOL. See the topic "The Auto-Restore Function" on page 58 in the Installation Guide for more information regarding pool support.

ARESCRDTn - This parameter, which is valid for Auto-Restore only, allows selection of the date to be used for the data set catalog and format-1 creation date. This parameter provides the same capability as the CREATE parameter does for Restore. The default value of N sets the catalog and format-1 creation dates to the original date. Specifying a value of Y sets the catalog and format-1 creation dates to the current date.

ARESPREXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the ARESPREX user exit. See the user exit description for "ARESPREX — Auto-Restore Screening Exit" on page 210 for more information.

ARESTENQy - Specify 'Y' or 'N' to indicate if DMSAR is to perform data set enqueues or not, respectively. The default value is 'Y'. Such enqueues have QNAME='SYSDSN' and RNAME=dsname.

CAUTION: The SYSDSN enqueue is MVS' way of serializing access to a data set. Disabling SYSDSN enqueues is usually not advisable. However, some installations can want to do this in some unusual circumstances.

ARESUNICxxxxxxxx - Specify for "xxxxxxxx" the 1- to 8-character unit name for the class of 3480 devices to be used for dynamic allocation of CA-Disk archive cartridges in the auto-restore function. The default unit name is "3480". A specific unit address can also be supplied to force all auto-restores from cartridges to be performed on a single 3480 device.

ARESUNITxxxxxxxx - Specify for "xxxxxxxx" the 1- to 8-character unit name for the class of tape devices to be used for dynamic allocation of CA-Disk archive tapes in the auto- restore function. The default unit name is TAPE. A specific unit address can also be supplied to force all auto-restores from tapes to be performed on a single tape device.

ARESUNInxxxxxxxx - Specify a 1- to 8-character unit name associated with special device type DYN1 (2 through 9). This unit name is used for dynamic allocation of archive volumes during auto-restore whenever the DYN1 (2 through 9) flag is on in the ARCHVOLS record. (This flag will have been set by prior archive or merge

runs that created the archive volume specifying DYN1 (2 through 9) for sysparm ARC0TYPE, ARCCTYPE, MERPnTYP, OR MERCnTYP.) A specific unit address or an esoteric name that maps only to the special devices is recommended as a value for these sysparms. They default to unit names DYN1, through DYN9, which are probably invalid for your installation, and therefore need to be changed to the unit names as generated for your system.

Note: An intended use is to relate MASSTOR M860, Redwood, or Magstar devices, which can appear to be tape, 3480, 3490, or 3590 devices to the operating system, to the special unit names that are required to obtain proper device allocation.

These sysparms are identical in purpose to DYN1UNIT, DYN2UNIT and DYN3UNIT, but are used exclusively by the auto-restore function.

ARJOBEXC_jobname - This sysparm is intended to assist with the IVP process of the test auto-restore function (TSTAR). The default value is _JOBNAME. Specify the job name or task name (i.e., userid) you want invoke TSTAR. All submissions of jobs or tasks by this name will have explicit access to the TSTAR started task. For related information, see the sysparm description for "ARDDNEXC_ddname" on page 128.

ARPOOLEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the ARPOOLEX user exit. See the user exit description for "ARPOOLEX — Auto-Restore DASD Pool Exit" on page 210 for more information.

ARRESDELn - This sysparm controls whether an IDCAMS 'DELETE dsname NOSCRATCH' will cause the data set being scratched to be restored prior to it being uncataloged. The value 'Y' restores the data set prior to being uncataloged exactly as DFHSM does and as it is documented in the IBM manual, Access Method Services for the Integrated Catalog Facilities. The value 'N' prevents the data set from being restored and uncatalogs it.

ARSECUREn - This sysparm allows you to specify the level of access a user must have to restore a data set using the CA-Disk auto-restore function, or to queue a data set using the DRESTORE command. It can be used only if your installation has installed CA-Top Secret or RACF, and has also installed and activated the corresponding CA-Disk security interface. It does not apply to CA-ACF2 users.

Since the CA-Disk auto-restore started task cannot be identified by a job statement or a LOGON procedure, most security packages have a Started Procedures Table to associate started task names with user IDs and possible group names. Most users with security packages supply an entry in this table to associate the CA-Disk auto-restore started task name with a user ID with unlimited access to data sets in your shop. However, these users will find that any system user can auto-restore any user data sets in the CA-Disk archives. Although this causes no security exposure (after an auto-restore completes, OPEN, SCRATCH, RENAME or IDCAMS processing prevents user access to data sets to which they are not authorized), this can cause confusion. This is the case if the default value, N, is in effect.

Specify this sysparm with a value of Y to have the CA-Disk auto-restore function convert the user ID (and group name, if it exists) of the CA-Disk auto-restore started task to that of the originating user. This means that to auto-restore a user data set in the CA-Disk archives, a user would need the same access to the data set as would be required by a batch restore run by the same user: ALLOCATE, and open for OUTPUT.

Specify this sysparm with a value of R to have the CA-Disk auto-restore function convert the user ID (and group name, if it exists) of the CA-Disk auto-restore started task to that of the originating user and check for INPUT authority. If the user is authorized, CA-Disk will convert back to the original user ID (and group name, if it exists). If the entry in the Started Procedures Table gives sufficient authority, the auto-restore task can then allocate the data set and open it for OUTPUT (to put the data in), even if the originating user has only INPUT authority. This means that to auto-restore a user data set in the CA-Disk archives, a user would need INPUT authority to the data set.

For DRESTORE, this sysparm allows you to determine what authority level a user needs to queue the request. A value of 'Y' requires you to have the authority as specified in the SECURDRS sysparm. A value of 'N' or 'R', in conjunction with the SYSCTLG parameter, will only require that the user has INPUT authority. The userid associated with the DISPOSE task will be used by the security subsystem to determine if the data set can actually be restored.

For additional considerations regarding DRESTORE and security, refer to "SYSCTLG" on page 257 in the User's Guide and "DRESTORE — Deferred Restore" on page 483 in the User's Guide.

ARTAPEOKy - The default value of Y allows tape mounts to be requested for the auto-restore facility. A value of N disallows all auto-restore requests that would require a tape mount. A value of B allows tape mounts only if a batch program invoked auto- restore — it would reject any request from a TSO user. A value of D allows tape mounts for both batch and TSO users, but changes immediate restore requests from TSO to deferred requests if a tape mount is needed.

Note: This parameter has no effect on data sets that have been archived to disk. They can always be restored unless the user screening exit specifically rejects the request.

ARTASKNMdmsar - This sysparm indicates the started task name to be used by the Auto-Restore facility. The default value is DMSAR. The value of this sysparm takes effect the moment the started task is installed. If you want to use a different name for Auto-Restore, update the value of this sysparm to reflect the new name, and issue a REFRESH.

The syntax of the REFRESH command is as follows:

```
S DMSAR,DMSAR=REFRESH
```

ARVOLPOLxxxxxxxx - Use this sysparm to identify to CA-Disk the name of the member in your parmlib data set that contains the source volume pool for auto-restores. If you choose to use pool support for auto-restores, we recommend that you use the name ARVOLPOL for your member. To specify this, code this sysparm as ARVOLPOLARVOLPOL. See the topic "The Auto-Restore Function" on page 58 in the Installation Guide for more information regarding pool support.

CARTCALCn - This system parameter instructs CA-Disk as to which method to use when determining cartridge capacities.

The default value of N uses a manual calculation. For every output ARCHVOL on cartridge, CA-Disk factors in the value specified for CARTFEET, along with the number of bytes, blocks, block size, density (fixed value at 38000 BPI), and the size of inter-record gaps to determine the cartridge capacity. The result is then added to the number of bytes and blocks previously written to the ARCHVOL. If the data set will fit, CA-Disk begins transferring the data. If the data set will not fit, CA-Disk closes the ARCHVOL and begins transferring the data onto a new ARCHVOL.

Special Considerations when setting CARTCALC to "N"

1. No attempt is made to recognize differences in cartridges and/or device types. If your data center changes device types, cartridge brands, or begins to use software and/or hardware compression (IDRC), you need to revisit your setting for CARTFEET. An incorrect setting can under utilize the capacity of your cartridges.
2. When using IDRC, the actual compression rate will vary because different data compresses differently. Consider changing CARTFEET to 9999 for maximum tape utilization. Any other value will result in wasted space. You can also want to save on CPU usage by disabling software compression. See the sysparm description for "DCDATACPn" on page 136 for details.

3. Do not use StorageTek's Redwood or IBM's Magstar devices in conjunction with this sysparm set to N. Even setting CARTFEET to 9999 will result in wasted space on these devices because CARTFEET only supports devices of up to 4GB capacities.

Specify this sysparm with a value of Y to use a more automated calculation. For every output ARCHVOL on cartridge, CA-Disk obtains attributes from the mounted cartridge itself. IDRC compression is factored in allowing for greater precision in predicting utilization. This information is stored in an internal table that associates specific attributes to a given cartridge. If necessary, these values can be adjusted through the use of CARTPRCT.

Special Considerations when setting CARTCALC to "Y"

1. When using IDRC, you can want to save on CPU usage by disabling software compression. In many cases, disabling software compression allows IDRC to achieve higher compression rates. See the sysparm description for "DCDATACPn" on page 136 for details on disabling software compression.
2. CARTFEET is ignored in all cases, with the exception of:
   - If the attributes did not map into a given table entry, the value of CARTFEET will be used to determine the capacity of the cartridge
   - During Merge processing, when the utilization of an ARCHVOL is measured via the PERCENT= parameter (refer to "PERCENT=" on page 349 in the User's Guide for details) and the input ARCHVOL was created prior to release 9.0, or when CARTCALCN is specified, the value of CARTFEET will be used in this situation to determine which cartridges meet the selection percentages.
   - There was no way of dynamically determining the capacity of the mounted cartridges for ARCHVOLS created prior to release 9.0. In this situation, the value of CARTFEET is used to determine the capacity of the cartridge. However, if Y is specified for sysparm CARTCALC, new ARCHVOLS will contain the appropriate information.
3. StorageTek's Redwood and IBM's Magstar devices must be specified as a DYNn unit. See "Using Multiple Types of Media" on page 60 for details.

CARTFEETnnnn - The default value for this sysparm is 0494, approximately the uncompressed size of a 200MB 3480 cartridge.

This sysparm is closely associated with CARTCALC. When CARTCALC is specified with a value of N, CA-Disk factors in the value specified for this sysparm, along with the number of bytes, blocks, block size, density (fixed value at 38000 BPI), and the size of inter-record gaps to determine the cartridge capacity. The result is then added to the number of bytes and blocks previously written to the ARCHVOL. If the data set will fit, CA-Disk begins transferring the data. If the data set will not fit, CA-Disk closes the ARCHVOL and begins transferring the data onto a new ARCHVOL.

CARTPRCTnnn - This sysparm is closely associated with CARTCALC. When CARTCALC is specified with a value of Y, CA-Disk obtains attributes from the mounted cartridge itself. This information is stored in an internal table that associates specific attributes to a given cartridge. If necessary, these values can be adjusted through the use of this sysparm.

This parameter is simply a percentage value relative to 100 that allows you to customize cartridge capacities. This is useful in cases where variations from different cartridge vendors result in substantial differences in cartridge lengths. When CA-Disk factors this percentage into the calculation derived from CARTCALC, CA-Disk adjusts the cartridge capacity accordingly.

Values above 100 force CA-Disk to assume that all cartridges are larger (i.e., a value of 125 indicates that cartridges are 25% larger than the information in the CA-Disk internal tables). Values less than 100 indicate cartridges will hold less data than the information contained in the CA-Disk internal tables.

CATBKDELn - When an ICF catalog is being recovered but the “damaged” catalog is still on disk, the damaged version will be overlaid only if its “export temporary” flag is on. If this flag is not on, the catalog must be deleted before it can be recovered. The default value of N for this sysparm causes CA-Disk to bypass the recovery under this condition; that is, CA-Disk will not delete the catalog. It must be deleted manually before rerunning the recovery.

Specify the sysparm with a value of Y to allow CA-Disk to delete an existing ICF catalog in order to continue with the recovery.

CATBKNAMnnnnnnnnnnnnnnnnnnn - When EXPORTF=YES has been specified for recovering an ICF catalog, the CA-Disk backup copy is written to a sequential data set on disk that you can then use as direct input to a native IDCAMS IMPORT. This sysparm specifies a base name for CA-Disk to use to generate the data set name when allocating the sequential data set. The base name can be from 1 to 19 characters in length. CA-Disk appends “EXPTCAT” and the date and time to the base name to make it unique.

The default value is DMS.CATEXPT, which results in the name “DMS.CATEXPT.EXPTCAT.Dxxxxxxx.Txxxxxxx”. As an example, if this sysparm were specified with the value “AAA.BBB.”, the resulting name would be “AAA.BBB.EXPTCAT.Dxxxxxxx.Txxxxxxx”.

CATBKSPC500 - When EXPORTF=YES has been specified for recovering an ICF catalog, the CA-Disk backup copy is written to a sequential data set on disk that you can then use as direct input to a native IDCAMS IMPORT. This sysparm specifies the number of blocks to allocate for the sequential data set. Both the primary and secondary quantities are allocated in blocks, each block being 2048 bytes. The default value is 500, the maximum value is 9999.

CATBKUNTunitname - When EXPORTF=YES has been specified for recovering an ICF catalog, the CA-Disk backup copy is written to a sequential data set on disk that you can then use as direct input to a native IDCAMS IMPORT. This sysparm specifies the unitname CA-Disk is to use when allocating the sequential data set. The default is SYSALLDA.

CATLOCWAnnnnnnn - This sysparm indicates the maximum number of bytes of memory to provide to Catalog Management for catalog work space when CA-Disk backs up ICF catalogs, creates a selection list of cataloged data sets, or processes a FIND command with a DSN parameter containing a CA-Disk pattern name.

CA-Disk will start with a work area of 20,000 bytes. If the initial work area is not large enough for Catalog Management, CA-Disk will create a larger work area based on the requirements returned by Catalog Management, then retry the request.

CA-Disk will not create a work area larger than the size specified by this sysparm. The default value of 0262144 allows CA-Disk to use up to a quarter megabyte if necessary. You can specify any other 7-digit number consistent with the memory and region constraints at your installation.

Keep in mind that CA-Disk also needs region for programs, I/O buffers and the like, and that most installations have more restrictive region constraints on TSO or ISPF sessions than on batch jobs or started tasks.

CATPURGEn - This sysparm indicates whether or not an unexpired catalog entry can be purged. VSAM catalogs allow an expiration date to be placed in the catalog entry for non-VSAM entries as well as VSAM. An unexpired entry causes an uncatalog request to fail, unless “purge” has been requested. Specify the sysparm with a value of Y (the default is N) to cause CA-Disk archive processing to purge a catalog entry even if it is found to be unexpired.

IBM message IEC331I 084-000 will still appear in the JES log to indicate that the initial attempt failed because of the date protection. However, no CA-Disk message will be issued unless the subsequent “purge” attempt also fails.

CCDRESETn - While processing the DSCL FIND command, if CA-Disk returns the message 2793 “NO CATALOG ENTRIES WERE FOUND TO MATCH DSN = add”, a job return code of at least 8 is issued for the job. Specifying this sysparm with a value of “Y” suppresses the job condition code to 0. All other errors encountered will still cause the appropriate job condition to be reset. The default value of “N” leaves the condition code of 8 unchanged.

CCINITEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the CCINITEX user exit. See the user exit description for “CCINITEX — BACKUPCC Processing Exit” on page 211 for more information.

CNVATCYLn - This sysparm controls whether or not the data set's original allocation type (tracks or cylinders) will be used during VSAM restores. The allocation type might change when, for example, the device type changes from the original archived device, or restore parameters (CISZ, IMBED/NOIMBED, SPAN/NOSPAN or RECSZ) alter the data set's original archived values.

The default value of this sysparm is N, meaning no change will be made to the allocation type. Specifying a value of Y allows CA-Disk to determine whether or not the allocation type needs to be altered. If the calculated CA-size is less than a cylinder, the allocation will be made in tracks; if equal to a cylinder, the allocation will be made in cylinders.

Note: Not available when using the DFSMSdss Data Mover interface.

DARCHIEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the DARCHIEX user exit. See the user exit description for “DARCHIEX — Process Deferred Archive Request” on page 212 for more information.

DARCKEEPxxx - Specify for “xxx” the number of days a deferred archive request is to be kept in the ARCHCMDS file. This number applies only after the archive has been completed. This allows the results to be inspected by the TSO LISTREQ command. The default is 005 days.

DASDPMAXnnn - Specify a 1- to 3-digit value indicating the total number of pools that can be defined in the DASDPOOL member of parmlib. This value is used to determine memory requirements before building a table with the number of pools and volumes available for non-VSAM volume pooling. See also sysparm DASDVMAX. The default value is 60.

DASDVMAXnnnn - Specify a 1- to 4-digit value indicating the total number of volumes that can be included in defined DASD pools (sum of volumes in all different pools). This also represents the maximum number of volumes that can be specified for a single pool. The default value is 100.

DATECUTO75 - This sysparm instructs CA-Disk on how to treat 2 digit year values entered from command input parameters or sysparms. Its purpose is to provide the user some flexibility in testing or implementing Year 2000 strategies. The default value is 75, indicating that any 2 digit year value of 75 or less will be considered as 20YY, these dates will be stored as MM/DD/20YY. If the "YY" portion of the date is greater than 75, the dates will be stored as MM/DD/19YY. You can specify any numeric value ranging from 00 to 99.

This allows the use of 2 digits for input to continue, or you can use 4 digit notations as desired. Examples with default value of 75:

- EXPDT=01/01/00 or 00001 will be saved as 2000.001
- EXPDT=01/01/75 or 75001 will be saved as 2075.001
- EXPDT=01/01/76 or 76001 will be saved as 1976.001
- EXPDT=01/01/99 or 99001 will be saved as 1999.001

Caution: There is no 'special date' exclusion on inputs. Except for special testing needs, DO NOT set DATECUTO to values of 97, 98, or 99.

For related information, refer to "Year 2000 Considerations" on page 6 in the User's Guide.

DBDSNCFDx - Specify for "x" the index level of the data set name that is to be used as a user code. The default value is 1. This sysparm is activated only if sysparm DBDSNINX has been specified with a value of Y. This sysparm is used in DASD space billing.

DBDSNINXn - Specify a value of Y to indicate that the user code is an index level from the data set name. The default value is N, indicating that the user code is found by sysparms DSCBUSRD and USRCDLEN. This sysparm is used in DASD space billing.

DBRCDBSZ20000 - Specify this sysparm only if DBRC support has been activated through sysparm DBRCSUPP. This sysparm either indicates the number of characters of memory to provide for the data base table when using DBRC or the number of data base data set names that will be included in the data base table.

The value for this sysparm is made up of 5 characters. If the number of data base data set names is selected, an 'N' must be the first character followed by as many as 4 digits. If the memory size is selected, up to 5 digits can be supplied. No special character is required for supplying memory size.

If supplying the number of data base names, the sysparm DBRCDBSZ must be edited in the PARMEDIT member of your CA-Disk PARMLIB. The numeric check must be removed for this sysparm.

The default value for this sysparm is 20000. The maximum number value is 99999 for memory size and N9999 for number of data sets.

DBRCICSZ80000 - Specify this sysparm only if DBRC support has been activated through sysparm DBRCSUPP. This sysparm indicates the number of characters of memory to provide for the image copy table when using DBRC support. All of the image copy data set names will be kept in this work area. If the storage area is not large enough to hold all names, the job will abend with a message to increase this number. The default is 80000; the maximum value is 999999.

DBRCPREXnnnnnnnn - Specify this sysparm only if DBRC support has been activated through sysparm DBRCSUPP. Specify this sysparm to provide the module name to be invoked for the DBRCPREX user exit. See the user exit description for "DBRCPREX — DBRC Control Statement Screening" on page 213 for more information.

DBRCRLIBnnn - Specify this sysparm only if DBRC support has been activated through sysparm DBRCSUPP. This sysparm is used to specify the 1- to 44- character default name of the DBRC RESLIB library. If the RESLIB dd statement is not supplied in the job stream, this library will be used to retrieve the name of the RECON data sets. The default value is IMSVS.RESLIB.

DBRCRLSEnnn - Specify this sysparm only if you have activated DBRC support through sysparm DBRCSUPP. This sysparm is used to specify the release of IMS in use at your site. The default value is 410, for IMS Release 4.1.0. Specify this value as the IMS release number, but omit the decimal points. For more details, see "Activating the DBRC Interface" on page 48 in the Installation Guide.

DBRCSMUT3400 - Specify a 1- to 8-character unit name for the unit name parameter generated during Sequential Migrate for the DBRC interface. New releases of DBRC use a unit name parameter in the CHANGE.IC statement to identify the unit type. 3400 is the default.

DBRCSUPPn - Specify a value of C for this sysparm to activate the CA-Disk interface to DBRC. Before activating support, refer to "Activating the DBRC Interface" on page 48 in the Installation Guide. The default value is N, which indicates the support is not active.

DBRCVCUT3380 - Specify a 1- to 8-character unit name for the unit name parameter generated during Move/Copy for the DBRC interface. New releases of DBRC use a unit name parameter in the CHANGE.IC statement to identify the unit type. 3380 is the default.

DCAVGCMPnn - Specify for this sysparm the average amount of compression realized when using data compression. To determine this value, run archive or backup with sysparm DCRTSTAT specified with a value of S or D. This value, given as a percent, will be used to estimate the size of a data set while allowing for compression. As a result, a better prediction can be made regarding how many feet of tape or 3480 cartridge will be needed. The default value is 40, or 40 percent average compression.

This sysparm also affects disk ARCHVOL allocation. The requested bytes will be adjusted to the indicated compression factor. With this sysparm active, if a backup or archive command receives an SB37 abend, CA-Disk will retry the ARCHVOL allocation using uncompressed byte amounts, as if DCAVGCMP00 was specified.

DCCMPEX0ADSUT371 - This sysparm names the module to call for data compression technique number 0. For more information, see the topic "Data Compression/Decompression" on page 386. Technique number 0 is reserved for CA-Disk.

DCCMPEX1ADSUT410 - This sysparm names the module to call for data compression technique number 1. This technique is for CA-Disk users who are also licensed to use the CA-Compress data compression product, and are using it to compress the CA-Disk archives. For more information, see "Activating Data Compression" on page 57. Technique number 1 is reserved for CA-Disk.

DCCMPEXnmmmmmmmm - These sysparms name the modules to call for data compression technique numbers 2 - F. Technique numbers 2 through 7 are reserved for future CA-Disk-supplied data compression modules. Technique numbers 8 through F are for user-supplied data compression modules. See the user exit description for "DCCMPEXn — Data Compression" on page 213 for more information.

DCCOMPTCn - Specify the technique number to be used for data compression during Archive/Backup/Vbackup processing. This sysparm setting is used when sysparm DCDATACP is specified with a value of Y.

When specified, the default technique number 0 is used for data compression. For more information about how compression is done with this technique, see the topic "Data Compression/Decompression" on page 386.

Technique number 1 is for CA-Disk users who are also licensed to use the CA-Compress data compression product. See "Activating Data Compression" on page 57.

Technique number 2 is like technique 0, though not as efficient. However, it requires less CPU overhead than technique 0. Consider using technique number 2 when job performance is more important than compression efficiency.

Technique numbers 0 through 7 are reserved for CA-Disk-supplied data compression modules. Technique numbers 8 through F are for user-supplied data compression modules.

For volume-level backup functions, only techniques 0 and 2 are valid. If data compression is turned on (sysparm DCDATACP specified with a value of Y) and a different compression technique is specified, technique 2 will be used.

DCDATACPn - The default value of this sysparm is N, which causes CA-Disk to not compress data. Specify a value of Y to activate data compression. For more information, see "Activating Data Compression" on page 57.

Note: This sysparm is overridden by specification of sysparms DCEXCTBL and DCINCTBL. That is, regardless of the setting of this sysparm, if a data set is in the data compression exclusion table in parmlib, it will not be compressed. If a data set is in the data compression inclusion table, it will be compressed, even if this sysparm is specified with a value of N. See the topic "Data Compression/ Decompression" on page 386.

DCDCPEX0ADSUT372 - This sysparm names the module to call for data decompression technique number 0. For more information see the topic "Data Compression/Decompression" on page 386. Technique number 0 is reserved for CA-Disk.

DCDCPEX1ADSUT411 - This sysparm names the module to call for data decompression technique number 1. This technique is for CA-Disk users who are also licensed to use the CA-Compress data compression product, and are using it to compress the CA-Disk archives. For more information, see "Activating Data Compression" on page 57. Technique number 1 is reserved for CA-Disk.

DCDCPEXnmmmmmmmm - These sysparms name the modules to call for data decompression technique numbers 2 through F. Technique numbers 2 through 7 are reserved for future CA-Disk-supplied data decompression modules. Technique numbers 8 through F are for user-supplied data decompression modules. See the user exit description for "DCDCPEXn — Data Decompression" on page 214 for more information.

DCDSNDEXmmmmmmmm - Specify this sysparm to provide the module name to be used for screening data sets for data compression. See the user exit description for "DCDSNDEX — Screen Data Sets for Data Compression" on page 214 for more information on this user exit. For more information see the topic "Data Compression/Decompression" on page 386. Also see "Activating Data Compression" on page 57 if you are using data compression technique 1.

DCEXCTBLxxxxxxxx - Specify the CA-Disk parmlib member name for "xxxxxxxx" that contains the list of data set or pattern names not to be compressed during the archive/backup. If you are using data compression technique 1, include in this list all data sets that are under the control of the CA-Compress data compression product. For further information, see "Activating Data Compression" on page 57.

DCINCTBLxxxxxxxx - Specify the CA-Disk parmlib member name for "xxxxxxxx" that contains the list of data set or pattern names to be compressed during the archive/backup.

DCRTSTATn - The default value of N for this sysparm suppresses the informational messages about data compression. Specify a value of S to see archive volume compression totals or D to see individual data set compression totals and archive volume compression totals. See message numbers "2805" on page 165 in the Message Manual and "2806" on page 165 in the Message Manual for format.

DEFARCEXmmmmmmmm - This sysparm causes CA-Disk to call a user exit. Specify for "mmmmmmmm" the name of a user-supplied exit module that is to be given control during dispose processing of deferred (queued)

archive requests. See the user exit description for "DEFARCEX — Screen Disposal of Deferred Archive Requests" on page 214 for more information.

DELETEOKn - The default value of this sysparm is N, which causes CA-Disk to reject the DSCL DELETE command with message 2663. To activate the DSCL DELETE command, specify a value of Y.

DERASEEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the DERASEEX user exit. See the user exit description for "DERASEEX — Process Erase TSO Deferred Processing Request" on page 215 for more information.

DFLTPOOLxxxxxxxx - Specify an 8-character name to be used as the name of the default DASD pool for data set allocation. This name specifies the name of a pool defined in the DASDPOOL member of the parmlib data set. For syntax information, see "DASD Pool List" on page 481. If this sysparm is not specified or is specified as blanks, CA-Disk will not use a default pool during data set allocation. Note that the pool used in restore and recover applies to non-VSAM data sets only.

DIAG0001n - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAG0002n - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGALOCn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGAUTHn - Normally the CA-Disk Security Interfaces print messages only for exceptions to normal processing. Specify this sysparm with a value of Y to have the CA-Disk Security Interfaces print additional messages regarding actions and key decisions. The default value of N causes CA-Disk to not print these messages. Because this is a diagnostic feature, the content and format of the messages can or cannot be documented.

DIAGDCTCn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGENQDn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGENQFn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGENQVn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGEXCPn - This sysparm governs printing of special error diagnostic messages during EXCP processing. The default value is N, which suppresses the additional messages. If specified with a Y, CA-Disk will include the IOB control block, the channel program and the data transmitted or received by the CCW, with any I/O error message returned from the EXCP access method.

DIAGLSTRn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGNOSEn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGRECDn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGSHRKn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGSTAEn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGSTMPy - Debugging purposes only and is activated by default, which provides debugging information. Specifying a value of N is not recommended unless requested by Technical Support.

DIAGTHRSn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DIAGVSRLn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

DINXUFEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the DINXUFEX user exit. See the user exit description for "DINXUFEX — Update DSNINDEX User Field" on page 215 for more information.

DISKPROTy - This sysparm provides password protection for archive data sets that are not DFSMS-controlled.

The default value of this sysparm is Y, which causes CA-Disk to password-protect, but not generate passwords for, archive data sets on disk and bypass the normal security checks when opening them. CA-Disk password- protects these data sets by setting the password indicator DS1IND10 bit in the format-1 DSCB. The password indicator prevents access to archive data sets for volumes that are not DFSMS-controlled.

When this sysparm is specified with a value of N, CA-Disk does not set the password indicator.

The sysparm setting has no effect when the volumes containing the archive data sets are DFSMS-controlled because password protection is not supported in this situation. However, when the sysparm is specified with a value of N or the volumes are DFSMS-controlled, users with READ access can browse disk archive data sets. To prevent this access, specify to your security package to grant NO access for ALL users for CA-Disk archive data sets.

For password-protection of archive tapes, see sysparm "TAPEPROTn" on page 183.

DLVOLMSGy - The default value for this sysparm is Y, which causes a message to be issued when processing begins for a given volume using DSCL. Specify a value of N to suppress this message.

DRESARCHy - The default value of Y causes the volsers that a data set is archived on to be displayed in message 2401 during a deferred restore request. To determine the volsers that the data set is archived on, the ARCHVOLS subfile must be accessed for INPUT authority.

Specify this sysparm with a value of N to prevent display of the volsers that the data set is archived on and to prevent access to the ARCVHOLS subfile.

DRESKEEPxxx - Specify for "xxx" the number of days a deferred restore request is to be kept in the RESTCMDS file. This number applies only after the restore has been completed. This allows the results to be inspected by the TSO LISTREQ command. The default is 5 days.

DRESTOEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the DRESTOEX user exit. See the user exit description for "DRESTOEX — Screen Deferred Restore Request" on page 216 for more information.

DSBALTRTn - CA-Disk DASD space billing uses a decimal-place unit for accumulating charges when this sysparm is specified with the default value of N. If your installation requires that charges for DASD space be expressed in a different unit, such as for Japanese yen or Italian lire, specify this sysparm with a value of 1. This changes the billing rates assigned in the DSBRATEC member from a dollars and cents type value to a unit of currency without decimal places.

If this feature is used, specifying "993.25" as an entry in the DSBRATEC member would indicate that either 3.25 lire or 3.25 yen are to be charged for each track or kilobyte day for all volumes assigned to category code 99. The amounts and totals shown on billing reports will be shown as full numbers with no decimal places. The rate value shown on the billing reports will be shown as 5 digits with 2 decimal places.

Note: This alternate rate specification can also be used for other forms of currency as appropriate. The only valid values for this sysparm are N and 1.

DSBCATSZxxxxx - This sysparm is used in DASD space billing to specify the buffer size for the category table. The default value is 00500.

DSBSELVSn - CA-Disk provides the ability to bill at the data set level for both VSAM and non-VSAM. Specify this sysparm with a value of Y to select VSAM data spaces and unique clusters within the non-VSAM billing runs. The default of N causes CA-Disk to not to select any VSAM spaces during non-VSAM selection, on the assumption that data set (cluster) level VSAM billing will be done also.

DSBTRKSZt - Specify a value of 'M' to use the maximum data block in determining the kilobyte days for a data set for billing purposes. The default value of 'T' uses the track capacity value.

TRKSZ vs MAXBLK = 47987 vs 47476 for 3380 devices
TRKSZ vs MAXBLK = 58786 vs 56664 for 3390 devices

CA-Disk billing is using 58K for 3390 calculations. By using K/B days, the truncated value is added to the accumulation records in a subfile. By using TRK days, no truncation or rounding problem occurs.

DSBUNITSxxx - This sysparm is used in DASD space billing to specify the units used for computing billing cost.

```
K/B       — kilobyte days (default value)
TRK       — track days
```

TRKSZ vs MAXBLK = 47987 vs 47476 for 3380 devices
TRKSZ vs MAXBLK = 58786 vs 56664 for 3390 devices

CA-Disk billing is using 58K for 3390 calculations. By using K/B days, the truncated value is added to the accumulation records in a subfile. By using TRK days, no truncation or rounding problem occurs.

DSBUSREXmmmmmmmm - This sysparm causes CA-Disk to call a user exit. Specify for "mmmmmmmm" the module name to be given control on every DASD billing record insert or update. The exit is passed the function type and the billing record. See the user exit description for "DSBUSREX — Screen Billing Record Inserts" on page 217 for more information.

DSBVOLCTnnnn - This sysparm is used in DASD Space Billing to indicate the maximum number of volumes to be processed in the Extend report. The default is 300. If CA-Disk issues a message indicating this value is too small, specify a value sufficiently large to allow for some growth. The maximum value allowed is 9999.

DSCBCVEXmmmmmmmm - Specify this sysparm with the module name to be invoked when CA-Disk maintained fields must be moved to new offsets within the format-1 DSCB. This user exit is used to move the CA-Disk maintained fields during any processing which allocates a user data set on DASD. See the user exit description for "DSCBCVEX — Format-1 DSCB Conversion Exit Location" on page 217 for more information.

DSCBJBNMxxx - The displacement of the jobname in the DSCB is expressed as a three-digit number. If this field is not present, zeroes should be specified. The default is 062.

CA-Disk maintains this jobname in the OPEN SVC. It can be one of the following four values as determined in the OPEN SVC.

1. Creating jobname
2. Last modified jobname
3. Last using jobname
4. Accounting information

DSCBLMODnnnp - The displacement of the last modification date in the DSCB is expressed as a three-digit number. If the date is carried in packed instead of binary format, the letter "p" should follow. If this date is not present in the DSCB, the "nnn" field should be 000. The default displacement is 070.

DSCBLUSDnnnp - The displacement of the last used date in the DSCB is expressed as a three-digit number. If the date is carried in packed instead of binary format, the letter "p" should follow. If this date is not present in the DSCB, the "nnn" field should be 000. The default displacement is 075.

Note: This is the SU 60 compatible offset. Early CA-Disk users who have not converted to the SU 60 compatible location (that is, are still using offset 045) must specify a value of 045.

DSCBOPCDnnnh - The displacement of the data set open count is indicated as a three-digit number. If this value is carried as a halfword instead of a fullword, the letter "h" should be placed after the number. If an open count is not maintained in the DSCB, zeros should be entered. The default displacement is 073.

DSCBSVMDnnn - The CA-Disk Open SVC maintains the SVC mode field in the format-1 DSCB. This sysparm indicates the displacement of this field in the format-1 DSCB. The default offset value is 103.

DSCBUSRDxxx - This sysparm is used in DASD space billing, and is the displacement of the user code in the DSCB. The default value is 000.

DSCLMSGSy - This sysparm controls the issuance and suppression of "broken" data set error messages encountered during DSCL SCAN REALVOLS and SCAN CATALOG processing. When this sysparm is specified with the default value of Y, CA-Disk is allowed to issue the messages. When this sysparm is specified with a value of N, CA-Disk suppress the "broken" data set messages, without affecting the job step return code. Following is a list of the messages that can be issued or suppressed depending on the setting of DSCLMSGS.

```
2505 2664 3262 3273 3316 3397 3435 3436 3437 3438 3439 3440 3467
```

DSCLOWRCn - This sysparm provides the option of reducing the return code from 4 to 0 when data sets are not selected for processing. When this sysparm is specified with a value of Y, CA-Disk will lower the return code in

certain circumstances. For details on how to customize this facility, see the topic "User-Specified Condition Codes" on page 89.

Note: Another option in this area is available for those instances where "NO CATALOG ENTRIES WERE FOUND" exists when DSCL is searching for candidate data sets. See "CCDRESETn" on page 133 for details.

DSCLRPLYn - CA-Disk DSCL processing can be instructed to allow the operator to terminate any DSCL job normally through an operator reply. To activate this feature, specify Y for the value of this sysparm. The default value of N deactivates the feature.

Review the topic "Graceful Shutdown" on page 34 in the User's Guide for more details.

DSCLSCEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the DSCLSCEX user exit. See the user exit description for "DSCLSCEX — DSCL Screening Exit" on page 218 for more information.

DSDISPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the DSDISPEX user exit. The exit module is passed a format-1 DSCB and the TASC field as defined by the retention control VOLSPEC table. See the user exit description for "DSDISPEX — Post Data Set Disposition Action" on page 219 for more information.

DSNDELEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the DSNDELEX user exit. See the user exit description for "DSNDELEX — Screen Index Maintenance Deletes (Prior Exit)" on page 220 for more information.

DSNDLPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the DSNDLPEX user exit. See the user exit description for "DSNDLPEX — Screen Index Maintenance Deletes (After Exit)" on page 220 for more information.

DSNDUPVLn - This sysparm is used during IXMAINT, DSCL, and RECOVER processing. IXMAINT processing uses it when the COPIES= parameter is specified. DSCL selection processing uses it when information from the files data set (that is, ARCDATE) is needed. RECOVER processing uses it in the same way.

When this sysparm is specified with the default value of N, CA-Disk processing considers DSNINDEX records for data sets with the same name that reside on different volumes to be different data sets and handles them separately for version checking, and retrieval of files data set information.

When this sysparm is specified with a value of Y, CA-Disk processing considers them to be the same data set. This allows CA-Disk to properly process data sets as they are moved from volume to volume.

See the description of sysparm "RESCRCATn" on page 167 for related information.

Note: In an SMS environment, this sysparm should be set to a value of "Y". The reason is basically because SMS does not allow uncataloged copies. This is explained further in the footnote that accompanies "Number of Backup Versions" on page 328 in the User's Guide.

DSNSELREn - Specifying a value of Y to this sysparm will require the user to code a DSNAMES= parameter on all SELECT statements. The default value of N is equivalent to coding DSNAMES=/ on a SELECT statement which

causes CA-Disk to attempt to process ALL data sets selected by the SELECT command. This can allow unwanted processing to occur. For example, if a user unintentionally runs:

```
SCAN REALVOLS
   SELECT CRITERIA=(DSORG,EQ,VSAM)
      DELETE
```

Note: For other related sysparms, refer to the descriptions of "SELECTREy" on page 175 and "VOLSELREn" on page 190.

DSSEIOTRn - This sysparm controls whether or not CA-Disk is to trace entries into the DFSMSdss exits. The default value of N deactivates the trace. Specifying a value of Y will cause a series of WTO's to be issued.

Note: Not available when using the DFSMSdss Data Mover interface.

DSSMSGDDxxxxxxxx - Specifying this sysparm instructs CA-Disk to store a collection of DFSMSdss messages. For the value of this sysparm, specify an 1- to 8-character DDname. The default is blanks, which disables the facility.

The use of this sysparm requires that the user add a DD statement in the backup or restore jobstep equal to that of this sysparm. For example, either of the following would be valid:

```
//TESTDD   DD SYSOUT=*
   OR
//DSSMSGDD DD DSN=output.data.set,DISP=OLD
```

Note: The value you choose for this sysparm should not be any DDname that is already in use by CA-Disk.

DSUCATLGy - This sysparm can be used to suppress catalog referencing in the preparation of the Data Set Utilization report by specifying a value of N instead of the default value of Y. The default value will cause CA-Disk to do catalog locates for every data set listed on the DSU report to provide the report item indicating if the data set is cataloged or not. If this information is not needed on the report, changing the value to N will cause a significant reduction in elapsed time for report processing.

DSUTILCF0 - The one-digit number designates which index level within the data set name is to be used as the control field for the Data Set Utilization report. The default value is 0, indicating no page breaks on index level change.

DSUTILSQd - The letter N can be specified to designate that the Data Set Utilization report should appear in data set name sequence. The default value is D for date of last use sequence.

DYNALDSNn - Specify Y to use the real data set name for the tape allocation. The system issues a shared ENQ on the data set name. The default value N uses the system generated name for tape allocation. The system issues an exclusive ENQ on the data set name.

DYNCUNITxxxxxxxx - Specify for "xxxxxxxx" the 1- to 8-character unit name for the 3480 or 3490 devices to be used for dynamic allocation of CA-Disk archive cartridges needed for the archive, backup, restore, recover or merge functions. The default unit name is "3480". A specific unit address can also be supplied to force all of these functions to use a single 3480 device. For additional information, see the sysparm description for "ARESUNICxxxxxxxx" on page 128.

Note: While simulating the MERGE command and for reporting purposes only, you can specify a different cartridge density. This is only valid in simulate mode. In live mode, the actual density of the device is used.

For example, if you want to simulate other devices, change this sysparm to one of the following values:

TABLE 28. List of Possible Value for DYNCUNIT

Value	Device Simulated	Capacity
34800	3480 Cartridge	200 MB
34900	3490 Cartridge	400 MB
34901	3490/E Cartridge	800 MB

For related information, see the topic "Simulate Mode Considerations" on page 345 in the User's Guide.

DYNEXPDTE99365 - If the //ARCHIVE0 or //ARCHIVEC dd statements for the Archive/Retain functions, or the //ARCHIVE1-5 or //ARCHIVC1-5 dd statements for the Merge function, must be dynamically allocated, this sysparm provides the equivalent of specifying either the EXPDT or the RETPD value in the JCL. Specify either Eyyddd, Eyyyyddd or Rddddd, where E indicates a Julian date follows. R indicates the following 5 digits are a retention period. A value of E00000 is not valid. The default value is E99365, which has a specific meaning to CA-Disk. See "Assigning Tape Expiration Dates" on page 45 for applicable rules.

For related information, see "Year 2000 Considerations" on page 6 in the User's Guide.

DYNEXPD2E00000 - If the //ARCHIVEC dd statement for the Archive/Backup functions must be dynamically allocated when using the Dual Primary Volume Option, this sysparm provides the equivalent of specifying either the EXPDT or the RETPD value in the JCL. Specify either Eyyddd, Eyyyyddd or Rddddd, where E indicates a Julian date follows. R indicates the following 5 digits are a retention period. A value of E00000 indicates that the DYNEXPDT sysparm is to be used. The value E99365 has a specific meaning to CA-Disk. See "Assigning Tape Expiration Dates" on page 45 for applicable rules.

For related information, see "Year 2000 Considerations" on page 6 in the User's Guide.

DYNTUNITxxxxxxxx - Specify for "xxxxxxxx" the 1- to 8-character unit name for the tape devices to be used for dynamic allocation of CA-Disk archive tapes needed for the archive, backup, restore, recover or merge functions. The default unit name is TAPE. A specific unit address can also be supplied to force all of these functions to use a single tape device. For additional information, see "ARESUNITxxxxxxxx" on page 128.

Note: While simulating the MERGE command and for reporting purposes only, you can specify a different cartridge density. This is only valid in simulate mode. In live mode, the actual density of the device is used.

For instance, if you want to simulate other devices, change this sysparm to one of the following values:

TABLE 29. List of Possible Values for DYNTUNIT

Value	Device Simulated
01600	3420 at 1600 BPI
06250	3420 at 6250 BPI

For related information, see the topic "Simulate Mode Considerations" on page 345 in the User's Guide.

DYNUKEEPn - This sysparm is used with dynamic allocation in all of the archive, backup, restore, recover and merge functions to determine whether or not the tape device should be released each time a new tape volume is needed. It applies to both input and output tapes. If this sysparm is specified with a value of N, the default value, the device is freed at completion of each archvols tape. If the sysparm is specified with a value of Y, the same device is kept for the entire job step.

Keeping the same device can aid the operations staff in knowing where tape mounts will be needed, but it can also increase run time since the next tape cannot be mounted until the prior one is rewound and dismounted.

DYNVREQDn - When dynamic allocation is being used, this sysparm controls whether mount requests for output archive volumes will occur at allocation time or at OPEN time. The default value of N means the archive volumes are mounted at OPEN time. If this sysparm is specified with a value of Y (and sysparm DYNUKEEP is also specified with a value of N), archive volumes are mounted at allocation time.

Note: If you are using MASSTOR M860 or Redwood devices for your output archive volumes, you must specify this sysparm with a value of Y and DYNUKEEP with a value of N.

DYNnUNITxxxxxxxx - Specify a 1- to 8-character unit name for special device type DYN1 (through DYN9). This unit name is used for dynamic allocation of archive volumes during archive, backup, restore, recover, retain and merge functions, whenever DYN1 (through DYN9) has been specified as the device type to be used via one of the sysparms ARC0TYPE, ARCCTYPE, MERPnTYP, or MERCnTYP. When DYN1 (through DYN9) is used, a flag is set in the ARCHVOLS record and the unit name specified in DYN1UNIT (through DYN9UNIT) is used for all functions relating to that archive volume. A specific unit address or an esoteric name that maps only to the special devices is recommended as a value for these sysparms.

An intended use is to relate MASSTOR M860 or Redwood devices, which can appear to be tape, 3480, or 3490 devices to the operating system, to the special unit names that are required to obtain proper device allocation.

Note: While simulating the MERGE command and for reporting purposes only, you can specify a different cartridge density. This is only valid in simulate mode. In live mode, the actual density of the device is used.

For instance, if you want to simulate other devices, change this sysparm to one of the following values:

TABLE 30. List of Possible Value for DYNnUNIT

Value	Device Simulated	Capacity
01600	3420 at 1600	
06250	3420 at 6250 BPI	
34800	3480 Cartridge	200 MB
34900	3490 Cartridge	400 MB
34901	3490/E Cartridge	800 MB
34990	Redwood Cartridge	10 GB
34991	Redwood Cartridge	25 GB
34992	Redwood Cartridge	50 GB

TABLE 30. List of Possible Value for DYNnUNIT

Value	Device Simulated	Capacity
35900	3590 Magstar Cartridge	10 GB
35901	3590 Magstar Cartridge	20 GB

For related information, see the topic "Simulate Mode Considerations" on page 345 in the User's Guide.

ENQADDTLn - If this sysparm is set to a value of Y and the USERSENQ user exit is supplied with a value of DMSNOENQ, an additional enqueue on the QNAME SYSDSN will be performed to determine if a data set is in use. If so, a message will be issued stating which data sets were in use at the time of the backup. The default value is N.

For related information, see the user exit description of "USERSENQ — ENQ/DEQ Exit" on page 256 of the this manual.

Note: Not available when using the DFSMSdss Data Mover interface.

ENQDSSIMn - This sysparm controls data set ENQUEUES issued during PDS Compress processing in simulate mode. The default value of this sysparm is N, which causes CA-Disk to bypass data set ENQUEUES in simulate mode. This eliminates data set contention and improves performance, but the report will include data sets that would not have been selected and processed in a live run because they were in use by another task.

If you need a more accurate report from the simulate run, specify this sysparm with a value of Y. CA-Disk will perform data set ENQUEUES during the simulate run, and thus select and report the same data sets that would have been selected in a live run.

ENQGDGBYn - This sysparm affects the enqueuing on GDG data sets. It allows you to optionally bypass the enqueue on the base name of a GDG.

The default value for this sysparm is N. Per IBM integrity conventions for generation data groups, CA-Disk enqueues on the base name before enqueuing on the full data set name. Therefore the enqueue on the base name, whether by CA-Disk or another application, can prevent other jobs from accessing any generation within the group; for example, CA-Disk will bypass all generations within the group until the other task releases (dequeues) the base name.

If you specify this sysparm with a value of Y, CA-Disk will bypass the enqueue on the base name of the GDG data set and process individual generations, based upon their enqueue status. Since IBM conventions are not being used, unpredictable integrity exposures can arise for which CA-Disk is no longer responsible.

Note: JES3 customers must specify this sysparm with a value of Y because the CA-Disk JES3 enqueue allocation technique automatically causes an enqueue of the GDG base.

ENQJES3Tdyn - This sysparm controls the data set serialization method used for JES3. When specified with the default value of DYN, CA-Disk enqueues the data set using Dynamic Allocation (SVC 99). This is recommended setting if you define your DASD to JES3 as execution devices and you don't use GRS or MIM to propagate the SYSDSN majorname. ENQGDGBY must also be set to Y. If you use GRS or MIM to propagate the SYSDSN enqueues or you don't have your DASD devices defined to JES3, then the value of this sysparm should be set to ENQ so that CA-Disk uses ENQ/DEQ to perform data set serialization.

ENQMSGESn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

ENQSTPOKn - When this sysparm is specified with the default value of N, CA-Disk will not process data sets allocated in the job step that is currently running. Enqueues for these data sets normally fail. This is to prevent CA-Disk from processing (that is, archive, delete, etc.) the data sets it has in use which can sometimes cause unpredictable results.

Specifying this sysparm with a value of Y will allow CA-Disk to process data sets in the job step that is currently running. Problems mentioned above can occur, however. Other errors can also occur. For example, you can receive SVC99 errors if you attempt to restore a data set which is also allocated to the currently running step.

This sysparm does not apply to auto-restore processing.

ENQVOLWTnnn - This sysparm controls the length of time between issuance of WTOR message 3017, which tells the operator how long CA-Disk has been waiting for a tape volume that is being used by another job. The default value is 8, for 8 minutes. You can specify a value from 1 to 999 minutes.

ENQWAITDn - This sysparm controls the Enqueue action when CA-Disk is processing its selection criteria. The default value of NN allows CA-Disk to process non-VSAM and VSAM data sets when enqueues can be immediately obtained.

Other possible values for the first variable are:

V wait on enqueues for VSAM data sets only

Y wait on enqueues for non-VSAM data sets only

B wait on enqueues for both VSAM and non-VSAM data sets

Other possible value for second variable when wait is requested:

W CA-Disk enqueue processing allows the operator to terminate the wait through an operator reply. This is invoked for non-VSAM BACKUP, ARCHIVE and RESTORE. For VSAM, only BACKUP and ARCHIVE are supported. This WTOR is not initiated for auto-RESTORE.

The WTOR message:

DMS4096 ENQWAITD- DATA SET WAITING FOR ENQ, DSN= (dddddd).
DMS4096 ENQWAITD- REPLY 'N' TO BYPASS".

This message is only issued when the first variable is: V, Y, or B. When the operator responds 'N' to DMS4096, the following message will be found in MSGPRINT:

4099 OPERATOR HAS BYPASSED DSN WAITING FOR ENQ: (dddddd)

EXPARSOPy - The default value for this sysparm is Y, which causes the change bit to be reset (turned off) after an explicit batch archive that does not scratch the data set. Specify a value of N to change the default to no reset.

EXPASCOPs - Specify a value of N to change the option to no scratch in explicit batch archive. The default is S, for scratch.

EXPDTUSEn - When data sets are archived, CA-Disk assigns an expiration date from either the CA-Disk command or from a default value set by a sysparm. If you want to use the expiration date from the data set's VTOC

entry (or its catalog entry for VSAM), if one exists, specify this sysparm with a value of Y. The default value is N, which disables this feature.

If Y is specified and the expiration date from the VTOC entry (catalog entry for VSAM) is less than the calculated default value, the default value will still be used.

Warning: This option applies to the expiration date being assigned to each DSNINDEX record, and does not directly affect the expiration date of associated ARCHVOLS records. All archive functions should use an expiration date of 99365 for the archive volumes if this sysparm option is used. Failure to do so can result in archive volumes expiring before the data sets contained on them are expired. Supply this expiration date in the JCL or by using sysparm DYNEXPDT.

FASTCOMPy - The default value of Y causes CA-Disk to compress PDSs using a newer faster method. Specify a value of N only if directed by the Technical Support Center.

FILESCNT12 - This sysparm controls how many subfiles can be contained in the files data set. The default value is 12.

FILESPAC1500U - This sysparm is active only when the files data set logging feature of CA-Disk has been activated (see sysparm FILOGNAM). This sysparm controls the number of blocks allocated when the files log data sets are created. The default value is 15000U, causing 15000 blocks to be allocated. You can specify this sysparm with any 5-digit numerical value followed by a character U, K or M, where the character indicates the multiplier to be used with the 5 digit value (U for 1, K for 1024 and M for 1,048,576).

FILEUNITnnnnnnnn - This sysparm is active only when the files data set logging feature of CA-Disk has been activated (see sysparm FILOGNAM). This sysparm controls the type of unit allocated when the files log data sets are created. The default unit name is SYSALLDA. You can specify any valid unit type for your installation.

FILOGNAMxxxxxxxxxxxxxxxxxxxxxxxxxx - Specify this sysparm to activate the files data set logging feature of CA-Disk. For the value of this sysparm, specify the base name (high-level node) to be used to create the files log data set name. The base name can be 1 to 25 characters. When this sysparm is specified, CA-Disk will dynamically allocate data sets to log activity to the files data set. The default is blanks, which disables logging for the files data set. See Files Data Set Logging, Activation section for restrictions.

FLYCOUNT2 - CA-Disk prints report separator pages on the SYSPRINT dd statement for all CA-Disk executions. It appears as the first page(s) printed before CA-Disk-generated output. We refer to the separator pages as "fly sheets". They provide two benefits for CA-Disk users; first, as a separator page for different report steps, and second, to provide useful information about the environment in which CA-Disk is running.

This sysparm controls the printing of CA-Disk fly sheets. It specifies the number of fly sheets to be printed. To suppress their printing, specify a value of 0 for this sysparm. To obtain two fly sheets before printed output, specify a value of 2, the default value. The value must be a single numeric digit.

FORCPOOLnnnnnnnn - When tape or disk archive pools are being used, the pool names associated with the archive tapes and disk volumes are specified in the POOLDEFS member of the parmlib data set. A pool name supplied in the "nnnnnnnn" field of this sysparm will override the normal definitions, forcing all archive volumes to be taken and returned to the named pool. This sysparm is intended for use in special testing or recovery environments only.

FMSSAFEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the FMSSAFEX user exit. See the user exit description for “FMSSAFEX — FMS After Selection Exit” on page 221 for more information.

FMSSPREXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the FMSSPREX user exit. See the user exit description for "FMSSPREX — FMS Prior Selection Exit" on page 221 for more information.

GDGCATEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the GDGCATEX user exit. See the user exit description for "GDGCATEX — GDG Data Set Catalog Action" on page 222 for more information.

HONORTTRn - This sysparm controls whether CA-Disk Restore or Move/Copy processing will honor the TTR value of records in data sets of DA, PS and unknown organization types.

The default value for this sysparm is N. This causes CA-Disk to honor the TTR for all data sets with DSORG=DA and a RECFM=U or V. At archive time, any other DA, PS or unknown DSORG data set in which an empty track is encountered prior to the end-of-file is also marked as requiring this processing at restore time. For other attribute combinations of DA, PS and unknown DSORG data sets, CA-Disk will copy the records in correct order but, depending on the data and device characteristics, the records can be moved to different relative track and record locations, per standard IBM conventions.

Note: For PS data sets with RECFM=FBS, this sysparm should be set to N.

If you specify this sysparm with a value of Y, Restore and Move/Copy processing will preserve the TTR value for data sets of these organizations (DSORGs), regardless of what the record formats (RECFMs) are. Specifying a value of Y for this sysparm is intended only for those exception cases in which the data sets do not follow/allow conventional access techniques for the attributes that are indicated.

Note: Not available when using the DFSMSdss Data Mover interface.

ICHEKAEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the ICHEKAEX user exit. See the user exit description for "ICHEKAEX — RACF ICHEINTY Processing Post-ICHEINTY Exit" on page 222 for more information.

ICHEKPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the ICHEKPEX user exit. See the user exit description for "ICHEKPEX — RACF ICHEINTY Processing Pre-ICHEINTY Exit" on page 223 for more information.

IDRCRATE00 - Specify for this sysparm the average amount of compression realized when using IDRC. This value, given as a percent, is used to estimate the size of a data set while allowing for compression. Therefore, a better prediction can be made regarding how many feet of tape or 3480 cartridge are needed. The default value is 35, or 35 percent average compression. This value is used when a DSNINDEX record for the data set cannot be found.

IDRCSIZE005 - CA-Disk uses this sysparm in order to determine when data set space usage should be calculated. The default value is 005, which equates to 5 megabytes. Data set below this size does not have their IDRC compression rate or tape utilization calculated.

When CARTCALC is specified as Y and IDRC compression is used (see sysparm "IDRCUSEDx"), CA-Disk issues I/O requests to tape controllers tracking the compression rate of data written onto cartridges. On devices attached to 3494 controllers, the overhead generated by the I/O requests is about half a second per data set (1.5 hours for 10,000 data sets). This sysparm allows you to reduce the overhead by instructing CA-Disk to only track IRDC compression rate for those data sets larger than the value specified. While this will improve tape I/O performance,

it also causes CA-Disk to assume zero compression for data sets smaller than the value specified (i.e., reduction in output tape usage). Specifying 000 deactivates this support.

IDRCSTATv - The default value of V for this sysparm allows the archive volume compression totals to be printed. Specify N to suppress these messages. See message number “3825” on page 299 in the Message Manual for the format.

IDRCUSEDx - This sysparm indicates whether or not IDRC (Improved Data Recording Capability) is activated when creating output tapes. The values that can be assigned are blank, Y, or N. When the sysparm is specified with the default value of blank, it means that whatever system setting exists as a result of the installation default and/or JCL parameter specification should be used. When the sysparm is specified with a value of Y, it means that the compaction feature (IDRC) should be activated if the capability is available. When the sysparm is specified with a value of N, it means that the compaction feature should not be activated when creating output tapes.

IGNLSTARy - With this sysparm, CA-Disk determines if a non-VSAM data set is empty for the purposes of archiving, backing up, or Move/Copy processing.

When CA-Disk checks if a non-VSAM data set is empty, it applies the following tests:

- the DS1LSTAR value in the format-1 DSCB is zero
- the DS1TRBAL value in the format-1 DSCB is non-zero
- the DS1VOLSQ value in the format-1 DSCB is one

The last test is based on the above sysparm setting. If the sysparm is set with a value of Y, the next test is also applied. The default value is Y.

- bit 0 of the DS1DSIND field in the format-1 DSCB is zero

If the value of this sysparm is N, this last check is not done.

If the data set meets all of these criteria, it is backed up or copied as empty. If the data set does not meet these criteria, it is not considered empty and is processed accordingly.

Some data sets meeting all but the last criterion could have been opened for output. So, data can exist if the program that wrote to the data set did not update the DS1LSTAR field. To capture data sets that cannot be truly empty, specify this sysparm with a value of Y.

This sysparm does not apply to direct access data sets with variable or undefined record formats. These data sets are never considered empty and are always backed up or copied until an end-of-file is detected.

IOCHNBLKn - Specify this parameter to turn on Archive Output Tape chaining. The value ‘n’, from 1 to 9 specifies the number of tape blocks to be written to the tape concurrently. A higher number will reduce the number of EXCPs issued to the tape, reducing elapsed time due to tape channel conflicts. It increases the storage utilization by (n-1)*tape blocksize*2. Using disk output media for either primary or secondary output will set this value to 1. The default value is 1.

IOERRLIMnnnnn - This sysparm determines the number of I/O errors allowed during a VBACKUP run before the volume should be bypassed. The default for this sysparm is 0, indicating that a volume is to be bypassed if any I/O error is encountered. Specify a 1- to 5-digit value to indicate the number of allowable I/O errors.

During VBACKUP, the number of I/O errors as specified in this sysparm up to a maximum of 100, are saved. For example, if an user specifies 150 for the IOERRLIM sysparm, 100 of the I/O errors will be saved. CA-Disk will continue processing the volume until more than 150 I/O errors have been encountered. CA-Disk saves the CCHH value for the track in error, along with two sense bytes indicating the type of error. The I/O errors detected are written to the archive tape as one record.

Note: Not available when using the DFSMSdss Data Mover interface.

IOMAXRECnnnnnn - This sysparm specifies the maximum logical record size for data sets processed by CA-Disk. The default value is 65,000 bytes. The maximum value for this sysparm is 999,999 bytes. To save real storage, this sysparm should be set to the lowest value possible that still allows normal CA-Disk processing.

Note: Not available when using the DFSMSdss Data Mover interface.

IOTRACKSxx - This sysparm controls the number of tracks that CA-Disk can read (for input processing) or write (for output) with a single I/O request. Since a buffer is required to hold the data for each I/O, this sysparm also establishes the buffer size. To allow I/O and CPU processing to be overlapped, CA-Disk always acquires two buffers for each data set that is opened. As should be evident, the value of this parameter has a very dramatic affect upon the amount of memory that will be required.

The default value is 30, which allows 30 tracks to be read or written with a single I/O. If the value specified is larger than the number of tracks per cylinder for the device being processed, CA-Disk sets it to the tracks per cylinder value; that is, 15 for a 3380, 30 for a 3350, etc.

Processing multiple tracks per I/O reduces the total I/O activity against the channels, control units and devices, and generally "improves" performance. However, memory requirements, paging activity, and many other factors affect performance and must also be considered when selecting and testing a value for this parameter.

Note: Not available when using the DFSMSdss Data Mover interface.

ISAMRECOc - This sysparm indicates what CA-Disk is to do after encountering an abend during processing of an ISAM data set. When an abnormal condition in an ISAM data set causes ISAM access methods to terminate with a system abend, CA-Disk intercepts the system abend, prints messages, and attempts to continue processing. The abend can have left ISAM system control blocks in a status that does not allow further ISAM processing. If CA-Disk attempts to process another ISAM data set, unpredictable results can occur. This sysparm indicates what CA-Disk is to do if an attempt is made to process another ISAM data set after an ISAM abend has previously occurred. The valid values are:

TABLE 31. Possible Values Available for ISAMRECO

Value	Meaning
C	Continue and attempt to process additional ISAM data sets. Processing another ISAM data set can cause CA-Disk to abend (default value).
A	Abend CA-Disk if another ISAM data set is selected for processing.
B	Bypass processing for any additional ISAM data sets selected for processing.

Note: Not available when using the DFSMSdss Data Mover interface.

IXCTLGEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the IXCTLGEX user exit. See the user exit description for "IXCTLGEX — User Exit for IXCATLG Utility" on page 223 for more information.

IXMCLEANy - This sysparm indicates if the CA-Disk archive index maintenance utility (IXMAINT JCL procedure) is to remove invalid records from the DSNINDEX and ARCHVOLS subfiles during processing. ARCHVOLS records must relate to existing DSNINDEX records, and DSNINDEX records must relate to existing ARCHVOLS records. Invalid records are those for which corresponding records do not exist in the opposite file.

The default value is Y, which indicates that invalid records are to be removed when found. A value of N indicates that records are to remain in the files.

IXMDUPDTn - Normally, CA-Disk archive index maintenance does not consider data sets archived on the same date to be different versions. For this reason the COPIES= parameter will not dispose of them as desired when this sysparm is specified with a value of N, the default value. If data sets with duplicate data set names archived on the same date are to be considered as different versions, specify this sysparm with a value of Y.

IXMQTIME0000 - Normally, CA-Disk archive index maintenance enqueues on the files data set and maintains this enqueue for the duration of the process. This processing method causes other functions (i.e., DMSAR, ARCHIVE, BACKUP, etc.) to wait for long periods of time before they can begin processing. This sysparm controls whether or not IXMAINT should periodically release the enqueue, and if so, for how long.

The default value is 0000, which causes IXMAINT to acquire and maintain the enqueue for the duration of the process. Values greater than zero indicate how long (in tents of a second) IXMAINT can hold the enqueue in a locked or enqueued state.

As an example, review Table 32 for possible values that can be supplied to IXMQTIME:

TABLE 32. Sample Values for Sysparm IXMQTIME

Value	Result	Definition
0000	0.0	Default value, IXMAINT will not release the enqueue until its processing is complete
0001	0.1	Release the enqueue 10 times a second
0005	0.5	Release the enqueue twice a second
0010	1.0	Release the enqueue once every second
0300	30.0	Release the enqueue once every 30 seconds. Since all Data Centers differ, this value might be a good starting point.
1000	100.0	Release the enqueue once every 100 seconds (just over 1.6 minutes)
9999	999.9	Release the enqueue once every 1000 seconds (approximately 16 minutes)

Larger values allow IXMAINT to process more efficiently at the expense of running other functions at a slower pace. Smaller values allow other functions to process faster at the expense of running IXMAINT slower. Other functions will acquire the enqueue only long enough to add or update a files data set record, after which IXMAINT will reacquire the enqueue and continue processing.

IXMREVEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the IXMREVEX user exit. See the user exit description for "IXMREVEX — Screen Index Maintenance Deletes (Post Exit)" on page 224 for more information.

IXMVOLONn - The default value of this sysparm is N, which causes CA-Disk index maintenance (DSNDELETE, VOLDELETE, PURGE) and merge processing to uncatalog obsolete catalog entries (see sysparm "UNCATPSUn"

on page 187 for related information) for data sets on offline volumes. Specify a value of Y to not uncatalog data sets on offline volumes.

IXUSRNOTn - This sysparm indicates if the CA-Disk archive index maintenance utility (IXMAINT JCL procedure) is to send a TSO message to a userid if only one copy of a data set is left in the files data set and that data set is uncataloged.

The default value is N, which indicates that no message is to be sent.

A value of Y indicates that a warning TSO message is to be sent. The TSO message warns that there is only one copy of the data set left. It is sent to the userid that either archived or backed up the data set, or if the expiration date for the data set in the files data set has been updated, to the userid that last did the update.

LISTDMEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the LISTDMEX user exit. See the user exit description for "LISTDMEX — Screen TSO LISTDMS Command" on page 225 for more information.

LISTDVEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the LISTDVEX user exit. See the user exit description for "LISTDVEX — List Archive Index (LISTD/LISTV) Exit" on page 225 for more information.

LISTREEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the LISTREEX user exit. See the user exit description for "LISTREEX — Screen TSO LISTREQ Command" on page 226 for more information.

LISTSORTn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

LISTVTOCn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

LUSEDATEn - This system parameter controls whether or not restore processing should alter the format-1 last use date of the data set being restored. The default value of N leaves the last use date unaltered. Specifying a value of Y causes restore to set the last use date to the current date.

Note: For VSAM, this system parameter supports only clusters cataloged in ICF catalogs.

For related information, see "Last Use Date Maintenance" on page 246 in the User's Guide.

MERCnNAMnnnnnnnnnnnnnnnnnnnnn - Merge processing can intelligently redistribute archived data sets on the basis of their expiration dates, by the use of the LIMITS= parameter. Two limit values cause a 2-way split, three a 3-way split, with a 5-way split being the maximum. Each of these new primary archive data sets can also have a duplicate copy created concurrently (duplexing). The new primary data sets are referenced via //ARCHIVE1 (2, 3, 4, 5) dd statements, and the copies by //ARCHIVC1 (2, 3, 4, 5).

If MVS dynamic allocation is used to allocate the //ARCHIVC1 (2, 3, 4, 5) dd statements, these sysparms provide the data set names to be used. The nnn value is either the 1- to 22-character base name, the complete name or the tape GDG index, depending on installation JCL and other sysparm options. The default values are DMS.MERGCOP1 (2, 3, 4, 5) for the duplicate copy data sets created by the merge process.

MERCnTYPtape - If the //ARCHIVC1 (2, 3, 4, 5) dd statement is not provided in the JCL, these sysparms indicates the type of device to be dynamically allocated. It must specify one of the following: TAPE, 3480, DISK, DYN1 through DYN9 or NULL. TAPE is the default. This feature is for MVS users only.

MERDSNEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the MERDSNEX user exit. See the user exit description for "MERDSNEX — Data Set Being Deleted Through Merge" on page 226 for more information.

MERFASTSy - Normally, simulate processing of the merge function does not invoke CA-Disk user exits. This is done to improve the performance of processing. To cause simulate processing to invoke the user exits, specify this sysparm with a value of N.

MERDISVLnnn - This sysparm controls the maximum number of disabled volumes that a CA-Disk MERGE job can process at one time. The default value is 25. Change this limit by specifying a value of 1 to 999 for this sysparm.

MERPnNAMnnnnnnnnnnnnnnnnnnnnn - Merge processing can intelligently redistribute archived data sets on the basis of their expiration dates, by the use of the LIMITS= parameter. Two limit values cause a 2-way split, three a 3-way split, with a 5-way split being the maximum. Each of these new primary archive data sets can also have a duplicate copy created concurrently (duplexing). The new primary data sets are referenced via //ARCHIVE1 (2, 3, 4, 5) dd statements, and the copies by //ARCHIVC1 (2, 3, 4, 5).

If MVS dynamic allocation is used to allocate the //ARCHIVE1 (2, 3, 4, 5) dd statements, these sysparms provide the data set names to be used. The nnn value is either the 1- to 22-character base name, the complete name or the tape GDG index, depending on installation JCL and other sysparm options. The default values are DMS.MERGPRI1 (2, 3, 4, 5) for the primary data sets created by the merge process.

MERPnTYPtape - If the //ARCHIVE1 (2, 3, 4, 5) dd statement is not provided in the JCL, these sysparms indicate the type of device to be dynamically allocated. It must specify one of the following: TAPE, 3480, DISK, DYN1 through DYN9. TAPE is the default. This feature is for MVS users only.

MERRELTMnn - Specify Merge release time of 01 to 99 minutes (default is 00) to be used in both Merge and DMSAR. MERRELTM is used where DMSAR is delayed by a Merge job outputting large tape volumes which hold an archived data set. During Merge, if MERRELTM is non-zero and there is room on the current volume for the next Archvol, the time interval since the last output volume swap is compared to MERRELTM. If that time interval is greater than or equal to MERRELTM, a test ENQ for DMSWVOLS is issued for that volume. If DMSAR has been waiting with a DMSWVOLS ENQ for that volume, the volume is released by Merge. DMSAR only issues the DMSWVOLS ENQ if MERRELTM is non-zero. The actual value of MERRELTM is picked up in Merge. Tuning can be done within Merge with MERRELTM from 01 to 99 minutes, depending on your needs.

MERREPLYn - CA-Disk MERGE or XCOPY processing can be instructed to allow the operator to terminate the process normally through operator reply between any input tape mount. If you want to activate this feature, specify 'Y' for the value of this parameter. The default value is 'N.' For MERGE or XCOPY, an operator response of 'END' to the WTOR will gracefully shut down at the end of the current input volume. For MERGE only, an operator response of 'NOW' to the WTOR will gracefully shut down at the end of the current input data set. Review the topic "Operation Concerns" on page 343 in the User's Guide for more details about this feature.

MERVOLEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the MERVOLEX user exit. See the user exit description for "MERVOLEX — Archive Volume Being Deleted By Merge" on page 226 for more information.

MIGBYPASn - Specify a Y for this sysparm if you want to see the reasons why the sequential migrate to tape function has bypassed a data set. The default value of N means that no messages are given for data sets not meeting selection criteria specified.

MIGCATLGy - The default value of Y for this sysparm causes all uncataloged non-GDG data sets to be cataloged when migrated to tape. Specifying a value of N causes no change to the catalog.

MIGCREDTn - Specify a value of Y for this sysparm to cause all data sets migrated to tape to have their creation date equal to the creation date the data set was assigned on DASD. The default value of N indicates that CA-Disk is to reset all creation dates to the current date.

MIGCTGDGn - Specify a value of Y for this sysparm to cause all uncataloged GDG data sets to be cataloged when migrated to tape. The default value of N will cause no change to the catalog. If a GDG data set is not in the catalog, no attempt is made to catalog it because VSAM catalogs will uncatalog the oldest generation when the index is full. See NOEMPTY parameter for the DEFINE GENERATIONDATAGROUP command in IBM's Access Method Services manual.

MIGDISEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the MIGDISEX user exit. See the user exit description for "MIGDISEX — Screen Sequential Migrate Disposition Request" on page 227 for more information.

MIGDSMAXnnnn - This sysparm specifies the maximum number of data sets that can be migrated to a single tape volume. The default value is 9999.

MIGDSORGn - Specify this sysparm with a value of Y if data sets with a null DSORG are to be migrated during sequential migration as if they were physical sequential. The default value is N, which causes CA-Disk not to migrate null DSORG data sets. This parameter applies only to the implicit SCAN command.

MIGEXPDTn - Specify a value of Y for this sysparm if the expiration date is to be taken from the JCL. The default value is N, causing CA-Disk to use the data set expiration date.

MIGFXPDTy - Specify a value of N for this sysparm to cause CA-Disk, in sequential migrate, to not add one day to the expiration date of the data set when it is equal to the current date. Tape management systems consider data sets on tape with an expiration date of the current date to be a tape work file and will not log the data set in the tape management catalog.

MIGGDGEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the MIGGDGEX user exit. See the user exit description for "MIGGDGEX — Return GDG Limit to Sequential Migrate" on page 228 for more information.

MIGMODELn - This sysparm's default value of N causes model DSCBs to be bypassed during implicit SCAN processing for sequential migration to tape. Specify Y to permit model DSCBs to be migrated.

MIGNOEXPn - This sysparm's default value of N causes expired data sets (non-zero expiration dates less than current date) to be bypassed during implicit SCAN processing for sequential migration to tape. Specify Y to permit expired data sets to be migrated.

MIGNXPDTy - This sysparm's default value of Y allows data sets with zeroes in the expiration date field to be selected during implicit SCAN processing for sequential migration to tape. Specify N to cause these data sets to be bypassed.

MIGPSCOPn - Specify a value of Y to create a duplicate backup tape concurrently with the primary sequential migration tape. The default value is N, for no duplicate.

MIGPSDENddddd - The default is 06250, which is appropriate for standard 3420 devices. By changing this value, you can specify a different tape density while simulating a Sequential Migrate to Tape job. It is only used for simulation purposes. In live mode, the actual density of the device is used. For instance, if you want to simulate other devices, change this sysparm to one of the following values:

TABLE 33. List of Possible Values for MIGPSDEN

Value	Device Simulaated	Capacity
01600	3420 at 1600	
06250	3420 at 6250 BPI	
34800	3480 Cartridge	200 MB
34900	3490 Cartridge	400 MB
34901	3490/E Cartridge	800 MB
34990	Redwood Cartridge	10 GB
34991	Redwood Cartridge	25 GB
34992	Redwood Cartridge	50 GB
35900	3590 Magstar Cartridge	10 GB
35901	3590 Magstar Cartridge	20 GB

Note: The simulated devices represented in this table will only work when sysparm CARTCALC is specified with a value of Y. If CARTCALC is set to N, all 3480 and/or 3490 devices use CARTFEET. For related information, review the sysparm descriptions of "CARTCALCn" on page 130, "CARTPRCTnnn" on page 131, and "CARTFEETnnnn" on page 131.

MIGREXITmmmmmmmm - This sysparm allows you to specify the 8-character name of the user-supplied exit module to receive control after a data set has been selected for migration from disk to tape. See the user exit description for "MIGREXIT — Sequential Migration Disposition Screening" on page 228 for more information.

MIGRSORTndv - Specify this sysparm to change the selection of sequential migration reports produced. Specifying values of UDV will cause CA-Disk to produce migration reports sorted by different fields as described below:

TABLE 34. Possible Values for sysparm MIGRSORT

Value	Meaning
U	print unsorted sequential migration report
D	print migration report in dsn/volume sequence
V	print migration report in volume/dsn sequence

Specify N for any report not desired. The default values are NDV.

MIGYRCHGn - Specify a value of Y if you want to have sequential migrate to tape mount a new tape volume when the expiration year changes for the data sets being migrated. The default value of N causes no tape switching to be done.

MVDCATLGy - This sysparm can be used to suppress catalog referencing during the preparation of the multiple volume data set dictionary by specifying a value of N instead of the default value of Y. The default value will cause CA-Disk to do catalog locates for every data set listed on the MVD report, in order to provide the report item

indicating if the data set is cataloged or not. If this information is not needed on the report, change the value to N to gain a significant performance improvement in report processing.

NEWPATHNy - This sysparm instructs CA-Disk as to which PATH naming conventions should be used during RESTORE and RECOVER.

TABLE 35. Possible Values for sysparm NEWPATHN

Value	Meaning
Y	Default action causing 9.0 PATH naming conventions to be used. When this value is used in conjunction with the RESTORE NEWNAME= parameter, CA-Disk uses the Base Cluster or AIX new name, depending on the path entry alignment, and appends ".PATH" to the PATH as the LLQ.
N	Causes 8.2 PATH naming conventions to be used. When this value is used in conjunction with the RESTORE NEWNAME= parameter, CA-Disk only changes the HLQ of the PATH matching the Base Cluster. The LLQ remains unchanged.

NOBLKMSGn - Specify a value of Y if you want to see the reason why a data set was not reblocked by the Move/Copy function. The default value of N means no messages are issued explaining why reblocking was not done.

OFFLDASDn - Specifying a value of Y instructs the RESTORE command to issue a DELETE NOSCRATCH catalog action prior to restoring the data set. This catalog action helps prevent abends in cases where the data set being restored has ONLY a catalog entry, or when the target volume is offline (no valid UCB). The default value of N bypasses any catalog action.

Note: This sysparm is not valid for the RECOVER command.

ONCCFAILcancel - This sysparm controls what action CA-Disk is to take when the Concurrent Copy initialization fails. The default value of CANCEL instructs CA-Disk to cancel processing for ALL data sets. Specify a value of CONTINUE if you want CA-Disk to continue processing, maintaining serialization on those data sets for which the initialization failed.

The value specified for this sysparm is passed as an output parameter to the CCINITEX user exit. However, if CANCEL is returned by the exit for any data set, CANCEL will be in effect for all data sets, and cannot be overridden by a return of CONTINUE for any other data set.

PASSNEWNn - Specify a value of Y if you want to allow password- indicated non-VSAM user data sets to be renamed. Since CA-Disk does not update the password control data set itself, renaming a password-indicated data set can be a security exposure. The default value of N prevents this condition.

PASSWORDn - The default value for this sysparm is N, which causes CA-Disk to bypass processing for all password-indicated data sets when they are encountered. Whenever a password-indicated data set is bypassed, a message to that effect will be issued.

Specifying a value of Y allows CA-Disk to process password- indicated data sets. The user can also have to use the PASSWORD parameter on the appropriate CA-Disk command. Processing of password-indicated data sets is supported in all functions.

If you have the security package SECURE installed, you should also specify sysparm SECUSUPP with a value of Y. You should also consider specifying sysparm PASSNEWN.

If sysparm SECUSUPP is not specified with a value of Y, CA-Disk supports processing for password-indicated data sets only in the MVS and SVS environments (SVS requires the installation of a system zap modification and the CA-Disk SVC). If the above conditions are not met, the value of N for this parameter will always be assumed.

PDSANMLYmmmmmmmm - This sysparm allows you to specify the 8-character name of a member in the parmlib data set that contains the PDS Anomaly Code and Options Specifications. Review the topic "Processing PDSs that Contain Anomalies" on page 75 for further information regarding CA-Disk-provided options to bypass or continue processing partitioned data sets in which CA-Disk has detected errors.

Note: Not available when using the DFSMSdss Data Mover interface.

PDSELOADn - Specify a value of Y if you want PDSE load libraries to be processed. The default value of N causes them to be bypassed.

PDSENQSHn - The default value for this sysparm is N, which causes CA-Disk to obtain an EXCLUSIVE ENQ during PDS Compress processing.

If this sysparm is specified with a value of Y and the USE parameter is specified for the compress command, a SHARED ENQ is maintained for the data set during compression. (Without the USE parameter, a value of Y for this sysparm is ignored.)

Using a SHARED ENQ might be desirable, for example, to recover dead space in TSO PDSs during high use periods, since the SHARED ENQ could be obtained more readily. And because CA-Disk also issues the ISPF and linkage editor enqueues when using a SHARED ENQ, processing between TSO/ISPF and PDS Compress are serialized, preserving the data set's integrity.

Specifying this sysparm with a value of Y has no effect on the RESERVE command parameter for PDS compress processing.

Note: Specifying this sysparm with a value of Y can cause an integrity exposure if a batch job (other than the linkage editor) is updating a PDS that has only a SHARED ENQ.

Also, if a SHARED ENQ is used in an environment in which other batch jobs are using the PDS concurrently, unpredictable results can occur; for example, member not found, S806-04 and S106 abends, and so on. In this case, rerun the batch jobs when compress processing has completed.

PDSEXSIMy - Specify a value of N for this sysparm if the exclusion table should be nullified during simulated PDS compression. The default value of Y indicates that the table is active even in simulate mode.

PDSMAXPR00000 - This sysparm is used during PDS Compress to set the maximum number of primary tracks to be calculated for reallocation of the PDS. If the CA-Disk calculation for primary tracks generates a number greater than this number, the value of this sysparm will be used unless it is less than the number of tracks used in the PDS. In this case, the actual number of used tracks will be used. If the value of this sysparm is set to zero, no maximum limit is set. This sysparm is in effect only if the user has specified a value other than 0 in the primary space specification of the ALLOCATE= parameter. The default value of this parameter is 00000.

If the PDS being compressed is allocated in cylinders, this value will be converted to a cylinder value. During this calculation it can be necessary to round the value up to a higher value that will fall on a cylinder boundary.

PDSMAXSE99999 - This sysparm is used during PDS Compress to set the maximum number of secondary tracks to be calculated for reallocation of the PDS. If the CA-Disk calculation for secondary tracks generates a

number greater than this number, the value of this sysparm will be used. If the value of this sysparm is set to zero, the secondary extent allocation amount will be set to zero. This sysparm is in effect only if the user has specified a value other than 0 in the secondary space specification of the ALLOCATE= parameter. The default value of this parameter is 99999 tracks.

If the PDS being compressed is allocated in cylinders, this value will be converted to a cylinder value. During this calculation it can be necessary to round the value up to a higher value that will fall on a cylinder boundary.

PDSMINDB00000 - This sysparm is used during PDS Compress to set the minimum number of directory blocks to be calculated for reallocation of the PDS. If the CA-Disk calculation for directory blocks generates a number of directory blocks to allocate less than this number, the value of this sysparm will be used instead. This sysparm is in effect only if the user has specified a value other than 0 in the directory block specification of the ALLOCATE= parameter. The default value of this parameter is 00000.

PDSMINPR00000 - This sysparm is used during PDS Compress to set the minimum number of primary tracks to be calculated for reallocation of the PDS. If the CA-Disk calculation for primary tracks generates a number less than this number, the value of this sysparm will be used instead. This sysparm is in effect only if the user has specified a value other than 0 in the primary space specification of the ALLOCATE= parameter. The default value of this parameter is 00000.

If the PDS being compressed is allocated in cylinders, this value will be converted to a cylinder value. During this calculation it can be necessary to round the value up to a higher value that will fall on a cylinder boundary.

PDSMINSE00000 - This sysparm is used during PDS Compress to set the minimum number of secondary tracks to be calculated for reallocation of the PDS. If the CA-Disk calculation for secondary tracks generates a number less than this number, the value of this sysparm will be used instead. This sysparm is in effect only if the user has specified a value other than 0 in the secondary space specification of the ALLOCATE= parameter. The default value of this sysparm is 00000.

If the PDS being compressed is allocated in cylinders, this value will be converted to a cylinder value. During this calculation it can be necessary to round the value up to a higher value that will fall on a cylinder boundary.

PDSRRMAXnnnnn - The specified "nnnnn" value determines how many 3-byte TTR entries will be reserved in an internal CA-Disk table. The table is used to maintain relative block numbers when processing PDS members with user TTR entries. The default value is 03000. If this value is too small, an error message will indicate to increase its value.

PDSSETBSy - The default value of Y allows CA-Disk to set an optimal blocksize for the work file used by the PDS Compress function. Specify a value of N if you want CA-Disk to honor the blocksize supplied on the // COMPWORK DD statement of the COMPRES step.

PS2LSTARn - The default value of N causes BACKUP/ARCHIVE to prcess data sets to EOF marker. If the value of Y is used, data sets are backed up t and including the record indicated by the LSTAR. This sysparm is only for IMS customers using CA-Disk Backup or Archive processing on single volume OSAM data sets.

POINTABSy - The default value of Y allows CA-Disk to determine if a 3480 is in full function mode or not, and to use POINT TYPE=ABS (high speed search) whenever possible. You can specify N to instruct CA-Disk not to use the high speed search facility.

PREALLOCn - The default value for this sysparm is N, which causes CA-Disk not to allow restore or recover operations to overwrite preallocated non-VSAM data sets. Specify a value of Y to allow Restore operations to overwrite pre-existing non-VSAM data sets (providing the DSORG of the target data set matches the DSORG of

the data set being restored, otherwise the restore will fail). Specify a value of C to allow the overwrite only if the preallocated data set's create date is the same as that of the archived version.

See sysparm "VSPREDEFn" on page 198 for restore of VSAM data sets.

In order to allow concurrent CA-Disk member-level restore processing and ISPF editing of a PDS, CA-Disk internally assigns and uses a value of "Y" for this sysparm when the MEMBER= parameter is specified and the ERASE parameter is not specified.

(To protect integrity, the CA-Disk auto-restore function internally assigns and uses a value of "N" for this sysparm; that is, an auto-restore will not overlay an existing data set.)

Note: Not available when using the DFSMSdss Data Mover interface.

PRIALLOCn - Specify a value of 'Y' to use the original primary allocation amount of the data set for allocation during the PDS Compress, RESTORE, RECOVER, COPY, and MOVE functions. Specify a value of "3" to use the combined value of the first three extents. The default value of 'N' uses the consolidated allocation amount unless it would exceed the volume capacity.

If there is no secondary allocation value PRIALLOCN is used automatically. If the consolidated allocation is greater than 64k, PRIALLOC3 is used automatically. If there is not enough space on the volume and PRIALLOC3 is specified, a retry using PRIALLOCY will be used. If the default value PRIALLOCN is specified and there is not enough space on the volume, a retry using PRIALLOC3 is used. If that fails, PRIALLOCY is used.

Note: Not available when using the DFSMSdss Data Mover interface.

PRTSORTFy - The default value of Y causes a footing line showing the sort order to be printed for online reporting. Specify N to not print the sort footing line.

PSUDEVTPn - When recataloging archived data sets to the volume specified in the RECATVOL sysparm, the default is to put an invalid device type in the catalog entry for that data set. This is done on the assumption that the RECATVOL volume is a pseudo-volume, and you want to prevent the operating system from calling for a mount of the pseudo-volume when the data set is referenced. A JCL error results indicating the data set must be restored first. If you desire to keep the original device type of the data set, specify a value of Y for this sysparm. The default value is N.

If you specify a value of Y for this sysparm and CA-Disk attempts to recatalog a data set to its original device but this device is offline, sysparm REDEVDEF can be specified to cause CA-Disk to use a default device (3380) rather than issue an error message. See sysparm "REDEVDEFn" on page 166 for further information.

If you have implemented only the IFG0EX0A (S213 abend) hook of the auto-restore support and want to direct the restored data sets to a common "staging" volume, you must specify a real online volume for the RECATVOL sysparm and specify this sysparm with a value of R. This causes CA-Disk to obtain the correct device type to be placed in the catalog entry.

If you are using the catalog management hook provided by CA-Disk for your auto-restores, this sysparm should be left with the default value of N and the RECATVOL sysparm should specify a non-existent (pseudo) volume. The default value for the RECATVOL sysparm is ARCIVE.

RAADDAEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RAADDAEX user exit. See the user exit description for "RAADDAEX — RACF RACDEF-ADDVOL Post Processing Exit" on page 228 for more information.

RAADDPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RAADDPEX user exit. See the user exit description for "RAADDPEX — RACF RACDEF-ADDVOL Processing Pre-RACHECK Exit" on page 229 for more information.

RACFALLOa - This sysparm is functional only if you have installed the CA-Disk RACF security interface. It allows you to control what level of authority CA-Disk will check for when allocating data sets. The default value of A causes CA-Disk to check for RACF ALTER authority before allocating data sets in the Restore, Recover and Move/Copy functions.

Note: For deferred restore requests, you must also set SECURDRS. For details, see "SECURDRSaaaaaaaa" on page 173.

Before allocating the data set, you can instruct CA-Disk to check for certain authority by specifying this sysparm with a value of:

TABLE 36. Possible Values Available for RACFALLO

Value	Meaning
C	to check for CONTROL authority
U	to check for UPDATE authority
R	to check for READ authority
N	to perform no authorization check

If you have a data set allocation control product or site-written DADSM IGGPRE00 allocation exit that checks for RACF ALTER authority before permitting data set allocations, use the default value for this sysparm. If you have generic profiles that disallow most users access to most usable data set names, trusting the ADSP feature of RACF to provide access to newly allocated data sets, specify this sysparm with a value of N. If you have a DADSM IGGPRE00 exit that checks for other than ALTER authority, specify this sysparm with a value consistent with your exit.

If you do not have a DADSM IGGPRE00 exit and are not a heavy user of the ADSP feature of RACF, specify this sysparm with a value consistent with your data management and security philosophy.

Note: Not available when using the DFSMSdss Data Mover interface.

RACFALWZy - Under most versions of IBM's operating systems, OPEN, SCRATCH and RENAME processing will query RACF for authorization regardless of the setting of the RACF-indicator bit. This feature is called always call. Data sets cataloged in ICF catalogs also cause a query of RACF for authorization regardless of the setting of the RACF-indicator bit.

Under some operating systems, data sets not cataloged in ICF catalogs will query RACF only if the RACF-indicator bit is on. For non-VSAM data sets, the RACF-indicator bit is the DS1IND40 bit (bit x'40' at offset 93 x'5D') located in the data set's format-1 DSCB.

CA-Disk security processing normally queries RACF for authorization regardless of the setting of the RACF-indicator bit. If you do not have the always call feature of the operating system and you do not use ICF catalogs, specify this sysparm with a value of N. The default value is Y.

RACFBITSy - When this sysparm is specified with the default value of Y, CA-Disk preserves the value of the RACF-indicator bit. Preserving this indicator bit is good security management in all installations that use RACF or CA-Top Secret. For non-VSAM data sets, the RACF-indicator bit is DS1IND40 (bit x'40' at offset 93 x'5D') in the format-1 DSCB. For VSAM data sets, the RACF-indicator bit is the x'80' bit in the SECF field in the cluster's catalog entry. If you specify this sysparm with a value of N, CA-Disk never sets the RACF-indicator bit in the data sets that it allocates. If you have CA-ACF2 or no security package, and can need to restore CA-Disk backup or archive tapes from sites with RACF or CA-Top Secret, set this sysparm to N to prevent security errors on those restores. To prevent security exposures, installations with RACF discrete profiles should NOT set this sysparm. To do so would cause data sets to become protected by generic profiles instead of their discrete profiles.

RACFBKUPn - If this sysparm is not specified or is specified with a value of N, any CA-Disk function that archives or backs up a data set protected by a discrete RACF profile will create a CA-Disk profile for the data set.

For some users, the additional profile can be costly both in terms of job performance and DASD space utilization in the RACF profile data set.

Generic profiles are not copied, backed up or deleted by CA-Disk.

If this sysparm is specified with a value of Y, CA-Disk will not create any profiles for data sets being backed up. CA-Disk will create a profile for data sets being archived.

If this sysparm is specified with a value of F, CA-Disk will not create any CA-Disk profiles, and in this case, it is the user's responsibility to ensure that a correct generic or discrete profile is available for any data set being restored prior to any user applications accessing that data set.

RACFDAEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RACFDAEX user exit. See the user exit description for “RACFDAEX — Decode CA-Disk RACF Encoded Profile Name (Post-decode)” on page 230 for more information.

RACFDPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RACFDPEX user exit. See the user exit description for “RACFDPEX — Decode CA-Disk RACF Encoded Profile Name (Pre-decode)” on page 230 for more information.

RACFDVL2dmsxxx - In order for PDS Compress processing to preserve data set profiles, CA-Disk temporarily adds a dummy volume to the RACF profile of the data set being compressed. This prevents the data set profile from being deleted when the data set is scratched and reallocated. The dummy volume is then automatically deleted by CA-Disk. This sysparm specifies the volume CA-Disk adds to the data set profile. The volume need not be a real DASD volume. Beware that if a real DASD volume is specified, it must not be a volume that will be processed by PDS Compress. The default value is DMSXXX.

RACFDVOLvvvvvv - This sysparm specifies the real online volume that is associated with all CA-Disk-saved RACF profiles. Specify this sysparm with a stable DASD volume name, such as, SYSRES, MVS001, CAT001, RESCUE or similar one- to six-character volume name.

It is a RACF requirement that discrete RACF profiles beassociated with a real online DASD volume. No I/O or RESERVEs are done to this volume, its sole purpose is to satisfy the RACF requirement.

Under normal circumstances, this sysparm value should not be changed. However, if you have a compelling reason to change it, perhaps because the old volume is being disconnected, see "Changing the Value of RACFDVOL" on page 359 for related information.

RACFEAEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RACFEAEX user exit. See the user exit description for "RACFEAEX — Encode CA-Disk RACF Profile Name (Post-encode)" on page 231 for more information.

RACFENTY2 - Specify this sysparm to select the CA-Disk RACF data set name encoding technique. Only two techniques are supported, 1 and 2. Encode technique 2 is the default.

Note: The default used to be 1. In Release 7.0 the default was changed to 2. For further information on encoding techniques, see "Reconciling CA-Disk and RACF Records" on page 346.

RACFEPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RACFEPEX user exit. See the user exit description for "RACFEPEX — Encode CA-Disk RACF Profile Name (Pre-encode)" on page 231 for more information.

RACFMDSNxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx - See sysparm RACFMODL. It describes the conditions in which this sysparm should be used to provide the name of a model profile.

RACFMODLn - CA-Disk restore processing can fail if a discrete profile that was saved during archive is not available when it is being restored.

Specify this sysparm with a value of Y to have CA-Disk copy a model discrete profile instead of the unavailable saved profile. Using this model profile allows the restore to complete, after which you can correct its profile.

If you specify this sysparm with a Y, you must create the model profile to be used if it does not already exist. Then specify the data set name of the model profile as the value of sysparm RACFMDSN, and the volume of the profile in sysparm RACFMVOL. (A data set does not need to exist for the model profile.)

RACFMVOLvvvvvv - See sysparm RACFMODL. It describes the conditions in which this sysparm should be used to provide the volume of a model profile.

RACFNEWNn - Normally, CA-Disk does not allow RACF-protected data sets to be renamed with CA-Disk functions that support the NEWNAME command parameter. If users want to allow CA-Disk to rename RACF-protected data sets, specify this sysparm with a value of Y. Its default value is N.

RACFPDSWn - Normally, CA-Disk does not explicitly protect the PDS compress work file. This creates a slight security exposure in the event that an abend or operator cancel leaves the work file on disk.

Specify this sysparm with a value of Y to instruct CA-Disk to protect the work file. CA-Disk will then set the RACF indicator bit in the format-1 DSCB of the work file to the same setting of the data set being compressed. If the data set has a discrete profile, CA-Disk will build an equivalent discrete profile for the work file. The work file will thus have protection equivalent to the data set being compressed.

If you want to compress data sets covered by generic profiles or by RACF Global Access Table entries (and not covered by discrete profiles) and you also want to set this sysparm to Y to protect the work file, you must create a generic profile or RACF Global Access Table entry to cover the work file. This is because CA-Disk does not copy generic profiles, and at least one profile must exist to cover the work file. This generic profile or table entry could be very restrictive — it need only allow access to the compress job and any potential reload jobs. This will ensure protection for all PDSs that can be compressed.

RACFPREDn - When restoring data sets protected by discrete profiles, specify a value of Y for this sysparm to have CA-Disk preserve already-existing discrete profiles. This option includes restoring to preallocated data sets, and restoring to non-preallocated data sets where discrete profiles already exist (perhaps in volume recovery situations when the RACF data set is still complete).

In the case of restoring to non-preallocated data sets where discrete profiles already exist, preserving the existing discrete profile is dependent on ADSP. If ADSP is active on the system and the RACF user has the ADSP attribute, CA-Disk will delete the existing discrete profile and create a new profile, using the CA-Disk-saved profile as the model. If CA-Disk determines that ADSP is not active, it will preserve the existing discrete profile.

The default value of N, causes CA-Disk to delete any existing discrete profiles. If saved CA-Disk profiles are available, these replace the existing profiles. Carefully coordinate the specification of this sysparm with that of sysparm RACFBKUP.

RACFPROCy - When this sysparm is specified with the default value of Y, CA-Disk processes RACF-indicated data sets. We recommend this option if you have installed RACF or CA-Top Secret at your installation. RACF users should also see sysparm RACFSUPP.

Specify a value of N to cause CA-Disk to bypass processing of RACF-indicated data sets. Normally, this value should not be used if you have RACF or CA-Top Secret at your installation.

RACFSEQMn - Normal sequential migration to tape processing will bypass RACF-protected data sets. If you want CA-Disk to attempt to process these data sets, specify a value of Y for this sysparm. The default value is N.

Note: CA-Disk does not perform discrete profile checking or copy discrete profiles to the tape data sets when a value of Y is specified.

RACFSEQTy - If sysparm RACFSEQM is specified with a value of Y and this sysparm is specified with a value of N, the output tape created by sequential migrate will not be RACF- indicated. The default value is Y, which causes tapes containing RACF-indicated data sets to be RACF- indicated.

RACFSMEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RACFSMEX user exit. See the user exit description for "RACFSMEX — Process RACF Protected DSN with Sequential Migrate" on page 232 for more information.

RACFSUPPn - When this sysparm is specified with the default value of N, the CA-Disk RACF support is OFF. CA-Disk does not check RACF authority or use CA-Disk RACF profile handling. CA-Disk tries to process RACF-protected data sets without interfacing to RACF. When this sysparm is specified with a value of Y, the CA-Disk RACF support is ON. CA-Disk checks RACF authority and uses CA-Disk RACF profile handling. Make sure that RACFPROC is specified with a value of Y.

RACFTESTn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

RACFUSIDxxxxxx - This sysparm is used to specify a 1- to 8-byte data set name prefix, which will be followed by a period and used as the first qualifier of the encoded data set name of CA-Disk-saved profiles. It can also be used to identify CA-Disk profiles and to allow them to be placed on a RACF data set apart from the standard RACF data set. This sysparm is required for CA-Disk discrete RACF profile support. The value of this sysparm should be chosen with care. Once specified, this prefix should not be changed.

It is a RACF restriction that this RACFUSID value represent a user ID or group ID. We recommend using a user ID, not a group ID, for the value for this sysparm. It should help you to identify CA-Disk profiles. We recommend using

"DMSOS". To avoid having RACF RACDEF processing update the PERMIT list, ensure that the user ID does not have the GRPACC attribute.

This RACF user ID will be able to restore any data set for which CA-Disk has saved a discrete profile. To prevent this exposure of unauthorized use of this RACF user ID, you can revoke it with the TSO command below. Revoking the RACF user ID in this manner will not prevent its use for CA-Disk discrete profile support.

```
ALTUSER racfusid REVOKE
```

RACFVCAVn - This sysparm indicates how the Move/Copy function is to create RACF discrete profiles for newly created data sets on target volumes. The default value is N, indicating that a RACF RACDEF will be done to create new profiles that point to the new data sets and volumes. A value of Y indicates that a RACF RACDEF with ADDVOL will be done to add the new volumes to the profile that already exists for the source data sets. ADDVOL processing will only be done in cases where the data set name for the source and target data sets are the same.

Note: ADDVOL processing will help to preserve RACF statistics maintained in RACF profiles since a RACDEF of a new profile loses statistics.

RACGDGVLvvvvvv - This sysparm is active only for sites with an active RACF support of CA-Disk.

GDG generation data sets can be RACF-protected by discrete profiles. Some releases of RACF also allow creating one model GDG profile with a name of the GDG base name that will cover each of the generations. This simplifies the security job, because only one profile is maintained instead of many. GDGs can also be protected by generic profiles.

Model GDG profiles are associated with a volume, just like discrete profiles. Generic profiles are not associated with a volume. At most installations, the volume associated with a model GDG profile is the same as the volume containing the catalog in which the GDG base is defined. By default, CA-Disk uses the catalog volume to determine the existence of model GDG profiles. CA-Disk does not copy model GDG profiles in archive, restore or any other functions.

Some installations have implemented non-standard model GDG profile handling code in their operating systems that causes all model GDG profiles in the shop to be associated with one single volume name. If your installation uses model GDG profiles and uses the non-standard support so that all model GDG profiles are associated with one volume, specify this sysparm with a value of that volume name. CA-Disk will use that volume in determining the existence of model GDG profiles.

RACHKAEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RACHKAEX user exit. See the user exit description for “RACHKAEX — RACF RACHECK Processing Post-RACHECK Exit” on page 232 for more information.

RACHKPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RACHKPEX user exit. See the user exit description for “RACHKPEX — RACF RACHECK Processing Pre-RACHECK Exit” on page 233 for more information.

RADEFAEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RADEFAEX user exit. See the user exit description for “RADEFAEX — RACF RACDEF Processing Post-RACDEF Exit” on page 233 for more information.

RADEFPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RADEFPEX user exit. See the user exit description for “RADEFPEX — RACF RACDEF Processing Pre-RACDEF Exit” on page 234 for more information.

RADELAEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RADELAEX user exit. See the user exit description for "RADELAEX — RACF RACDEF (Delete) Processing Post-Processing" on page 235 for more information.

RADELPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RADELPEX user exit. See the user exit description for "RADELPEX — RACF RACDEF (Delete) Processing Pre-RACHECK Exit" on page 235 for more information.

RADELSVCnnn - If CA-Disk saves discrete RACF profiles (see sysparm RACFBKUP), and users have been given the CA-Disk ISPF function to delete index records for these data sets (see function "ADSINDX" on page 91 in the Installation Guide), users can have trouble deleting the DSNINDEX records for which CA-Disk has saved a discrete profile.

To keep the records synchronized, when CA-Disk is told to delete a DSNINDEX record, CA-Disk tries to delete the CA-Disk-saved profile, if any. Deleting a discrete profile requires APF-authorization (in fact, the system will abend non-APF-authorized users who attempt it). Most installations run their TSO and ISPF sessions non- APF-authorized, and therefore cannot delete profiles unless APF-authorized by some means.

This sysparm should not be specified if you:

1. do not have discrete RACF profiles
2. specify sysparm RACFBKUP with a value of F
3. do not use the ADSINDX option of CA-Disk ISPF support
4. run your ISPF sessions APF-authorized

If you do not meet any of these criteria and your installation maintains an SVC for authorizing ISPF sessions, specify this sysparm with a value representing your authorization SVC's 3-character number (CA-Disk will issue that SVC to get authorization).

If you do not meet any of these criteria and your installation does not maintain an SVC for authorizing ISPF sessions, any CA-Disk attempt to delete a CA-Disk-saved profile will abend. In this case, remove option ADSINDX from use by your users.

REBLIMITnnnnn - The value of this sysparm controls which technique CA-Disk will use to find a new blocksize when reblocking data sets. If the target blocksize specified is below this value, the algorithm is very straightforward; simply combine logical records into a block until one more would exceed the target blocksize.

If the target blocksize is above the value specified, CA-Disk uses a "best fit per track" algorithm, taking into account the key lengths and overhead bytes per block that can vary for each data set. The track size for the target device is divided by the target blocksize to give a target goal of xx blocks per track. CA-Disk then uses the LRECL, key length and overhead values to find the actual blocksize that will still yield the goal of xx blocks per track.

The default value of this parameter is 99999, such that the first algorithm is always used.

Note: Not available when using the DFSMSdss Data Mover interface.

REBLKBITn - For DFP release 3.1 and above the system can determine what the blocksize of a data set should be during allocation. Data sets that can be reblocked by the system have the DS1REBLK bit on in the DS1SMSFG field of the format-1 DSCB. CA-Disk will honor this bit during allocation depending on sysparm REBLKBIT.

When this sysparm is specified with the default value of N, CA-Disk does not honor the reblock bit. The data set is not reblocked by the system.

When this sysparm is specified with a value of Y, CA-Disk honors the reblock bit during allocation. During a restore allocation, the reblock bit is honored only if the data set was archived with release 8.1 or above.

Note: Not available when using the DFSMSdss Data Mover interface.

REBYPASSy - Specify a value of N for this sysparm if no messages are to be issued to explain why retain has bypassed a data set. The default value of Y causes these messages to be printed. This default value was chosen because the retain function is used for backup, and any data sets being bypassed should be known. Once these are determined to be all right, it can be desirable to suppress the messages.

RECATPSUr - When recataloging to the pseudo-volume, this sysparm's default value of R will cause CA-Disk to recatalog all data sets, whether or not they were originally cataloged. Specify a value of N to not recatalog data sets if they were not cataloged originally.

RECATVOLvvvvvv - Specify the volume serial to be used when the RECATALOG parameter is specified either on an ARCHIVE command or in the RETAIN volume specification table. The default volume serial is ARCIVE. If the value of the parameter is blanks (for example, "RECATVOL "), recatalog processing is not done. For related information, see sysparm "PSUDEVTPn" on page 159.

RECCHCATn - This sysparm is used for volume selection for non-VSAM data sets during the Recover function. Usage of either the VOL= or POOL= parameter will bypass this sysparm. When the default setting of 'N' is used, the volume recovery request will recover each data set to its original volume. Specify a value of `Y' to instruct CA-Disk to recover the data set using its cataloged volume.

RECPRIEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RECPRIEX user exit. See the user exit description for "RECPRIEX — Screen Recover Requests" on page 236 for more information.

REDEVDEFn - This sysparm is used in conjunction with sysparm "PSUDEVTPn" on page 159. When PSUDEVTP is specified with a value of Y, CA-Disk attempts to use the original volume's device type when recataloging a data set. This can cause a problem if the original volume is no longer online. (This condition applies primarily to the IXCATLG utility described in "Auto-Restore Implementation Guidelines" on page 80 in the Installation Guide).

If this sysparm is specified with a value of N, the default value, and the original volume is no longer online when CA-Disk attempts to recatalog a data set, CA-Disk will issue an error message. When this sysparm is specified with a value of Y, CA-Disk will use a default device type (3380) if the original volume is not online.

RESAFTEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RESAFTEX user exit. See the user exit description for "RESAFTEX — Post Restore Exit" on page 236 for more information.

RESCHCATy - This sysparm is used for volume selection for non-VSAM data sets during the Restore function. Usage of either the VOL= or POOL= parameter will bypass this sysparm. When the default setting of 'Y' is used, the restore request will restore the data set to its cataloged volume(s). Specify a value of `N' to instruct CA-Disk to skip the catalog check and to restore the data set to the original volume(s).

The value of this sysparm does not affect the auto-restore function since auto-restore uses different logic to determine the target volume during restore. For further information, review the topic "The Auto-Restore Function" on page 58 in the Installation Guide.

RESCOPYSn - Specify a value of Y in conjunction with RESCOPYTy to instruct CA-Disk to use the primary ARCHVOL when a copy ARCHVOL was not created. The default value is N, which will result in message 0097 when no ARCHVOL copy exists.

RESCOPYTn - Specify a value of Y to instruct CA-Disk to use the duplicate copy tapes for all data set restores, including volume recovery. Default processing calls for the primary archive tape unless it has been disabled.

Note: Use caution when specifying this sysparm with a value of Y during an FMS RECOVER job. Doing so might result in a subset of your data being recovered. For an example of how this could be possible, see Figure 37, "Sample FMS RECOVER Tape Pull List," on page 274 in the User's Guide.

RESCRCATn - This sysparm controls scratch processing for both Restore and Recover functions for preallocated non-VSAM data sets. Invocation of this sysparm occurs when the SCRATCH parameter is specified.

TABLE 37. Possible Results when using RESCRCAT

Data Set Type	Results
Uncataloged data sets	When the default setting of 'N' is used, CA-Disk will scratch the uncataloged preallocated data set if it resides on the same volume as the one to which allocation is being directed
Cataloged data sets	When the default setting of 'N' is used, CA-Disk will delete the cataloged data set if the cataloged volume is the same as the one to which allocation is being directed
Cataloged data sets only	When 'Y' is specified, CA-Disk will delete the cataloged data set prior to volume selection

See the description of sysparm "DSNDUPVLn" on page 141 for related information.

RESCRNEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RESCRNEX user exit. See the user exit description for "RESCRNEX — Screen Restore Requests" on page 237 for more information.

RESIXDLBn - This sysparm is used only in conjunction with RESIXRPD, and controls processing of the backup DSNINDEX record following a successful restore or recover. Backup DSNINDEX records do not have the scratch flag turned on (the data set was not scratched at archive time). The default value of N indicates no change is to be made to the backup DSNINDEX record. Specify a value of Y to cause the backup DSNINDEX record to be processed in the same manner as the archive DSNINDEX record after a successful restore or recover (as controlled by RESIXRPD).

For discussion of this sysparm, see "Using Archive Functions to Manage Active Data" on page 71.

RESIXRPDnn - This sysparm helps control the management of active data sets through the archive function; that is, placing data sets that are used only occasionally in the CA-Disk archives in order to conserve disk space. (These "active" data sets are usually archived to disk in a compressed format, and should be thought of as temporarily compressed rather than archived. See "Using Archive Functions to Manage Active Data" on page 71 for a more detailed explanation of this procedure.)

The default value of 99 for this sysparm indicates that restore processing should not change the expiration date of a data set in the CA-Disk archives.

Specify a value of 00 to delete the data set from the CA-Disk archives after an archived data set is restored or recovered successfully under its original name. Specify a value of 01 to 98 to have CA-Disk assign a new

expiration date to the data set in the CA-Disk archives after it is restored or recovered successfully under its original name. (See also sysparm "IXMCLEANy" on page 151).

The new expiration date for the archived data set is the date it is restored plus the number of days specified by this sysparm. (You might choose this last option to allow the data set to be backed up before the earlier archive rolls out of the CA-Disk archives.)

To prevent a potential loss of archived data, if any of the NOLOAD, NEWNAME, MEMBER or AIXNAME parameters are specified on a Restore command, this function is not active and no change is made to the expiration date of the DSNINDEX record.

This sysparm can be used in conjunction with RESIXDLB to control the processing of DSNINDEX records associated with backup copies; that is, should the DSNINDEX record be deleted or reset after a restore from a backup copy of the data set.

Warning: The use of this sysparm my cause adverse affects on SMS managed data sets. In situations where you don't want to keep multiple archive DSNINDEX records (i.e., after restore) to the expiration date as defined by the management class, this sysparm should be set to a non-default value. This is because CA-Disk does not review the number of archive copies, only backup ones.

However, using this sysparm on an SMS managed data set will sever the relationship between DFSMS and the data set. The result is that all future CA-Disk processing of the data set (including MERGE and IXMAINT) will be done using only CA-Disk rules based upon the expiration date calculated from this sysparm value (i.e., MERGE will ignore its Management Class attributes when determining whether or not to merge the data set based upon level 1 days).

RESPRIEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RESPRIEX user exit. See the user exit description for "RESPRIEX — Screen Restore Requests" on page 237 for more information.

RESPRISQ12 - This sysparm allows you to determine which DSNINDEX records can be candidates for Restore and Recover functions when using the Dual Primary Volume feature. the value specified for this sysparm indicates which DSNINDEX record (P1 or P2) will be selected, or if either record could be selected and which has precedence. the values for this sysparm are:

TABLE 38. Possible Values for RESPRISQ

Value	Definition
1	This value will select only DSNINDEX records that are not flagged as Primary2 records. Users who keep the P1 volumes onsite will normally use this setting for onsite restores.
12	This allows either P1 or P2 DSNINDEX records to be selected. The P2 record will be used only if the P1 record does not exist (default).
2	This selects only DSNINDEX records that are flagged as Primary2 records. Users who send the P2 volumes offsite will mormally use this setting for offsite restores.
21	This allows either P1 or P2 DSNINDEX records to be selected. The P1 record will be used only if the P2 record does not exist.

When using Dual Primary Volumes, both Primary1 and Primary2 are counted as the same version number unless either is bypassed for selection.

Note: If you are not using the Dual Primary Volume feature, changing the sysparm value to a non-default value may cause no DSNINDEX records to be selected and your data set recovery to fail.

RESRETPDnnnnn - This sysparm controls the number of days that a restored data set is to be exempt from archive disposition, and can be specified with a 5-digit numerical value. The default "grace period" is 00030, but is activated only if sysparm RETEXCLD is specified with a value of Y.

RESTCTLGy - This sysparm can be specified with a value of N to suppress catalog referencing during preparation of the Restored Data Set report. The default value of Y will cause CA-Disk to do catalog locates for every data set restored.

Note: If a large number of data sets are being restored (for example, volume recovery), checking the catalog incurs significant overhead. If this information is not needed on the report, change the value to N to cause significant performance improvements in report processing.

RETAFTEXmmmmmmmm - Specify this sysparm to provide the 8-character name of the user-supplied exit module to receive control after data set disposition determination by CA-Disk. See the user exit description for "RETAFTEX — Retention Control Data Set Screening (Post-validate)" on page 238 for more information.

RETDSNCDccasc - This sysparm controls retention control at the data set level, based on the creation date. Specify for "cc" the year code that must be matched to the data set's expiration year (code). Specify for "asc" the desired archive, scratch and catalog actions to be taken if the data set's create date plus (ddd) days (from the expiration date field) has expired.

RETDSNCLccas - This sysparm controls retention control at the data set level, based on catalog status. Specify for "cc" the year code that must be matched to the data set's expiration year (code). Specify for "as" the desired archive and scratch actions to be taken if the data set is not cataloged.

RETDSNDFccddd - This sysparm is used only within the data set level control processing option of retention control. Specify for "ccddd" the default expiration date that is to be given to all data sets that have a zero expiration date field. This control code and retention period are then used to control the analysis and processing as called for by the three data set level parameters.

RETDSNLUccasc - This sysparm controls retention control at the data set level, based on last used date. Specify for "cc" the year code that must be matched to the data set's expiration year (code). Specify for "asc" the desired archive, scratch and catalog actions to be taken if the data set's last used date plus (ddd) days (from the expiration date field) has expired.

RETEXCLDn - Specifying this sysparm with a value of Y causes a "grace period" record to be placed in the RETEXCLD subfile each time a data set is restored. The presence of such a record prevents retention control from automatically disposing of a restored data set until the grace period of 30 days has elapsed. The default value of N disables this facility. The default grace period is controlled by sysparm RESRETPD. The grace period is checked for implicit data sets specified in SELECT or FIND statements with patterns.

The record inserted into the RETEXCLD subfile contains the data set name but not the volume to which it was restored. A data set restored to a different volume is regarded as the same data set, and will be exempt from retention control processing. This enables customers using pooling products such as DFSMS or CA-Allocate to redirect their data sets to different volumes or pools during RESTORE or MOVE, and still delay their eligibility for rearchival.

Note: If you are running in a non-pooling environment or have many data sets with duplicate names, you can decide to modify this process. The default processing can be changed by modifying the key to the RETEXCLD subfile in your FILEDEFN member in PARMLIB. Changing the key length to a value of 50 causes the target volume of the RESTORE to be part of the key. Doing so allows rearchival, but bypasses the RESRETPD of data sets moved restored to a different volume.

RETEXCMDn - This sysparm only has effect when sysparm RETEXCLD is specified with a value of Y. The default value of N indicates that users cannot override the restore default grace period (specified by sysparm RESRETPD) with the RETPD and EXPDT restore command parameters. If a value of Y is specified for this sysparm, the RETPD and EXPDT command parameters will override the default grace period.

The RETPD and EXPDT command parameters are not documented for the RESTORE and DRESTORE commands. They can be specified if this sysparm is specified with a value of Y.

RETPD is the number of days that the data set remains excluded from archive processing from the date of the restore. Specify 1 to 5 numeric digits (that is, RETPD=7). EXPDT is a Julian date that can be specified instead of RETPD. It has the same effect but is expressed in the form of a date (that is, EXPDT=1983.365).

RETGDGEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RETGDGEX user exit. See the user exit description for "RETGDGEX — Return GDG Limit to Implicit Archival for a DSN" on page 239 for more information.

RETGDGNMxxxxxxxx - When the G selection criteria is used in the retain function, different limits for different generation base names can be specified in a GDG DASD generations member of the parmlib data set. To activate the list, specify this sysparm with the name of the member.

RETMULTIn - Specify a value of Y for this sysparm if multivolume PS and DA data sets are to be processed by requests processed by the RETAIN proc. The default value is N, not to process multivolume data sets.

See sysparm "ARCMULTIn" on page 126 for controls on archives done by the ARCHIVE proc. There are no controls on archives done by the DMS proc.

RETNODSPyy - Specifying a year value in this sysparm will preclude retention control disposition of any type for any data set having the same year code with zero (ddd) in its expiration date field.

RETNOMIGyy - Specifying 99 or 98 in this sysparm precludes sequential migration to tape of any data set having the same year code with zero (ddd) in its expiration date field. The default value of 00 causes sequential migration to ignore this sysparm. Any other values of yy results in an error message being issued.

RETPREFTc - The default value of C will cause CA-Disk to use the creation date for the data set from the VTOC entry for retention date calculation. Specify a value of U to have the retain volume specification type R selection criteria use the last use date from the VTOC entry for retention date calculation.

RETPRIEXmmmmmmmm - Specify for "mmmmmmmm" the name of a user-supplied exit module to receive control prior to inspection of the DSCB during retention control processing. See the user exit description for "RETPRIEX — Retention Control DSN Screening (Pre-validation)" on page 239 for more information.

RETRETPDnnnnn - When data sets are selected to be archived during archive or backup processing, the 5-digit value of this sysparm is added to the current date to designate the archive retention period. This value is used unless an overriding parameter has been specified on the ARCHIVE or BACKUP command. The default value of this sysparm is 00030, to keep each data set in the archives for 30 days. See also sysparm "EXPDTUSEn" on page 146.

For related information, see "Year 2000 Considerations" on page 6 in the User's Guide.

RLSEAFEXmmmmmmmm - Specify the 8-character name of the user-supplied exit module to receive control after a data set has been selected for idle space release processing. See the user exit description for "RLSEAFEX — Idle Space Release Screening Exit" on page 240 for more information.

RLSEDIAGn - Specify a value of Y to cause a message to be printed stating why a data set was bypassed for Idle Space Release. The default value of N suppresses such messages.

RLSEMAXSnnnnn - This sysparm is used to specify the maximum number of tracks allowed when resetting the secondary space allocation for any given data set. The default value is 99999 tracks. Since the resetting of secondary allocation is based on a percentage of the new primary allocation, this allows the user to specify a ceiling secondary allocation value on data sets with a large primary allocation.

Note: This value is in tracks. Should the data set be allocated in cylinders, this value will be rounded up to the next whole cylinder, thus actually allocating more space than the maximum amount specified by this sysparm.

Example: On a 3350, there are 30 tracks per cylinder. The user specifies a value of 00045 for this sysparm. If the data set is allocated in cylinders and the secondary allocation is computed to be equivalent to 35 tracks, this amount will be converted to cylinders and rounded up to the nearest whole cylinder, which would be two cylinders. On a 3350, this is equivalent to 60 tracks, which is actually over the maximum amount specified by this sysparm.

RLSEMINSnnnnn - This sysparm specifies a minimum number of tracks allowed when resetting the secondary space allocation for any given data set. The default value is 00000 tracks. Since the resetting of secondary allocation is based on a percentage of the new primary allocation, this allows the user to specify a default secondary allocation value on data sets with a small primary allocation.

Note: This value is in tracks.

RLSESECRa - The default value, A, will cause CA-Disk to reset the secondary space allocation on all data sets (which meet all other RELEASE criteria) whenever specified via the PCTSECONDARY parameter for DSCL or through the PSSEC= or POSEC= parameters for batch. Specify a value of Z to reset only those secondary allocations with current value of zero, or a value of N to reset only those secondary allocations that currently have non-zero values.

RLSESYS1n - Specify a value of Y to allow idle space release processing of SYS1 data sets. The default value of N prevents idle space release processing of SYS1 data sets.

RLSEUNMVy - Specify a value of N to prevent idle space release processing of unmovable data sets. The default value of Y will permit idle space release processing of unmoveable data sets.

RPTCLSELxx...x - This sysparm can be used to specify the default group codes to be listed on the VSAM Cluster Detail report. Specify for the "xx..xx" string the desired groups, and the order in which they are to appear on the report. When this sysparm is not specified, a default value of U is assumed, causing use statistics to be printed.

For additional information, review Table 36, "CLDOPT Values and Descriptions," on page 144 in the User's Guide.

RPTDSBEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RPTDSBEX user exit. See the user exit description for "RPTDSBEX — Process Data Set Billing Report Records" on page 240 for more information.

RPTDSNEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RPTDSNEX user exit. See the user exit description for "RPTDSNEX — Screen Data Sets to be Reported On" on page 241 for more information.

RPTDSUEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RPTDSUEX user exit. See the user exit description for "RPTDSUEX — Data Set Utilization Report Record Screening" on page 241 for more information.

RPTLINES - This sysparm, which defaults to 058, controls the number of lines per page for the following reports:

ALLOCS	DSUTIL	LISTD	MVDICT
ATTRBS	FREESP	LISTV	PDSTAT
DISTRS	ISAMDS	MAPPER	SVDICT
DSINDX	LAYOUT	MEMBER	PDS COMPRESS
Utilization by DATE and TIME			

All other reports can be controlled via the following zap.

```
NAME ADSUT190 LINEMAX
VER 0000 058C          58 lines per page
REP 0000 0nnC          nn lines per page
```

RPTMVDEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the RPTMVDEX user exit. See the user exit description for "RPTMVDEX — Multiple Volume Dictionary Record Screening" on page 241 for more information.

RPTNMTSZxxx - Specify this sysparm with a numeric value to represent the number of reports included in the parmlib member REPORTNM. The default value is 100. If the value of this sysparm is exceeded, an error message will be issued stating that this value should be increased.

RSCHGBITn - Specify a value of Y to cause CA-Disk to set the change bit on each restored data set. Setting the change bit on each restored data sets can increase the number of data sets backed up in implicit archive runs, but can simplify the recovery of packs. The default value of N causes CA-Disk to leave the change bit set the same as before the data set was archived or backed up. See also sysparm "VCCHGBITn" on page 190.

RSUPPRESn - Specify a value of Y to suppress issuance of the RESERVE macro on the volume where the files data set resides during index maintenance functions. The integrity of the files data set cannot be guaranteed if this parameter is specified with a Y. In a JES2 environment, the user is responsible for ensuring that no other updates are occurring through a cross system enqueue package. The default value is N.

SAMSYSNMaim - This sysparm indicates the subsystem name of the CA-Vantage product. The default value is AIM. If you use a different name for this subsystem, update the value of this sysparm to reflect the new name. For more information, see "Activating the Subsystem" in the CA-Vantage Reference Manual.

SCRASYS1n - With the default value of N in effect for this sysparm, CA-Disk will not allow users to scratch any data sets prefixed by "SYS1." through its facilities. This is to prevent users from inadvertently scratching data sets critical to the operating system. To override this feature and allow CA-Disk to scratch data sets prefixed by "SYS1.", specify a value of Y.

SCRTCHEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the SCRTCHEX user exit. See the user exit description for "SCRTCHEX — Screen CA-Disk Scratch Data Set Request" on page 242 for more information.

SECARVOLarcive - This sysparm specifies a volume to be used for CA-Disk security-checking of data sets that exist in the CA-Disk archives. It is currently used only in conjunction with the CA-Disk CA-ACF2 and CA-Top Secret Security Interfaces. When authorization to process an archived data set is being checked, this volume will be passed to the security routine as the volume on which the archived data set exists. The default volume is ARCIVE.

SECFRVOLn - The default value of N for this sysparm passes the dummy volume (specified by sysparm SECARVOL) to the security routine during restore processing. To pass instead the volume from which the data set was originally archived, specify a value of Y.

SECURAEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the SECURAEX user exit. See the user exit description for "SECURAEX — Security Validation Exit, Post-CA-Disk Decision" on page 242 for more information.

SECURCMDn - To activate the security validation of CA-Disk commands, specify Y as the value of this sysparm. The default value of N indicates that no security checking is to be done on CA-Disk commands. This feature is active only when the ACF2, Top Secret or RACF supports are active. Review the installation procedures for these features (beginning on page 26 in the Installation Guide) for more information about this feature.

SECURDRSaaaaaaaa - This sysparm allows the user to specify the level of authority CA-Disk will check for in a deferred RESTORE request. Authority levels of OUTPUT, SCRATCH, and ALLOCATE can be specified. The default is OUTPUT. If SECURDRS is set to OUTPUT, and the SCRATCH parameter has been specified on the command, CA-Disk will check for both OUTPUT and SCRATCH authority to the data set. OUTPUT, ALLOCATE and SCRATCH are internal CA-Disk authorization levels. They correspond to authorization levels of the security packages as described below.

TABLE 39. SECURDRS Authorization Levels

CA-Disk	ACF2	Top Secret	RACF
OUTPUT	WRITE	UPDATE	UPDATE
SCRATCH	ALLOCATE	ALL	ALTER
ALLOCATE	ALLOCATE	CREATE	ALTER

Note: If you have RACFALLO set in your environment (see "RACFALLOa" on page 160), and you want to have that setting honored for deferred restore, then set SECURDRS to a value of ALLOCATE.

SECURFILn - To activate the security validation of access to subfiles of the files data set, specify a value of Y for this sysparm. The default value of N indicates that no security checking is to be done. This feature is active only when the ACF2, Top Secret or RACF supports are active. Review the installation procedures for these features (beginning on page 26 in the Installation Guide) for more information about this feature.

SECURLIMnnnn - This sysparm can be used to limit the number of security violations permitted by the CA-Disk Security Interface. The default value of 0000 will allow CA-Disk to continue processing, regardless of the number of violations detected by the Security Interface.

Specify a non-zero 4-digit value for this sysparm to have CA-Disk issue message 2582 and abend user 913 if the CA-Disk Security Interface detects more violations than the value of this sysparm. Security violations generated by system operations, such as OPEN or SCRATCH, will not be counted by this processing, and will not cause the message 2582 and abend to occur.

SECURLOGy - In each CA-Disk Security Interface, CA-Disk queries your security package for authority to process a resource. CA-Disk then examines the return code from your security package. If authorized, CA-Disk

continues with the processing of that resource. If not authorized, CA-Disk prints an appropriate message and bypasses that resource. Either way, CA-Disk continues processing the next resource as required by the CA-Disk commands.

In addition to the CA-Disk authorization failure message, by default each security package issues its own authorization failure message to the system log, the security console, and to the TSO user or job log. It can also create SMF records for a report writer indicating access violations. If you want to inhibit CA-Top Secret or RACF from issuing their own authorization failure messages or creating SMF records during queries from their respective CA-Disk Security Interfaces running as APF-authorized tasks, specify this sysparm with a value of N.

To prevent system abends, if CA-Disk is running as a non- APF-authorized task, such as under most TSO and ISPF sessions, CA-Disk will ignore the setting of this sysparm and allow system logging.

This control does not apply to users of the CA-ACF2 Security Interface.

Regardless of the value of this sysparm, each security package will continue to issue its own authorization failure messages for authorization failures in CA-Disk processing that are not part of the CA-Disk Security Interfaces, or for authorization failures in non- APF-authorized tasks, such as under most TSO and ISPF sessions.

SECURPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the SECURPEX user exit. See the user exit description for "SECURPEX — Security Validation Exit, Pre-CA-Disk Decision" on page 243 for more information.

SECURRPTn - To activate security validation during CA-Disk report generation (including the reports that list the ARCHVOLS records), specify a value of Y for this parameter. The default value of N indicates that no security checking is to be done. This feature is active only when the ACF2, Top Secret or RACF supports are active. Review the installation procedures for these features (beginning on page 26 in the Installation Guide) for more information about this feature.

SECURVOLy - If this sysparm is specified with the default value of Y, and you have sysparm RACFSUPP or TOPSSUPP specified with a value of Y, and you have the RACF or CA-Top Secret DASDVOL class active, CA-Disk will query your security package for volume-level (SCRATCH operations only) authority during data set-oriented functions and volume-oriented functions.

If you specify this sysparm with a value of N, or you do not have sysparm RACFSUPP or TOPSSUPP specified with a value of Y, or you have the RACF or CA-Top Secret DASDVOL class inactive, CA-Disk will not query your security package for volume-level (SCRATCH operations only) authority during data set-oriented functions and volume- oriented functions.

The CA-Disk CA-ACF2 interface always queries ACF2 for volume-level (SCRATCH operations only) authority during data set-oriented functions and volume-oriented functions.

SECUSUPPn - The default value for this sysparm is N which, in conjunction with sysparm PASSWORD, causes CA-Disk to bypass security processing for all password-indicated non-VSAM user data sets when they are encountered. This avoids prompting of the operator for the passwords to password-indicated non-VSAM user data sets.

If this sysparm is specified with a value of Y, CA-Disk will not bypass security processing for password- indicated non-VSAM user data sets. This will allow the security package SECURE to check authorization in OPEN of user data sets if you have SECURE installed. It will also allow the operator to be prompted for passwords in OPEN of user data sets if you do not have SECURE installed.

If you have the security package SECURE installed, you should specify this sysparm with a value of Y. You should also consider specifying sysparm PASSNEWN. See "Activating CA-Disk Security Features and Interfaces" on page 25 in the Installation Guide.

SELECTREy - The intention of this sysparm is to help prevent runaway jobs from occurring. With the default of Y specified, DSCL requires that at least one SELECT statement be included on every SCAN command. (This sysparm has no effect on the FMS PROC). Regardless of its value, no SELECT statements are required for the FIND command. However, if you specify a value of N and fail to provide a SELECT statement on the SCAN command, CA-Disk will attempt to process ALL data sets found by the SCAN. This can allow runaway jobs to occur! For example, if a user unintentionally runs:

```
SCAN REALVOLS
    DELETE
```

Note: For other related sysparms, refer to the descriptions of "DSNSELREn" on page 141 and "VOLSELREn" on page 190.

SHRNKFDTdmsosfdt - This sysparm is used with the CA-Disk data compression technique number 1, which uses the CA-Compress data compression product to compress the CA-Disk archives. (See also sysparms "DCCOMPTCn" on page 135 and "DCDATACPn" on page 136.)

This sysparm provides the name of the CA-Compress FDT used to compress the CA-Disk archive data sets. The default value is "DMSOSFDT". If you are using a different FDT name, specify this sysparm with your FDT name. For more information on this subject, see "Activating Data Compression" on page 57.

SHRNKSYSxxxx - Supply the name of your CA-Compress for MVS subsystem. The default is blanks. When this sysparm contains a value the CA-Compress subsystem is queried for every VSAM data set to determine if it is under the control of CA-Compress. If the data set is under the control of CA-Compress, its attributes and FDT table entry are saved at archive/backup time and the data set is archived/backed up in its compressed format and CA-Disk compression is bypassed. At scratch time the data set is deleted from CA-Compress. At restore time, it is re-defined to CA-Compress.

SMFCDATDnnn - The displacement of the close date in the SMF record is indicated as the 3-digit value for this sysparm. The default value is 006.

SMFCTIMDnnn - The displacement of the create time in the SMF record is indicated as the 3-digit value for this entry. The default value is 002.

SMFDFLAGnnn - The displacement of the flag byte in the SMF record that indicates a DASD data set is specified by the 3-digit value of this sysparm. When 000 is specified, all records will be treated as DASD. The default value is 038.

SMFDSNMDnnn - The displacement of the data set name in the SMF record is indicated as the 3-digit value for this entry. The default value is 064.

SMFJDATDnnn - The displacement of reader/interpreter job date in the SMF record is indicated as the 3-digit value for this entry. The default value is 026.

SMFJTIMDnnn - The displacement of reader/interpreter job time in the SMF record is indicated as the 3-digit value for this entry. The default value is 022.

SMFTYP14nnn - The value specified for this sysparm identifies the TYPE 14 data set activity record. The value must be specified from the range of 001 through 255. The default value is 014.

SMFTYP15nnn - The value specified for this sysparm identifies the TYPE 15 data set activity record. The value must be specified from the range of 001 through 255. The default value is 015.

SMFVLSRDnnn - The displacement of the volume serial in the SMF record is indicated as the value of this parameter. The default value is 182.

SMSACSDFn - Provide default class names and application ID for VSAM define of ACS routine variables to support security packages (currently RACF and TopSecret).

- When this system parameter is specified with the default value of N, ACS routine variables are not provided.
- When this system parameter is specified with a value of Y, ACS routine variables are provided.

For more information see ACSDEFAULTS in the IBM manual "MVS/ESA Storage Administration Reference" and the IGDSMSxx member of SYS1.PARMLIB.

SMSALLOCn - CA-Disk has the capability of passing SMS Data Class, Management Class, and Storage class names to the allocation process for both VSAM and non-VSAM data sets.

- For RESTORE or RECOVER, the source of these names is either the DSNINDEX record being restored, or the STORCLAS=, MGMTCLAS=, and DATACLAS= command parameters.
- For MOVE or COPY, the source of these names is either the data set being moved or copied, or the STORCLAS=, MGMTCLAS= and DATACLAS= command parameters.

When this sysparm is specified with is default value of “N”, CA-Disk will not pass the class names to the allocation process. This recommended unless specific ACS rules are written to change class names and/or the STORCLAS=, MGMTCLAS=, DATACLAS= parameters are coded during a RESTORE, RECOVER, MOVE, or COPY job.

When this sysparm is specified with a “Y”, CA-Disk will pass, if available, the class names as described above to the allocation process. This is similar to specifying the class names in JCL, IDCAMS define, or option M from the ISPF 3.2 panel. The difference is that the target volser is always passed into the allocation.

Refer to the following locations for more information:

- IBM's Storage Administration Reference regarding coding of ACS rules.
- “DATACLAS=” on page 256 in the User’s Guide for the specification of DATACLAS=, MGMTCLAS=, or STORCLAS= parameters during RESTORE processing.
- “DATACLAS=” on page 272 in the User’s Guide for the specification of DATACLAS=, MGMTCLAS=, or STORCLAS= parameters during RECOVER processing.
- “DATACLAS=” on page 367 in the User’s Guide for the specification of DATACLAS=, MGMTCLAS=, or STORCLAS= parameters during MOVE processing.
- “DATACLAS=” on page 369 in the User’s Guide for the specification of DATACLAS=, MSMTCLAS=, or STORCLAS= parameters during COPPY processing.

SMSBYBKVn - When this sysparm is specified with a value of 'Y', the user has Storage Administrator authority, and the Management Class assigned to a data set is defined to allow 13 backup versions, CA-Disk will ignore the number of backup versions defined in the Management Class. The number of backup versions to keep will be based on normal CA-Disk parameters.

When this sysparm is specified with the default value of 'N', the number of backup versions defined in the management class will be honored.

SMSCACHEn - When this sysparm is specified with a value of ‘Y‘, CA-Disk builds incore tables for SMS construct definitions. Doing so reduces the amount of time spent on SMS subsystem requests. The default value of ‘N’ does not create any incore tables.

SMSCPROCy - When this system parameter is specified with the default value of Y, CA-Disk processes SMS data sets.

When this system parameter is specified with a value of N, CA-Disk does not process but bypasses all SMS data sets.

SMSDPROCy - When this system parameter is specified with the default value of Y, use the SMS default management class when a management class is not indicated. Refer to sysparm SMSIPROC to see what happens when the default management class is invalid.

When this system parameter is specified with a value of N, use CA-Disk processing rules when the management class is not present for an SMS-controlled data set.

For related information, review Figure 1, “SMS Hierarchy flowchart,” on page 22 in the User’s Guide.

SMSFORCEn - This system parameter can be specified with 1 of 3 values to indicate how to process SMS data sets when the management Class or management class default is invalid or both are blank.

TABLE 40. Possible Values for SMSFORCE

Value	Definition
N	Default, stops processing when SMS is not active.
Y	Process SMS data sets as non-SMS when SMS is not active.
B	Bypass SMS data sets when SMS is not active.

SMSGBRST99 - When processing a DFSMS data set by its management class, this system parameter is used to control the retention period of Guaranteed Backup copies. When specified with the default value of 99, CA-Disk uses the number of data set backup copies defined in the Management Class to control the Guaranteed Backup copies.

In a situation where the data set has not been modified but the Guaranteed Backup frequency selects the data set for backup multiple times before archival, the number of copies will be increased each time a Guaranteed Backup is taken. This could cause a problem where previous changed versions are deleted to add an unchanged Guaranteed Backup. If this is a concern, setting this sysparm to a value from 1-98 will flag the previous Guaranteed Backup copy of the same data set name as non-SMS managed and will change the retention period to today's date plus the number of days specified in this sysparm value.

See also SMSIPROC and SMSDPROC for their relationship to processing when no management class is defined for this DFSMS controlled data set name.

SMSIPROCy - When this system parameter is specified with the default value of Y, use CA-Disk rules to process the data set when the SMS Management Class is not indicated or is invalid. Refer to sysparm SMSDPROC for related information.

When this system parameter is specified with a value of N, bypass the data set when the default management class is invalid.

SMSLOWRCn - When this sysparm is specified with a value of Y, the DSCL Explicit FIND command will not notify the user (via a higher condition code) that the SMS MC disallowed the Backup or Archive. When this sysparm is specified with its default value of N, Backup and Archive will continue to notify the user (with a condition code of 4) whenever the SMS MC disallows the explicitly requested data set name.

This sysparm has no effect on DSCL Implicit commands.

SMSMCBYPn - When this sysparm is specified with a value of Y or B, and the user is a storage administrator (see sysparm SMSSTGAD), CA-Disk processes a data set without regard for the SMS Management Class attributes associated with the data set. Using a value of B will only bypass SMS control for BACKUP and ARCHIVE processing. When specified with the default value of N, CA-Disk honors the SMS Management Class attributes.

SMSSGBYPn - When this sysparm is specified with a value of Y, and the user is a storage administrator (see sysparm SMSSTGAD), CA-Disk processes a data set without regard for the SMS Storage group attributes associated with the data set. When specified with the default value of N, CA-Disk honors the SMS Storage group attributes.

SMSSTGADddd - This sysparm is used to specify the FACILITY resource class profile name (44 bytes maximum). CA-TOP SECRET users should note that TOP SECRET will actually query your IBMFAC class instead of the FACILITY class. The default value is STGADMIN.DMS.STGADMIN. The profile name is used by CA-Disk to determine if a user is a storage administrator. CA-Disk considers all users who have read authority to the profile name specified by this sysparm to be storage administrators.

If the profile does not exist, or the facility class is not active, the user will be considered a storage administrator. CA-Disk uses an IBM SAF call to make this determination. This sysparm is used only during CA-Disk DSCL BACKUP, for SMS controlled data sets whose Management class is set to only allow storage administrators to issue Backup commands.

SMSUPDATy - This sysparm is used to control the updating of the DSNINDEX record in the files data set after a RESTORE or RECOVER operation. It also enhances the restart capabilities of failed recover jobs if left at its default value. For further details, see the criteria descriptions of "RESDT — Last RESTORE Date (date format)" on page 123 in the User's Guide and "RESTIME — Last RESTORE Time (time format)" on page 124 in the User's Guide.

SMSUPDAT defaults to Y, to perform the update in every case. This includes driving the User Exit (DSNXUFEX) for updating DSNINDEX.

If N is specified, the update is never done, and the User Exit is not driven. This setting disables DSCL Recover restart capabilities by eliminating the validity of RESDT. For details, refer to the criteria description for "RESDT — Last RESTORE Date (date format)" on page 123 in the User's Guide.

If S is specified, the updates are only done for SMS managed data sets. The User Exit is only driven if the update is performed.

If you do not need the SMS information update (for example, if you want faster operation at a disaster recovery site), specify SMSUPDAT with a value of N to bypass the update.

SORTPADDy - If the installation sort routine allows a sort key length greater that the actual record length for variable-length records, you can specify a value of N for this sysparm to prevent CA-Disk from padding all report records to a length equal to the key. The default value of Y permits the padding of the report records.

SPACECEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the SPACECEX user exit. See the user exit description for "SPACECEX — Space Conversion Exit (Non-VSAM)" on page 245 for more information.

SPACEPRInnn - When archiving to disk, you must specify a primary space allocation value for the archive data set. The default value of 999 causes CA-Disk to set the primary quantity equal to 1/16 of an assumed backup tape. You can specify a value from 1 to 998, which represents the number of megabytes to allocate.

SPACESECnnn - When archiving to disk, you must specify a secondary space allocation value for the archive data set. The default value of 999 causes CA-Disk to set the secondary quantity equal to 1/16 of an assumed backup tape. You can specify a value from 0 to 998, which represents the number of megabytes to allocate.

SPFAUNITnnnnnnnn - Specify this sysparm to change the unit type allocated for the work data sets used in ISPF online reporting. The default unit name is SYSALLDA.

SPFDSNAMnn....nn - This sysparm is used to specify a system-wide data set to be used to store ISPF report definitions. This is the same data set name that is specified when you create the report definition data set, either using the ISPF online facility or the batch utility provided (review the topic "Overview of ISPF Online Reporting Facility" on page 439 in the User's Guide for more information on creating this data set). The default data set name can be overridden both on the primary online report panel and in any batch job that uses the ISPF reporting feature. It is used only with the ISPF support.

SPFENQQNnnnnnnnn - When saving JCL in the CA-Disk ISPF support, CA-Disk ENQs with a DISP=OLD on the data set in which the JCL is to be saved. The QNAME used is SPFEDIT. If you desire to change the QNAME, specify this sysparm with another name, using a maximum of 8 characters.

SPFENQTYr - When saving JCL in the CA-Disk ISPF support, CA-Disk ENQs with a DISP=OLD on the data set in which the JCL is to be saved. If you do not want CA-Disk to do the ENQ, specify this sysparm with a value of N.

Note: Specifying this sysparm with a value of N is an integrity exposure in that two or more persons could possibly update the same data set, thus destroying data.

SPFENQWTnn - When processing partitioned data sets, IBM's ISPF editor and linkage-editor routines use enqueue conventions beyond the standard data set name enqueue under QNAME of SYSDSN. Therefore, while processing partitioned data sets, CA-Disk also issues an exclusive enqueue on the data set name with QNAME of SPFEDIT (this QNAME value can be changed with sysparm SPFENQQN). If the record format of the data set is also undefined, an exclusive enqueue on the data set name and QNAME of SYSIEWLP is done as well. These enqueues are done to prevent a data loss that can occur if another task is processing the data set at the same time CA-Disk is processing it.

Specify this sysparm with a one- or two-digit value to indicate the number of seconds the enqueue routine will wait for the exclusive enqueues. The default value is four seconds. If the enqueues are not obtained within the specified time limit, the data set will not be processed. If a value of zero is specified, no enqueue processing for the QNAMEs of SPFEDIT or SYSIEWLP will be done.

This processing is not affected by the CA-Disk user exit USERSENQ.

Note: Specifying this sysparm with a value of zero is an integrity exposure, in that two or more tasks could possibly update the same data set, thus destroying data.

SPFHDGEXmmmmmmmm - This sysparm, used only with the ISPF support, specifies the module name to be invoked for the SPFHDGEX user exit. See the user exit description for "SPFHDGEX — ISPF Report Heading Exit" on page 246 for more information.

SPFHILVNnnnnnnnn - Specify this sysparm to use a high-level node (maximum of 8 characters) in the work data sets used in ISPF online reporting. The default is spaces.

SPFORDEXmmmmmmmm - This sysparm, used only with the ISPF support, specifies the module name to be invoked for the SPFORDEX user exit. See the user exit description for "SPFORDEX — ISPF Online Reporting Report Definition Authorization" on page 247 for more information.

SPFORSEXmmmmmmmm - This sysparm, used only with the ISPF support, specifies the module name to be invoked for the SPFORSEX user exit. See the user exit description for "SPFORSEX — ISPF Online Reporting Authorization" on page 247 for more information.

SPFORSPCnnnn - Specify this sysparm to change the number of blocks allocated for the REPORTS ddname in ISPF online reporting. The default is 150 blocks.

SPFORSPEnnnn - Specify this sysparm to change the number of blocks allocated for the SPFRPTS ddname in SPF online reporting. The default is 100 blocks.

SPFRPTDTy - This sysparm can be specified to suppress printing of detail report lines for the ISPF online reporting facility. The default value of Y causes all detail lines to be printed. If you set the value to N, CA-Disk will print only subtotal and grand total lines. This feature is useful in place of the volume summary reports if you want to limit the totals to a subset of the data sets on the volume (with SELECT or EXCLUDE commands). If you want to override this parameter in the JCL with the SYSPARMS dd statement, you must do it in the print step (that is, PRINT.SYSPARMS DD *). It is used only with the ISPF support.

SPFSYSINnnnnnnnn - This sysparm can be specified to provide the ddname to use in place of SYSIN in the ISPF support only. The default is SPFSYSIN. To change the ddname, specify this sysparm with another name (maximum of 8 characters).

SPFUSRIDn - This sysparm applies only to the ISPF support. It can be specified with a value of Y to restrict ISPF users to archive and restore only those data sets that begin with their user ID. You can exempt selected users from this restriction by placing their user IDs in the TSOUSERI member of the parmlib data set. The default value for this sysparm is N, which allows all users access to all data sets.

SVCNODMSnnn - Specify for "nnn" the decimal number of the CA-Disk SVC as installed. The default value is 244.

SVDCATLGy - This sysparm can be specified with a value of N to suppress catalog referencing during the preparation of the single volume dictionary report. The default value of Y will cause CA-Disk to do catalog locates for every data set listed on the SVD report to provide the report item indicating if the data set is cataloged or not. If this information is not needed on the report, change the value to N to cause significant performance improvements in report processing.

SYSOUTEXmmmmmmmm - Unlike other CA-Disk user exits, the SYSOUT exit is not activated by specifying a sysparm. The module name must be superzapped into the CA-Disk output writer module. This entry in the sysparm

section is maintained solely as a reference aid. See the user exit description for "SYSOUTEX — SYSOUT Exit" on page 247 for more information.

SYSPARMOn - CA-Disk provides the capability for users to override default sysparm values and entries in the SYSPARMS member of the parmlib data set at execution time via the //SYSPARMS dd statement. This provides the flexibility of tailoring parameter values for various executions without having to update the parmlib data set.

The default value of N prevents all users from overriding sysparms. To allow all users to override some or all sysparms, or to allow only authorized users this ability, specify this sysparm with a value of Y. (See "Resources Protected by the CA-Disk Base System" on page 313.)

For a further discussion on overriding sysparms, see "Overriding Sysparms Instream" on page 106.

SYSPARMSn - This sysparm controls the display of system parameters on the message dd statement during CA-Disk executions. It gives the user the option of seeing what sysparms were active during the execution.

The default value of N indicates no display is to be generated. Other valid values are:

TABLE 41. Possible Values Available for SYSPARMS

Value	Meaning
O	print only those sysparms overridden by SYSPARMS DD
S	print only those sysparms supplied by user
D	print only the default sysparms
B	print all sysparms in effect

TABLESDDparmlib - Specifying this sysparm instructs CA-Disk to use that DDname for retrieving the parmlib entries specified by TABLES= on the DSCL FIND, DSCL SELECT/EXCLUDE and the FMS SELECT/EXCLUDE commands. The value can be specified as any 1- to 8-character DDname. The default is parmlib, which causes CA-Disk to use the PARMLIB DDname as normal.

Note: If the user specifies any entry which uses tables, (either Sysparms, User Exits or the TABLES= parameter), then the member containing the table must be found in the DDname specified, or the job will terminate.

The use of this sysparm requires that the user add a DD statement in the jobstep equal to that of this sysparm. Figure 12 below attempts to show the relationships required for this sysparm to work:

FIGURE 12. Sample JCL Using the TABLESDD Statement

```
File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
--------------------------------------------------------------------------------
EDIT            .UTIL.CNTL(BACKUP) - 01.02                Columns 00001 00072
Command ===> _____________________________________________ Scroll ===> CSR
000300 //BACKUP   EXEC DMS,P='WORKSHOP.R90.INSTALL',Q='WORKSHOP.R90'
000400 //STEPLIB   DD  DISP=SHR,DSN=WORKSHOP.R90.LOADLIB
000500 //FILES     DD  DISP=SHR,DSN=SAMS.DISK.FILES
000600 //MSGPRINT  DD  SYSOUT=*
000700 //PARMLIB   DD  DISP=SHR,DSN=SAMS.DISK.PARMLIB
000800 //DKTABLES  DD  DISP=SHR,DSN=DISK.USER.TABLES
000900 //SYSPRINT  DD  SYSOUT=*
001000 //SYSIN     DD  *
001100  SCAN REALVOLS
001200   SELECT TABLE=CICS01
001300  BACKUP RETPD=180
001400 //SYSPARMS  DD  *
001500 TABLESDDDKTABLES
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
--------------------------------------------------------------------------------
EDIT             DISK.USER.TABLES(CICS01) - 01.00          Columns 00001 00072
Command ===> _____________________________________________ Scroll ===> CSR
000100 CICSTST./
000200 CICSHTP./
4 ■                        Aa   A TCPIP        R 4  C 15        17:58   5/27/98
```

See "Alternate DDname for User Tables" on page 478 for further information.

Note: The value you choose for this sysparm should not be any DDname that is already in use by CA-Disk.

TAOPPOEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the TAOPPOEX user exit. See the user exit description for "TAOPPOEX — Archive Tape Processing Exit (Post-Open)" on page 248 for more information.

TAOPPREXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the TAOPPREX user exit. See the user exit description for "TAOPPREX — Archive Tape Processing Exit (Pre-Open)" on page 248 for more information.

TAPCTLEXmmmmmmmm - Specify this sysparm to provide the module name, other than ADSTH016, to be invoked for the TAPCTLEX user exit if more than one tape management system is used. For related information, see the sysparm description of "TMSCTLEXmmmmmmmm" on page 184 and the user exit description for "TAPCTLEX — Tape Management System Control Interface Exit" on page 249.

TAPEDLEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the TAPEDLEX user exit. See the user exit description for "TAPEDLEX — Archive Volume Being Deleted from Archives" on page 249 for more information.

TAPEFEETnnnn - The 4-digit value specified for this sysparm is the number of feet to be used for writing data sets on each reel of tape. In the event that a reel that has less than the specified number of feet is mounted, it is possible that a data set will overflow onto a second reel. However, the following data set will be archived to the beginning of another reel, precluding the possibility that more than one reel of tape would have to be passed to accommodate a subsequent restore request. The default value for this sysparm is 2300.

TAPEFSCRn - For systems with DFP 1.1 and above, specifying a value of Y allows CA-Disk to bypass expiration date and security checking when writing to a tape — in other words to "force" its use as a valid scratch tape. The standard operator prompt for permission to write on a date- protected tape will not be issued. Similarly, the operating system will not abend the job when a security-protected tape is mounted for output. This can be particularly useful when CA-Disk tape pools are used. (Tapes are recycled, and by definition, CA-Disk knows that it is okay to write on them again.) The default value is N, however, since using this facility with general scratch tapes represents the loss of a safety check, not to mention the data, if wrong tapes are mounted.

On systems without DFP 1.1 or above, you should keep in mind that IBM does not allow you to change tape data set names after they have been protected. You must initialize the tapes to change the data set names. This is documented in IBM's OS/VS Tape Labels manual.

For example, if your archive tape data set name is DMS.ARCHPRIM and your backup tape data set name is DMS.BACKPRIM, you must initialize your protected archive tapes before they can be recycled and used for backup tapes.

Note: If your tapes are managed by CA1, do not specify a value of Y, or all CA1 processing will be bypassed.

TAPEPOOLn - Specify a value of Y to cause CA-Disk to take output tapes for archiving out of the TAPEPOOL table in the DMSPOOLS subfile. The default value of N will call for the operator to mount any available scratch tape.

TAPEPROTn - When this sysparm is specified with the default value of N, CA-Disk does not password-protect archive data sets on tape and allows normal security checking when opening them.

When this sysparm is specified with a value of Y, CA-Disk sets the password-indicator on the tape label, but does not generate any passwords. Also, CA-Disk bypasses all security checking on the archive output tape volumes.

If this sysparm is specified with a value of Y, sysparm TAPEFSCR must also be specified with a value of Y to allow CA-Disk to reuse tapes. (At last report, this was incompatible with the CA1 tape management system from Computer Associates.) If a CA-Disk archive tape was created with sysparm TAPEPROT specified with a value of Y, you must relabel the tape when it is released from the CA-Disk archives to turn off the password-indication of the tape label. This applies to CA-Disk as well as other applications.

For password-protection of archive data sets on disk, review the sysparm description for “DISKPROTy” on page 138.

TAPEPRTRy - Specify a value of N for this sysparm to cause CA-Disk to use normal security checking on reading archive tapes. CA-Disk will prompt the operator for the password for archive tapes created with sysparm TAPEPROT specified with a Y.

The default value of Y will cause CA-Disk to bypass security checking when reading archive tapes. CA-Disk will read archive tapes created with the sysparm TAPEPROT specified with a Y without operator intervention.

TAPEVVERy - The default value of Y causes CA-Disk to use the IBM DFP volume verification exit to verify that each tape mounted for output is indeed a scratch tape, (not in the CA-Disk archives already) and, if it is not, to

dynamically request another tape. All of this can be done without damaging the mounted tape, and processing can continue.

If vendors of other products used in your installation state that they do not support this exit, this sysparm should be specified with a value of N to deactivate its use. When a "wrong tape" is mounted under these conditions, CA-Disk will issue appropriate messages and terminate processing in an effort to preserve the data already on the tape.

TAPMFEETnnnn - This sysparm specifies the number of feet to be used for writing data sets on each tape reel in the sequential migration to tape function. The default value is 2300. If a reel that has fewer than the specified number of feet is mounted, a data set could overflow onto a second reel.

TERMCLERy - The default action in TSO is to clear the terminal screen before displaying the output. Specifying a value of N will cause CA-Disk to not clear the terminal screen before displaying the output.

TESTMSGSn - Debugging purposes only and is inactivated by default, which provides no debugging information. Specifying a value of Y is not recommended unless requested by Technical Support.

TMSCTLEXmmmmmmmm - Specify this sysparm to provide a module name for the TMSCTLEX user exit. See the user exit description for "TMSCTLEX — Tape Management System Control Interface Exit" on page 249 for more information.

Note: In order to activate the RMM support, this sysparm must be specified with a value of ADSTH016. See sysparm "TAPCTLEXmmmmmmmm" on page 182 if more than one tape management system is used.

TOPSALWZy - Two circumstances cause a query of CA-Top Secret for authorization regardless of the setting of the RACF- indicator bit:

1. OPEN, SCRATCH and RENAME processing under most versions of IBM operating systems (this feature is called "always call").
2. data sets cataloged in ICF catalogs.

Under some operating systems, data sets not cataloged in ICF catalogs will query CA-Top Secret only if the RACF-indicator bit is on. For non-VSAM data sets, the RACF-indicator bit is the DS1IND40 bit (bit x'40' at offset 93 x'5D') located in the data set's format-1 DSCB.

CA-Disk security processing normally queries CA-Top Secret for authorization regardless of the setting of the RACF-indicator bit. If you do not have the "always call" feature of the operating system and you do not use ICF catalogs, specify this sysparm with a value of N. The default value is Y.

TOPSFORCn - This sysparm is active only for users who have specified sysparm TOPSSUPP with a value of Y. The default value is N, which will cause CA-Disk to abend with a user 100 abend if Top Secret is not active. Specify a value of Y to cause CA-Disk to function even if TOP Secret is not active. No Top Secret security checking will be done in this case. The RACSTAT macro is used to determine if Top Secret is active on the system.

TOPSSUPPn - Specify this sysparm with a value of Y to activate the Top Secret support. The default value is N.

TPMGMTEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the TPMGMTEX user exit. See the user exit description for "TPMGMTEX — Tape Management Interface Exit" on page 253 for more information.

TPOOLCHGn - This sysparm applies only to the ISPF support. The online facility to list the tapes in a tape pool does not normally allow the "scratch" or "in-use" status of a tape to be changed. The entry can only be deleted (by

placing a "D" beside it) and then added again as a scratch tape if desired. Specifying a value of Y for this sysparm permits you to enter an "A" beside the tape entry to change it to an available (scratch) status, or an "I" beside the entry to mark it as in-use. (Use this feature with extreme caution! Careless use can wreak havoc upon the tape pools in very short order.)

TPOOLSEQs - The default value of S for this sysparm keeps scratch tapes in the tape pool in sequence by when they were scratched. Specify a value of V to keep them in volume serial sequence. If you use the EDM interface, you should not set this sysparm to a value of V because CA-Disk can attempt to use the tape again before it has been scratched by the nightly TMSCLEAN run.

TRKOVCHGn - Specify a value of Y for this sysparm to cause CA-Disk to allow the track overflow attribute of a data set to be dropped when the target device does not support that feature (for example, 3375 and 3380 devices) but the blocksize will fit on the target track without the attribute. This sysparm applies only to the MOVE/COPY DSCL commands.

The default value of N will cause CA-Disk to bypass these data sets with a diagnostic message such that the user can decide if these data sets are to be copied.

Note: Not available when using the DFSMSdss Data Mover interface.

TSCHKAEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the TSCHKAEX user exit. See the user exit description for “TSCHKAEX — CA-TOP Secret Post-RACHECK Exit” on page 254 for more information.

TSCHKPEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the TSCHKPEX user exit. See the user exit description for “TSCHKPEX — CA-TOP Secret Pre-REACHECK Exit” on page 255 for more information.

TSODACATu - The default value of U for this sysparm causes CA-Disk to uncatalog a data set when performing an ISPF or TSO deferred archive. This action can be changed by specifying either a value of N for no catalog change, or a value of R to recatalog to the CA-Disk pseudo-volume.

TSODARESy - The default value of Y for this sysparm causes CA-Disk to reset (turn off) the change bit when doing an ISPF or TSO deferred archive without scratching the data set. This action can be changed to not reset the change bit by specifying a value of N.

TSODASCRs - The default value of S causes CA-Disk to scratch a data set when doing an ISPF or TSO deferred archive. This action can be changed to "no scratch" by specifying a value of N.

TSOTULMTnn - Specify for "nn" the maximum number of tape units that can allocated concurrently for TSO dynamic (immediate) restores. The default value is 01. When this number is reached, any subsequent TSO dynamic restore request will be denied with a message that the maximum numbered of tape units are already allocated.

TSOTUNITnnnnnnnn - Specify for "nnnnnnnn" the 1- to 8-character unit name for the class of tape devices to be used for dynamic allocation of CA-Disk archive tapes under MVS/OS. The default unit name is TAPE.

TSOUSRIDn - This sysparm applies to the TSO support. It can be specified with a value of Y to restrict users queueing archive and restore requests to only those data sets that begin with their user ID. You can exempt selected users from this restriction by placing their user IDs in the TSOUSERI member of the parmlib data set. The default value for this sysparm is N, which allows all users access to all data sets.

TSOVOLOFn - The default value of N for this sysparm causes the TSO command processors to reject requests issued against volumes that cannot be found online. Specify a value of Y to allow requests to be made against data sets that cannot be found online (they are assumed to be contained on offline volumes).

TTRULMOVn - The default value of N for this sysparm prohibits CA-Disk from moving a data set with TTR addressing to a different device type. Specifying a value of Y allows CA-Disk to move a data set with TTR addressing to a different device type.

Note: Because of the unique nature of these types of data sets, it is imperative that the user verify the integrity of the data set after the move.

UCBALLOCxx - This sysparm controls the maximum number of volumes that a CA-Disk job can have dynamically allocated at one time. This corresponds to the number of UCBs in an MVS environment. The default value is 21. Change this limit by specifying a value of from 1 to 64 for this sysparm. We recommend specifying this sysparm to a value no lower than 3 (higher if you have specified sysparm RETMULTI).

UCC1BYPSn - This sysparm is valid only when TMSCTLEX is specified with a value of ADSTH025. If you want to have CA-Disk keep running even if the CA-1 system is down, specify a value of Y. The default value of N will cause CA-Disk to abend with an appropriate message.

This sysparm is only used when the user exit program name for sysparm TMSCTLEX is "ADSTH025".

UCC1EXPRy - This sysparm is valid only when TMSCTLEX is specified with a value of ADSTH025. The default value of Y will cause CA-Disk to expire tapes in the CA-1's TMC by setting the expiration date of the tape to the current date. Specify a value of N if sysparm UCC1SCRT is specified with a value of Y.

This sysparm is only used when the user exit program name for sysparm TMSCTLEX is "ADSTH025".

UCC1EXTNnnnn - The default value of 0000 for this sysparm will cause CA-Disk (in CA-1 support) to expire a tape by setting the expiration date of the tape to the current date. If you want to allow a grace period before the tape is expired, specify a 1- to 4-digit number indicating the number of days to be added to the current date to determine the expiration of the tape in CA-1's TMC.

This sysparm is only used when the user exit program name for sysparm TMSCTLEX is "ADSTH025".

UCC1SCRTn - The default value of N for this sysparm will cause CA-Disk (in CA-1 support) to expire a tape by setting the expiration date of the tape to the current date. If you want to have CA-1 recognize the tape immediately as a scratch tape, specify a value of Y for this sysparm. A value of Y for this sysparm will supersede sysparm UCC1EXPR.

This sysparm is only used when the user exit program name for sysparm TMSCTLEX is "ADSTH025".

UCC6CPDSn - Because data sets produced by the UCC6 program product contain non-standard control blocks within some of their members, CA-Disk cannot maintain them properly.

The default value of this sysparm is N, which causes CA-Disk to reject processing for UCC6-controlled PDSs in all functions except archive, backup, restore and recover. When a UCC6 PDS is restored, it will be in standard IBM PDS format, not UCC6 format.

If you want CA-Disk to attempt to process these data sets in other functions, specify a value of Y for this sysparm.

Note: The default value of N will also prevent CA-Disk from producing the PDS status report on UCC6 PDSs. To obtain the PDS status report, this value must be specified as Y.

UNCATARCn - This sysparm determines the catalog action to be taken for any archive tape dropped (deleted) in the index maintenance functions (merge/purge/delete) processing. The default value of N causes CA-Disk to leave the catalog status unaltered. A value of Y causes CA-Disk to uncatalog any tape that is deleted from the archives. This sysparm is intended for joint use with sysparm ARCTNAME.

UNCATDELn - If you let sysparm UNCATDEL default to a value of N, CA-Disk treats DSNINDEX Julian expiration date "99000" as a never scratch date. This is equivalent to the other "never scratch" dates of 99365, 99366 and 99999.

But if you set sysparm UNCATDEL to a value of Y, CA-Disk treats Julian expiration date "99000" as a "catalog control date." That is, data sets backed up or archived with an expiration date of 99000 will be deleted automatically by the CA-Disk function IXMAINT when the data sets become uncataloged. What this means is, if your catalog goes bad or someone uncatalogs some data sets, that all backups and archives of these data sets will be deleted on the next IXMAINT run. Although you can use EXPDT=99000 for any purpose, it was intended for use with archiving generation data groups (GDGs), the generations of which are managed by the catalog. For example, to archive all GDG generations, leaving the two most recent on disk, deleting generations that roll off of the catalog, specify sysparm UNCATDEL with a value of Y and:

```
SCAN REALVOLS SELECT CRITERIA=(GDGGEN,LE,-2)
   ARCHIVE RECAT,EXPDT=99000
```

UNCATFDRn - If this sysparm is specified with the default value of N, CA-Disk does not do additional catalog checking during UNCATPSUy processing. This avoids the additional overhead caused by the catalog checking. If this sysparm is specified with a value of Y, CA-Disk does additional catalog checking prior to deleting a catalog entry during UNCATPSUy processing. The additional catalog checking is in the form of a catalog locate to determine if the catalog entry can have been created by FDRABR. Once the catalog locate has been done, the first byte of DSCB TTR information is tested for a x'80' value. If the test is true, CA-Disk assumes the entry was created by FDRABR and does not uncatalog the data set.

UNCATPSUn - If you are recataloging archived data sets to the CA-Disk pseudo-volume, or if archived data sets are being left cataloged to permit auto-restores, specify a value of Y for this sysparm to remove obsolete catalog entries during merge, index maintenance, or ixupdate processing when the last DSNINDEX record is deleted with that data set name. CA-Disk uncatalogs data sets from the pseudo-volume only if the DSNINDEX record being deleted is an archive copy (that is, the data set was scratched at the time the copy was made), and a value of R is specified for this sysparm.

When this sysparm is specified with a value of Y, CA-Disk determines if this is an "obsolete" catalog entry by verifying that this is the last DSNINDEX entry for the specific data set being deleted (that is, this copy of the data

set has expired and no other copies exist). At that time, it checks to see if the data set is still cataloged. If so, the following actions are taken:

- If it is cataloged to the CA-Disk pseudo-volume, CA-Disk uncatalogs it.
- If it is still cataloged to its original source volume or to the "staging volume" being used for auto-restores, CA-Disk issues an OBTAIN macro to see if the data set is still there. It can have been restored or re-created since it was archived and scratched). If the data set is found, the catalog is left unchanged; otherwise the entry is deleted.
- If it is cataloged to a volume other than those mentioned above, no further processing is done. The catalog entry is left unchanged.

For auto-restore, when no DSNINDEX record is found, a value of Y for UNCATPSU will cause the psuedo-volume catalog entry to be deleted. A value of N will not delete the catalog entry.

Care should be taken when using this sysparm in a shop where offline production volumes exist. For if sysparm IXMVOLONn (which is the default), is also specified the OBTAIN will fail for data sets cataloged to offline volumes. These data sets will be uncataloged. See sysparm "IXMVOLONn" on page 151 for more information.

If this sysparm was specified with a value of R, CA-Disk follows the above steps only if the DSNINDEX record being deleted is the LAST ARCHIVE copy in this files data set. No uncatalog action is done if this is a backup copy, even if it is the last DSNINDEX record of this data set name in the files data set.

If the DSNINDEX record being deleted was SMS managed at the time of creation, MERGE, IXMAINT, and IXUPDATE DELETE processing automatically assigns a value of R for that record only. This facilitates expiration date verification for deletion of SMS managed backup copies. The SMS managed backup copies remain after the archive copy has been expired for the time defined in the Retain Days Only Backup Versions (Data Set Deleted) in the Management Class assigned. See the IBM Storage Administration Reference for your release of DFP for more information about the way SMS data sets are deleted.

If this sysparm is specified with a value of X, all catalog actions for processed data sets will be bypassed during IXMAINT. This will leave the cataloged entries intact for any subsequent REBUILD processing.

UNLOADEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the UNLOADEX user exit. See the user exit description for "UNLOADEX — Process Unloaded Files Data Set Record" on page 255 for more information.

UNMOVEARn - The default value of N for this sysparm prevents CA-Disk from moving unmoveable data sets during archive/backup and restore processing. CA-Disk does not officially support unmoveable data sets and therefore cannot guarantee the integrity of the data set after processing it. To permit archive/backup and restore processing of unmoveable data sets, specify this sysparm with a value of Y.

UNMOVEVCn - The default value of N for this sysparm prevents CA-Disk from moving unmoveable data sets using the Move/Copy function. CA-Disk does not officially support unmoveable data sets and therefore cannot guarantee the integrity of the data set after processing it. To move unmoveable data sets using the Move/Copy function, specify this sysparm with a value of Y.

USEDSSIO - This sysparm controls whether or not the DFSMSdss data mover path will be used while writing ARCHVOLS, and if used, what data set organizations are to be selected. The default value of *blank* prevents CA-Disk from using the DFSMSdss data mover.

The following table represents the list of valid values available for this sysparm:

TABLE 42. Possible Values Available for USEDSSIO

Value		Sysparm Format	Meaning
	—	‘USEDSSIO ‘	Default value that forces the use of the CA-Disk data mover
HFS	—	USEDSSIOHFS	Processes only Hierarchical File System data sets
HWCOMPR	—	USEDSSIOHWCOMPR	Processes only hardware compressed data sets
MAXVOLS	—	USEDSSIOMASVOLS	Processes only those data sets that span more than 20 volumes
PDSE	—	USEDSSIOPDSE	Processes only PDS/E program object libraries
*	—	USEDSSIO*	Processes all supported data sets

For example, to process HFS and PDS/E data sets using the DFSMSdss data mover, simply code:

```
//SYSIN   DD  *
SCAN REALVOLS
   SELECT VOL=/
      BACKUP RETPD=180
//SYSPARMS  DD  *
USEDSSIOHFS,PDSE
/*
```

USERSENQmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the USERSENQ user exit. See the user exit description for “USERSENQ — ENQ/DEQ Exit” on page 256 for more information.

To perform its intended task, the CA-Disk auto-restore function internally assigns and uses value "DMSNOENQ" for this sysparm.

USERVENQmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the USERVENQ user exit. This exit is used to monitor and/or update the allocation type (DISP=SHR or OLD) CA-Disk will use to allocate VSAM data sets. See the user exit description for “USERVENQ — VSAM DISP=SHR/OLD Exit” on page 256 for more information.

USRCDLENxx - This sysparm specifies the LEN of the user code in the DSCB. The default value is 12. This sysparm is used in DASD Space Billing.

USRCODEXmmmmmmmm - This sysparm specifies the module name to be invoked for the USRCODEX user exit. See the user exit description for “USRCODEX — User Code Substitution Exit” on page 257 for more information.

VBRCHCATn - For cataloged data sets, it’s corresponding DSNINDEX record is considered the same version of the data set independent of the volume. If a data set is not-cataloged, only the corresponding DSNINDEX record and volume is considered the same version of the data set. When this sysparm is specified with N, CA-Disk records the catalog status of the data set at VTOC-image backup time. Also, at VTOC based recovery time, all data sets by the same name are considered the same data set.

When this sysparm is specified with the value Y, CA-Disk records the catalog status of the data set at VTOC-image backup time. When similarly set at VTOC based recovery time, data sets with the same name are considered eligible for restore dependent on their catalog status or volume orientation.

Note: Requesting catalog information at VTOC image backup time incurs significant overhead.

If you have an environment in which uncataloged data sets exists, specifying Y is a good option. While it sacrifices performance, it ensure accuracy.

VBRIVSAMn - By default, the CA-Disk VTOC parameter on the SELECT command only processes non-VSAM data sets. This sysparm activates support to include VSAM clusters as well. The default value of N is not recover VSAM. With this sysparm specified with a value of Y, VSAM is eligible during VTOC based recovery.

VTOC base recovery has the following limitations:

- Data and Index components for AIX clusters does not invoke a restore
- Clusters that do invoke a restore must have their associated catalog available, or the restore does not take place
- Recovery of sub-allocated VSAM is not supported

VCBYPASSn - The default value of this sysparm is N, which suppresses printing of messages stating the reason why the Move/Copy function bypassed a data set for migration. To cause such messages to be printed, specify a value of Y.

Note: The COPY command will always print a message stating the reason why it bypassed a data set.

VCCHGBITn - The default value of this sysparm is N, which causes CA-Disk, when it moves a data set, to leave the change bit the same as it was before the data set was moved. To cause CA-Disk to set the change bit on each moved data set, specify a value of Y. Setting the change bit on each moved data set can increase the number of data sets backed up in implicit archive runs, but can simplify the recovery of packs. See also sysparm "RSCHGBITn" on page 172.

VCCOPYEXmmmmmmmm - This sysparm specifies the module that is given control by the Move/Copy function just prior to copying an explicitly named data set (using the COPY command). All selection screening by CA-Disk has been completed and the user's exit module is given a chance to override the copying. A return code of 2 from the exit causes CA-Disk to bypass the data set. See the user exit description for "VCCOPYEX — Move/Copy Screening" on page 257 for more information.

VCDMIGEXmmmmmmmm - This sysparm specifies the module that is given control by the Move/Copy function just prior to the migration of a data set (using the MIGRATE command). All selection screening by CA-Disk has been completed and the user's exit module is given a chance to override the migration. A return code of 2 from the exit causes CA-Disk to bypass the data set. See the user exit description for "VCDMIGEX — Move/Copy Screening" on page 258 for more information.

VCUDSORGn - The default value of N for this sysparm causes the Move/Copy function to bypass data sets with unknown organization types. If you want the Volume Configurator to treat such data sets as sequential data sets, specify a value of Y.

VOLSELREn - Specifying a value of Y for this sysparm will require the user to code a VOLUME= parameter on all SELECT statements. The default value of N is equivalent to coding VOLUME=/ on a SELECT statement which

causes CA-Disk to attempt to process ALL data sets selected by the SELECT command. This can allow unwanted processing to occur. For example, if a user unintentionally runs:

```
SCAN REALVOLS
   SELECT CRITERIA=(DSORG,EQ,VSAM)
      DELETE
```

Note: For other related sysparms, refer to the descriptions of "SELECTREy" on page 175 and "DSNSELREn" on page 141.

VSACCESSc - The default value for this sysparm is "C", which instructs CA-Disk to read VSAM clusters with control interval access. Specify this sysparm with a value of "L" to force CA-Disk to read VSAM clusters using logical access methods. When specifying this sysparm, also look at the value specified for sysparm VSARCFMT.

VSAM clusters that do not observe logical record definitions, such as DB2 data sets, Oracle data sets and some data sets created by MSA, can only be read with control interval access. CA-Disk automatically detects these data sets, and reads them using control interval access, regardless of the value specified in this sysparm.

VSAM clusters that have been compressed with SOFTWORK's VDC product can only be read with logical access methods. CA-Disk automatically detects these data sets, and reads them using logical access methods, regardless of the value specified in this sysparm.

RACF users should note that using control interval access requires having CONTROL authority for the VSAM clusters. This is higher than UPDATE authority, which is higher than READ authority required by logical access methods. If it is unacceptable to require CONTROL access to read VSAM clusters, specify this sysparm with a value of "L".

Note: If the CIARPT parameter on the REPORT command is present in the same SYSIN as the BACKUP or ARCHIVE command, control access is required and VSACCESSC will be used.

VSACESEXmmmmmmmm - This sysparm specifies the name of the user exit that is given control prior to any read of a VSAM cluster. The purpose of the user exit is to interrogate (and change if desired) both the access type that CA-Disk will use on a given data set, and also the format in which to write the data set when archiving it either to tape or disk. See the user exit description for "VSACESEX — Change VSAM Read/Write Processing Options" on page 259 for more information.

VSAIXDEFy - This sysparm is used during VSAM restore and recover processing to determine if the alternate indexes that were associated with a VSAM cluster are to be defined after the base cluster is restored by CA-Disk. The default value of Y causes the indexes to be defined. Specify a value of N to cause CA-Disk to only restore the base cluster, without redefining the alternate indexes. This sysparm can be overridden on the RESTORE and RECOVER commands by specifying the parameters AIXDEF or NOAIXDEF. Refer to the topic "Parameter for VSAM Clusters" on page 258 in the User's Guide for further information regarding these parameters. See also sysparm "VSBLDINXy" on page 193 for information regarding building the alternate index after it is defined.

Note: Not available when using the DFSMSdss Data Mover interface.

VSALLOEXmmmmmmmm - This sysparm specifies the module that is given control prior to all VSAM defines in CA-Disk. The purpose of the user exit is to interrogate (and change if desired) the primary and secondary space allocation amounts to be used in the define. See the user exit description for "VSALLOEX — Change VSAM Space Allocation Amounts" on page 260 for more information.

VSALOSHRn - This sysparm controls the disposition used for allocating VSAM data sets. The default value is N, which causes CA-Disk to allocate only those data sets defined with share options (1 3) with DISP=SHR, and all others with DISP=OLD. To cause CA-Disk to use DISP=SHR for all VSAM data set allocations, specify a value of Y. Note that specifying a value of Y can cause an integrity exposure since CA-Disk does no ENQ for VSAM resources.

To perform its intended task, the CA-Disk auto-restore function internally assigns and uses a value of "Y" for this sysparm, which causes DISP=SHR to be used for all VSAM data set allocations.

VSAMSUPPn - The default value for this sysparm is N, which indicates to CA-Disk that the VSAM support is not active. Specify a value of Y to activate the VSAM support.

VSAPPENDn - This sysparm is used during restore and recover of VSAM data sets. Its purpose is to generate new data and index component names for those clusters originally defined with VSAM-generated component names based on timestamp information. Data and index components that were given explicit names when originally defined are not affected by this sysparm.

If a value of Y is specified, CA-Disk generates each new name (to replace the timestamp version) using the base cluster's data set name, with ".DATA" appended for the data component, and ".INDEX" appended for the index component.

For example, if a base cluster of LABS.TJP.PMRFILE is being restored, the data component's name will be LABS.TJP.PMRFILE.DATA and the index component's name will be LABS.TJP.PMRFILE.INDEX. If the base cluster's name is too long to add the suffix (.DATA or .INDEX), CA-Disk will let VSAM generate a new name based on timestamp information.

Two other sysparms, VSDATSUF and VSINXSUF, can also be used for this same purpose, with or without specifying VSAPPEND. They control the new name for each component independently, and require you to specify the suffix to be appended. Therefore they are normally used only if suffixes other than .DATA and .INDEX are desired.

If the default value of N is in effect, and neither VSDATSUF nor VSINXSUF is specified, CA-Disk will instruct VSAM to generate another component name based on timestamp information.

See sysparms “VSDATSUFxxxxxxxx” on page 194 and “VSINXSUFxxxxxxxx” on page 195 for related information.

VSARCAIXn - When this sysparm is specified with the default value of N for an archive or backup run, CA-Disk bypasses processing alternate indexes and clusters as separate entries. This sysparm has no effect for an explicit archive or backup of a base cluster.

When specified with a value of Y for an implicit backup run, CA-Disk processes the alternate indexes as separate data sets. And, entries are placed in the files data set for the alternate indexes. This sysparm must be specified with a value of Y for an explicit archive or backup of the alternate index.

Note: This sysparm is ignored for an implicit archive run. This is because all AIX associations and DEFINE information is saved with the base cluster before the delete takes place, not as a separate data set.

VSARCFMTl - This sysparm controls the format of archive data for VSAM, if VSACCESSc has been specified. If VSACCESSl is specified, this sysparm is not considered and the data is archived in the logical record format. When the default value of L is specified, logical records are written to the archive medium. Specify a value of C to force control interval image format for all ESDS clusters, regardless of the naming convention they observe. When specifying this sysparm, also look at the value specified for sysparm VSACCESS.

VSAM clusters that do not observe logical record definitions, such as DB2 data sets, Oracle data sets and some data sets created by MSA products, cannot use logical formats. These VSAM data sets are automatically detected by CA-Disk and formatted in control interval images, regardless of the value specified in this sysparm.

VSAM data sets that have been compressed with SOFTWORK's VDC product can only use logical formats. CA-Disk automatically detects these data sets, and formats them in logical records, regardless of the value specified in this sysparm.

When a control interval image format is used, the control interval size cannot be changed when the data set is restored.

VSATTREXmmmmmmmm - This sysparm specifies the module that is given control prior to all VSAM defines in CA-Disk. The purpose of the user exit is to interrogate (and change if desired) the BUFFERSPACE AND CISZ values to be used in the define. In addition, certain data set attributes can be changed by this exit: RECOVERY vs. SPEED, UNIQUE vs. SUBALLOC, REUSABLE vs. NONREUSABLE, and data set SHARE OPTIONS. See the user exit description for "VSATTREX — VSAM Bufferspace and CI Size Exit" on page 261 for more information.

VSATTRIBn - The default value of this sysparm is N, which causes CA-Disk to use the original CISZ and BUFFERSPACE values found on the archive copy when processing data sets. By specifying a value of Y, all data sets that are defined by CA-Disk in the job step will have CISZ and BUFFERSPACE calculated by AMS. These values can or cannot correspond to the original values with which the cluster was defined.

This sysparm should not be used when processing ESDS or LDS clusters. Therefore, CA-Disk automatically detects these data sets and will not modify the CISZ for them.

Note: This sysparm should not be used if IMS data bases are restored by CA-Disk. If the CISZ is changed inadvertently, the data base becomes inoperable. It must be restored with its original CISZ! Its also not available when using the DFSMSdss Data Mover interface.

VSBIXCATxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx - Specify the name of the catalog into which the workfiles are to be defined to build alternate indexes when restoring VSAM clusters. If this sysparm is not specified, the catalog will be determined by the following order: (1) any user catalog specified in a STEPCAT or JOBCAT dd statement, or (2) if the data set names on the IDCUT1 and IDCUT2 dd statements are identified with qualified entry names and the first qualifier is the same as the name of a user catalog or the alias of a user catalog, the user catalog so identified is used, or (3) the master catalog is used.

VSBIXPSWnnnnnnnn - If the catalog specified in the VSBIXCAT sysparm is password-protected, specify this sysparm with the catalog's update (or higher-level) password. If this sysparm is not specified with the password, the operator can be prompted to supply the correct password.

VSBLDINXy - This sysparm is used during VSAM restore and recover processing to determine if alternate indexes are to be built automatically after they are defined by CA-Disk. The default value of Y causes CA-Disk to build the indexes. Specify a value of N to cause CA-Disk to only define the alternate index without actually rebuilding it. This sysparm can be overridden on the RESTORE and RECOVER commands by specifying the parameters BLDINX or NOBLDINX. Refer to the topic "Parameter for VSAM Clusters" on page 258 in the User's Guide for further information regarding these parameters.

VSCATMAXnnnn - Specify a 4-digit number at least one greater than the number of user catalogs defined in the master catalog. This value is used to build a dynamic table of user catalog names in the implicit VSAM support. Note that this table is only built when sysparm VSCATPTN is specified as Y (or C) and a catalog data set name pattern is specified (as opposed to an explicit catalog name) on the CATALOG= parameter of the SCAN command. The default value for this sysparm is 0050.

VSCATPTNy - The default value for this sysparm is Y, which allows the user to enter a catalog data set name pattern (along with explicit names) on the CATALOG= parameter of the SCAN command in the implicit VSAM support. Specify a value of N to disallow this feature.

If a value of Y (or C) is specified and the user specifies a catalog data set name pattern, the master catalog is searched for all user catalogs defined in it. These user catalogs, along with the master catalog, are compared to the specified pattern for a match. All catalog names that match will be processed as if they had been explicitly named on the CATALOG parameter.

Note: This feature does incur the additional overhead required to search the master catalog.

When a value of Y is specified and a catalog pattern is entered on the SCAN CAT= command, CA-Disk will verify that at least one catalog matches the pattern specified. If none match, processing is terminated and none of the commands in the job stream are executed. If you want to fail only this SCAN command and allow any others to complete, specify a value of C for this sysparm. This causes CA-Disk to continue processing other commands even though an invalid catalog pattern name was specified.

VSCIACATy - The default value of this sysparm is Y, which causes catalog information to be printed on the cluster internal analysis report (CIA) of the VSAM support. Specifying a value of N causes the information to be suppressed; it can then only be printed by explicit specification on the CIAOPT= parameter.

VSCIADETn - The default value of this sysparm is N, which causes the graphic analysis section of the cluster internal analysis report (CIA) of the VSAM support to be suppressed. Specify a value of Y to have the graphic analysis section of the report printed.

VSCOMPNMn - The default value of this sysparm is N, which suppresses printing of the data set name associated with the data and index components on the Cluster Data Report of VSAM. Specify a value of Y to cause CA-Disk to print the data set name in this report. If this parameter is specified in the job stream, it must be specified in the PRINT step of the job.

VSCRIMAXnnnn - Specify as "nnnn" the maximum number of subparameters to be allowed on the SELECT and EXCLUDE commands for the implicit VSAM support. The default value is 0150. Normally the default value should not require any change. The maximum value that can be specified is 0255.

VSDATEFTn - The default value of this sysparm is N, which causes CA-Disk to use the IBM formats and names supporting 4-digit year numbers for VSAM Restore and Recover. Specify a value of Y if it is determined that your system does not support the default date formats.

VSDATSUFxxxxxxxx - This sysparm specifies a 1- to 8-character suffix to be used by CA-Disk at Restore time or in the Move/Copy function at COPY newname time to generate new data component names for those clusters that have VSAM-generated component names based on the timestamp. The sysparm defaults to a value of blanks, such that the newname function is inactive. To activate the newname function, specify a non-blank value for this sysparm or specify sysparm VSAPPEND with a value of Y. If both sysparms are specified, VSDATSUF will override VSAPPEND.

For example, if this sysparm is specified with a value of D, and cluster LABS.TJP.XREFFILE is being restored, the data component will be named LABS.TJP.XREFFILE.D.

Note: Specifying sysparm VSAPPEND with a value of Y is the same as specifying this sysparm with a value of DATA and sysparm VSINXSUF with a value of INDEX.

VSDEFCATy - The default value of this sysparm is Y, which allows catalog management to determine the catalog in which to put the cluster, based on IBM-documented catalog selection criteria. Specify a value of N to cause CA-Disk to not let catalog management determine the catalog. Instead, CA-Disk will define the cluster in the same catalog from which it was archived. If this catalog is not the same one that catalog management would have used, a STEPCAT dd statement for the catalog will probably be needed also.

Note: This sysparm has no effect when the catalog is explicitly named on the RESTORE or RECOVER command. Its also not available when using the DFSMSdss Data Mover interface.

VSDSPACEn - This sysparm lets you specify if sub-allocated VSAM data sets in VSAM data spaces should be bypassed for processing using the DSCL SCAN REALVOLS command. This sysparm is only relevant for the DSCL SCAN REALVOLS command, and not the DSCL SCAN CATALOG command. Processing the sub-allocated VSAM data sets incurs extra overhead for SCAN REALVOLS, due to additional catalog processing.

When this sysparm is specified with the default value of N, CA-Disk processing bypasses sub-allocated VSAM data sets when using the DSCL SCAN REALVOLS command.

When this sysparm is specified with a value of Y, CA-Disk processing allows selection of sub-allocated VSAM data sets when using the DSCL SCAN REALVOLS command.

VSIDCUTPxxxxxxxx - Because of improvements to the AMS-BLDINDEX function, this sysparm is not required and is ignored if your system has DFP 3.0 and above or DFSMS installed.

If desired, specify a one- to eight-character prefix to be used by CA-Disk if it needs to generate data set names for the IDCUT1 and IDCUT2 dd statements during restore or recover operations. If a value is not specified, the high-level qualifier of the first alternate index being built will be used. For more details, see "Activating VSAM Support" on page 43 in the Installation Guide. Note, this sysparm is ignored if your system has DFP 3.0 and above or DFSMS installed.

VSIDCUTVxxxxxx - Because of improvements to the AMS-BLDINDEX function, this sysparm is not required and is ignored if your system has DFP 3.0 and above or DFSMS installed.

If you do not specify the IDCUT1 and IDCUT2 dd statements directly in your restore or recover JCL, you must specify this sysparm with the volume(s) that CA-Disk can use for the sort work data sets required by BLDINDEX processing. For more details, see "Activating VSAM Support" on page 43 in the Installation Guide. Specify from 1 to 10 volumes in the list, all of which must be the same device type. This sysparm is ignored if your system has DFP 3.0 or later or DFSMS installed.

VSINXSUFxxxxxxxx - This sysparm specifies a 1- to 8-character suffix to be used by CA-Disk at Restore and in the Move/Copy function at COPY newname time to generate new index component names for those clusters that have VSAM-generated component names based on the timestamp. The sysparm defaults to a value of blanks, such that the newname function is inactive. To activate the newname function, specify this sysparm with a non-blank value, or specify sysparm VSAPPEND with a value of Y. If both sysparms are specified, the value specified for this sysparm will override VSAPPEND.

For example, if this sysparm is specified with a value of INX, and cluster LABS.TJP.XREFFILE is being restored, the index component will be named LABS.TJP.XREFFILE.INX.

Note: Specifying sysparm VSAPPEND with a value of Y is the same as specifying this sysparm with a value of INDEX and sysparm VSDATSUF with a value of DATA.

VSMAFTEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the VSMAFTEX user exit. See the user exit description for "VSMAFTEX — Implicit VSAM Data Set Screening (Post-Validate)" on page 262 for more information.

VSMPRIEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the VSMPRIEX user exit. See the user exit description for "VSMPRIEX — Implicit Data Set Screening (Pre-Validate)" on page 263 for more information.

VSMPURGEy - The default value of this sysparm is Y, which causes CA-Disk to delete unexpired data sets. Specify a value of N to instruct CA-Disk to not allow deletion of unexpired data sets. The default value of Y is equivalent to the use of the PURGE parameter when issuing a DELETE via Access Method Services.

VSMSGEMPn - The default value of this sysparm is N, which causes CA-Disk to bypass issuing message 2517 for empty VSAM clusters that are being archived. Specify a value of Y to have the message be issued whenever the condition is encountered. For more information, refer to the message description for "2517" on page 133 in the Message Manual.

VSMSGTOTn - The default value of this sysparm is N, which causes CA-Disk to bypass issuing message 2562 when a difference in record totals is detected between CA-Disk processing and the catalog. Specify a value of Y to have the message issued whenever the condition is encountered. For more information, refer to the message description for "2562" on page 144 in the Message Manual. Note that message 2832 might not be issued, depending on the value specified for sysparm VSMSGWRN.

VSMSGWRNn - The default value of this sysparm is N, which causes CA-Disk to bypass issuing message 2832 when the CA-Disk record total is less than the value found in the catalog for the data set. Specify a value of Y to have the message issued whenever the condition is encountered. For more information, refer to the message description for "2832" on page 169 in the Message Manual.

VSMSPWEXmmmmmmmm - This sysparm specifies the module that is invoked during initialization processing in the VSAM support. The purpose of the module is to allow the user to change the default CA-Disk master password (currently implemented only in the VSAM support). See the user exit description for "VSMSPWEX — VSAM Master Password Exit" on page 264 for more information.

VSMSTATSn - The default value of this sysparm is N, which causes CA-Disk to bypass printing of additional data and index component statistics on the Cluster Detail Analysis report. Specify a value of Y to force printing of these additional statistics. The statistics for each component will be printed on an additional print line. Included with these statistics are the component's high used and high allocated RBAs, along with tracks allocated.

Note: If this parameter is overridden in the user's job stream, the override must be specified in the PRINT step.

VSNEWNEXmmmmmmmm - This sysparm specifies the module that is given control prior to all VSAM defines in CA-Disk. The purpose of the user exit is to interrogate (and change if desired) the data set name with which the component will be defined. The exit can either change a name that CA-Disk will use, or it can instruct CA-Disk to let AMS generate a name for the component. See the user exit description for "VSNEWNEX — VSAM New Name Exit" on page 264 for more information.

VSNOERASn - The default value of this sysparm is N, which causes CA-Disk to honor the erase option when deleting a VSAM cluster. Specify a value of Y to cause CA-Disk to override the erase option when deleting a VSAM cluster. If CA-Disk attempts to delete a cluster that was defined with the erase option and an error prevents its deletion, this sysparm will cause CA-Disk to reattempt the delete with a no erase option. This causes the cluster to be deleted from the catalog without overwriting the data component with binary zeroes. Note that even if this

sysparm is specified as a Y, the default erase/no erase value will be attempted first. The delete with no erase will only be attempted if the first delete fails.

VSNOSCROn - The default value for this sysparm is N, which causes CA-Disk not to honor the SCRATCH option if it has been specified and the delete fails. This would most likely occur when a data sets catalog entry is pointing to an unavailable volume.

Specify a value of Y to cause CA-Disk to retry a delete that failed. The second attempt to delete will be performed with the NO SCRATCH option specified. This causes the entry to be deleted from the catalog (BCS or basic catalog structure) without trying to access the volume. CA-Disk will only retry a SCRATCH for the following IDC3009I return codes:

TABLE 43. Common IDC3009I Return Codes

Return Code	Reason Code	Meaning
8	214	Unit(s) not available
8	218	Specified volume is not mounted
50	18	I/O error occurred while processing a VVDS
102	5	I/O error occurred while processing a VVDS

Typically this sysparm would be set to Y when restoring a VSAM data set that has a cataloged entry pointing to a corrupted VVDS, and you want to restore to a different volume. Issuing a standard delete (either by CA-Disk or IDCAMS) will fail because the BCS still contains an entry for the data set. When this occurs, the NOSCRATCH option is required to clean up the catalog entry without requiring access to an unavailable volume.

We recommend that you use the default value of this sysparm (N) for normal CA-Disk processing, and only specify a value of Y during a volume recovery situation. Note that even if this sysparm is specified as Y, a standard delete request will be attempted first; the delete with NOSCRATCH will be attempted only if the first delete fails with a specific set of reason codes.

VSONLINEn - This sysparm controls the amount of checking done to determine if the volumes containing VSAM clusters are online to the system. It applies only to the implicit mode of processing VSAM data sets.

Since removable volumes and offline volumes (or even obsolete/garbage entries pointing to currently invalid volumes) are generally exceptions rather than the rule, the default value of this sysparm is N. CA-Disk will avoid the overhead of checking, and assume that needed volumes (the volumes associated with each cluster in the catalog) are available. Mount requests can occur if some are not available. This can be desirable if you know a few volumes are offline, but you want them mounted for backup processing, for example.

Specify a value of Y if you want CA-Disk to verify that needed volumes are online. This will prevent unexpected mount requests for volumes that are not available. Verification is done by dynamically allocating the cluster with the "NOMOUNT" attribute. If the allocation of the cluster fails, processing for cluster will be bypassed.

The Y option will eliminate unexpected mount requests for VSAM data sets residing on offline volumes, but unusual catalog conditions and entries can cause other unexpected mounts before CA-Disk gains control. These are usually exception cases. Examine the catalog entry, the alias pointers in the master catalog, and other relevant information to determine and correct the condition causing the unexpected mount request.

Note: The verification done as a result of this sysparm being set to a value of Y can increase the run time for your job. It can also increase the number of messages written to your job log. Adjusting the value of the MSGLEVEL parameter on your JOB card can reduce the number of messages.

VSPREDEFn - The default value of this sysparm is N, which will cause CA-Disk to not allow restore or recover commands to overwrite preallocated VSAM clusters. Specify a value of Y to allow a restore to overwrite preallocated VSAM clusters with the REUSE attribute. Note that CA-Disk cannot be held responsible for the integrity of the restored data when a value of Y is specified.

You should not specify a value of Y during VSAM volume recover after an ICF catalog recovery when the information in the VVDS is not correct.

See sysparm "PREALLOCn" on page 158 for restore of non-VSAM data sets.

(To protect integrity, the CA-Disk auto-restore function internally assigns and uses a value of "N" for this sysparm, which prevents VSAM data sets being auto-restored into existing clusters.)

VSRESCLDn - The default value for this sysparm is N, which causes CA-Disk to suppress dynamic printing the VSAM cluster detail report. Specify a value of Y to cause CA-Disk to print a cluster detail report dynamically whenever a VSAM cluster is recovered or restored. This report will be printed on the restore report that is normally produced at restore time. The lines that are printed are based on the values specified for sysparm RPTCLSEL.

VSSEQCNTnnn - This sysparm allows you to specify a numeric quantity from 1 to 999 to indicate the number of "key out of sequence" errors CA-Disk should tolerate (per data set) at restore time before terminating the restore for that data set. If the default value of 0 is used, the first such error will cause the restore to be terminated for that data set. If you specify a value greater than 0, and the number of errors is less than the value specified, the data set will be restored but the records that were out of sequence must be loaded manually back into the cluster. A dump of every record that is out of sequence will be printed to the MSGPRINT error report for user inspection.

This sysparm will normally be required only if you have used sysparm VSSEQERR to allow a VSAM cluster with such problems to be archived.

VSSEQERRy - This sysparm has a default value of Y, which causes CA-Disk to stop processing a VSAM cluster if an out-of-sequence key is encountered. Specify a value of N to cause CA-Disk to ignore out-of-sequence keys in a key-sequence data set and to continue processing the data set. Message 2839 will be issued regardless of the value specified for this sysparm. If a value of N is specified, no records will be lost on the archive medium but the data set can have to be restored by non-conventional means (such as extract, sort, and then using the REPRO facility of IDCAMS) since the keys are not in sequence.

VSSPEEDOy - The default value of this sysparm is Y, which causes CA-Disk to use the "speed" option, regardless of the value with which the cluster was defined. Specify a value of N to cause CA-Disk to use the "speed" or "recovery" option with which the cluster was defined.

Note: Not available when using the DFSMSdss Data Mover interface.

VSSPUSEDy - The default value of this sysparm is Y, which causes CA-Disk to consolidate secondary extents in the original cluster and increase the primary allocation (when required) accordingly. This results in defragmentation of the data set. However, if the pack is badly fragmented already, the larger amount of space cannot be available for the primary request. In this case, specify a value of N to cause CA-Disk to use the original allocation amounts for all DEFINE requests.

Note: Not available when using the DFSMSdss Data Mover interface.

VSSTUMPLnnnnnn - This sysparm allows you to specify the number of bytes required to hold the largest catalog entry returned from a catalog locate. The default value is 026000. Specify this sysparm only in response to message 2561.

VSSYSVERn - This sysparm is applicable only to those users who have implemented the Data Facility/Extended Function program product from IBM. Normal OPEN processing of a VSAM data set in this environment will cause a VERIFY to be issued against any cluster with an out-of-sync end-of- file marker. This is internal to OPEN processing and causes the VERIFY to be issued prior to the application program receiving control back from OPEN. This sysparm's default of N turns off this automatic VERIFY feature. Specify a value of Y to allow the automatic VERIFY to take place.

Note: Not available when using the DFSMSdss Data Mover interface.

VSUNOPENn - The default value of this sysparm is N, which causes CA-Disk to ignore a defined but never opened VSAM cluster when the criteria (MODIFIED,EQ,YES) is specified. Specifying a value of Y forces CA-Disk to select such clusters once, but never again until it is actually modified.

There is one exception, and that occurs on the day the cluster is defined. On that day, CA-Disk will select "defined, but never opened" clusters every time the criteria (MODIFIED,EQ,YES) is executed.

The standard way to detect if a VSAM cluster has been modified is reviewing the TIMESTAMP information and/or the DS1DSCHA (Change bit) in the format-1 DSCB. AMS sets neither of these for a never-opened cluster. So, when the TIMESTAMP is zero, CA-Disk compares the CREDT to the ARCDT. If CREDT is greater than or equal to the ARCDT, CA-Disk considers the cluster modified.

Specifying VSUNOPEN with a value of Y ensures that a "defined, but never opened" cluster is backed up when the criteria MODIFIED,EQ,YES is executed.

VSUSEBLKn - The default value of this sysparm is N, which causes CA-Disk to use the control interval size (CISZ) of the data component for selection criteria BLKSZ and to be printed under the BLKSZ header in the VSAM MVD and SVD reports. Specify a value of Y to use the cluster's physical block size instead of the CISZ.

VSUSEMODn - The default value for this sysparm is N, which causes CA-Disk to print the last use date from the format-1 DSCB for the data component of a VSAM cluster under the USEDT heading in the VSAM MVD and SVD reports (ICF only). For a standard VSAM catalog data set using the default value, blanks are printed in this column. Specify a value of Y to cause a VSAM cluster's last modified date to be printed under the USEDT heading.

VSVERIFYn - The default value for this sysparm is N, which causes CA-Disk to bypass processing of a cluster when the need to verify it is detected at OPEN time. The cluster must be verified manually. Specify a value of Y to instruct CA-Disk to issue a VERIFY when the need is detected at OPEN time. If the VERIFY succeeds, CA-Disk will then attempt to OPEN the data set again and continue processing.

Specify a value of A to instruct CA-Disk to always issue a VERIFY when archiving or backing up a cluster. A VERIFY prior to a BACKUP or ARCHIVE identifies some broken VSAM KSDS data sets that when processed could result in a loop, but a small increase in CPU time should be expected as a result of these VERIFY operations.

VTOCREADe - This sysparm controls which of three VTOC readers CA-Disk will use. Although these VTOC readers are functionally equivalent, they perform quite different. The default value is E.

TABLE 44. VTOC Reader Names and Descriptions

Reader	Description
E	uses EXCP processing and is quite fast.
B	uses the BSAM reader. This is a standard IBM facility, but is quite slow. This can increase CA-Disk processing time up to 200 percent on some report writers.
O	uses the OBTAIN macro VTOC reader. This also is a standard IBM facility and is even slower than option B. This can increase CA-Disk processing time up to 1100 percent on some report writers.

Use options B or O only when VTOC problems prohibit the use of the CA-Disk EXCP VTOC reader.

Note: Refer to the appropriate IBM manuals for detailed information on the sensitivities of these VTOC readers.

WTORESRLy - The default value of this sysparm is Y, which causes CA-Disk to issue a WTO message for each RESERVE issued on a pack during PDS Compress in a shared-DASD environment. Specify a value of N to cause CA-Disk to not issue a WTO message under these circumstances.

XCPARCEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the XCPARCEX user exit. See the user exit description for "XCPARCEX — XCOPY Archive Volume Selection Exit" on page 265 for more information.

XCPDSNEXmmmmmmmm - Specify this sysparm to provide the module name to be invoked for the XCPDSNEX user exit. See the user exit description for "XCPDSNEX — XCOPY Data Set Level Selection Exit" on page 265 for more information.

CHAPTER 5

User Exits

General Information

Modules can be incorporated in CA-Disk to satisfy unique processing requirements of an installation. These modules are identified to CA-Disk by appropriate system parameter entries in member SYSPARMS of the parmlib data set. During execution of CA-Disk, the exit modules are dynamically loaded when they are called for the first time.

User exit modules can directly invoke CA-Disk functions by use of the MI dynamic interface, as illustrated in the topic "Module Call Interface" on page 302. User exits using the MI dynamic interface must be link-edited with the MI interface module from the CA-Disk library and should be placed in an APF authorized load library. Instructions for doing this are under the "Assembly and Link-Edit of CA-Disk Modules" on page 299.

If your exit is written in COBOL or PL1 rather than assembler, and calls other routines, it is recommended that you use the MI2 interface rather than MI (fast path). See "MI Fast Path" on page 275 for more information about MI2.

Note: User exits can need to be re-assembled each time you upgrade your CA-Disk release level (i.e., source code to ADSMI002 has changed).

In order to provide compatibility for existing CA-Disk modules with 24-bit addressing, only the following AMODE/RMODE combinations are valid for user exits:

```
AMODE 24, RMODE 24
AMODE 31, RMODE 24
AMODE ANY, RMODE 24
```

Each of the user exits is discussed under the sysparm entry name that is necessary to define the respective module to CA-Disk for invocation.

Assembler language conventions are followed in passing parameters to the exit module. Register 1 will point to a list of addresses, each of which points to a specific parameter as defined for the exit. In most cases, the exit module must return a halfword code via the designated parameter to indicate to CA-Disk what further actions are to be taken.

If you use the Common File Handler Interface (described in "Common File Handler Interface" on page 33), and you are not already executing a CA-Disk function, and files data set logging is activated, the function you are running needs to be stumped within your user exit prior to any file updates. This is necessary for files data set logging as the current function is part of the information captured during logging.

To implement this, add the following to your user exit (this call need only be executed once):

```
* Stump the function to retrieve during files data set logging
  #CALL ADSUI302,($ISRT,FUNCTION,CTYPE,K08,USRFUNC,UI302RC),VL
*
$ISRT    DC CL5'$ISRT'    Service = Insert
FUNCTION DC CL8'FUNCTION' Field name for retrieval
CTYPE    DC CL2'C'        Service type = character data
K08      DC H'8'          Length of field
USRFUNC  DC CL8' '        Fill in user function name
UI302RC  DC H'0'          Return code
```

Using Previous Releases of User Exits

Note: If you are upgrading from an 8.2 release of CA-Disk, you can ignore this section.

User exits built for previous releases of CA-Disk are compatible with release 9.0 with one exception:

1. If the CA-Disk provided PROC macro is used to begin the exit program, verify whether or not the PROC macro is specified with an OPTIONS=(blank) parameter. For example:

   ```
   csectname PROC OPTIONS=
   ```

   If this OPTIONS= parameter (without a value) is found in any user exit, that exit must be reassembled.

   Note: Modifications to the user exit are not necessary, reassembly alone will use an updated version of the PROC macro provided with release 9.0, thus creating a release 9.0 compatible module.

Use of PROC statements with other values are all compatible with release 9.0 and can be moved directly into the 9.0 libraries. All user exits using standard assembly entry and exit code are also fully compatible with release 9.0. For example, typical uses of the PROC macro that generate code compatible with release 9.0 are:

```
csectname PROC OPTIONS=STATIC
csectname PROC OPTIONS=NOSAVE
csectname PROC OPTIONS=MAIN
csectname PROC
```

Finally, if you have not correctly implemented the PROC macro, attempting to invoke a user exit from CA-Disk will result in User Abend 1547-16 followed by:

MTF005I OBSOLETE MIVT SERVICE REQUESTED

User Exit Cross Reference Matrix

TABLE 45. User Exits

User Exit Name	pg #	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ. MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF	XCOPY
ALLOEXIT	206				•	•		•															
ALOCATEX	207				•	•		•											•				
ALOPOSEX	207				•	•													•				
ARCDSNEX	208		•	•																			
ARCEODEX	208		•	•			•																
ARCHIVEX	209		•	•														•					
ARESPREX	210				•																		
ARPOOLEX	210				•																		
CCINITEX	211		•	•																			
DARCHIEX	212		•	•																	•		
DBRCPREX	213		•	•	•	•		•									•	•	•				
DCCMPEX0	213		•	•			•																
DCCMPEXn	213		•	•																			
DCDCPEX0	214		•	•			•	•															
DCDCPEXn	214		•	•				•															
DCDSNDEX	214		•	•																			
DEFARCEX	214																						
DERASEEX	215																				•		
DINXUFEX	215		•	•	•	•	•	•															
DRESTOEX	216				•																•		
DSBUSREX	217																				•	•	
DSCBCVEX	217				•	•																	
DSCLSCEX	218		•	•			•																
DSDISPEX	219		•	•																			
DSNDELEX	220									•													
DSNDLPEX	220									•	•												
FMSSAFEX	221					•																	
FMSSPREX	221					•																	
GDGCATEX	222				•	•																	
ICHEKAEX	222		•	•	•	•	•	•											•				

TABLE 45. User Exits

User Exit Name	pg #	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ. MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF	XCOPY
ICHEKPEX	223		•	•	•	•	•	•											•				
IXCTLGEX	223				•										•								
IXMREVEX	224									•													
LISTDMEX	225								•														
LISTDVEX	225								•														
LISTREEX	226																				•		
MERDSNEX	226										•												
MERVOLEX	226										•												
MIGDISEX	227															•							
MIGGDGEX	228															•							
MIGREXIT	228															•							
RAADDAEX	228																		•				
RAADDPEX	229																		•				
RACFDAEX	230		•	•	•	•	•	•	•	•	•												
RACFDPEX	230																				•		
RACFEAEX	231		•	•	•	•	•	•	•	•	•												
RACFEPEX	231		•	•	•	•	•	•	•	•	•												
RACFSMEX	232																•						
RACHKAEX	232		•	•	•	•	•	•	•	•	•								•				
RACHKPEX	233		•	•	•	•	•	•	•	•	•								•				
RADEFAEX	233		•	•	•	•	•	•											•				
RADEFPEX	234		•	•	•	•	•	•											•				
RADELAEX	235									•	•												
RADELPEX	235									•	•												
RECPRIEX	236					•																	
RESAFTEX	236				•	•		•															
RESCRNEX	237				•	•														•			
RESPRIEX	237				•	•		•															
RETAFTEX	238		•	•																			
RETGDGEX	239		•	•																			
RETPRIEX	239		•	•																			
RLSEAFEX	240														•								

TABLE 45. User Exits

User Exit Name	pg #	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ. MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF	XCOPY
RPTDSBEX	240	•																				•	
RPTDSNEX	241	•																					
RPTDSUEX	241	•																					
RPTMVDEX	241	•																					
SCRTCHEX	242		•	•	•	•		•											•				
SECURAEX	242		•	•	•	•	•	•	•	•	•	•	•						•	•		•	
SECURPEX	243		•	•	•	•	•	•	•	•	•	•	•						•	•		•	
SPACECEX	245				•	•													•				
SPFHDGEX	246	•																				•	
SPFORDEX	247																					•	
SPFORSEX	247																					•	
SYSOUTEX	247		•	•	•	•		•	•	•	•	•	•						•	•		•	
TAOPPOEX	248		•	•			•				•												
TAOPPREX	248		•	•			•				•												
TAPCTLEX	249									•	•	•	•										
TAPEDLEX	249									•	•												
TMSCTLEX	249									•	•	•	•										
TPMGMTEX	253									•	•	•	•										
TSCHKAEX	254		•	•	•	•	•	•	•	•	•								•				
TSCHKPEX	255		•	•	•	•	•	•	•	•	•								•				
UNLOADEX	255						•																
USERSENQ	256		•	•	•	•	•	•											•				
USERVENQ	256																						
USRCODEX	257																				•	•	
VCCOPYEX	257																	•	•				
VCDMIGEX	258																	•	•				
VSACESEX	259		•	•																			
VSALLOEX	260				•	•		•															
VSATTREX	261				•	•		•															
VSMAFTEX	262																	•					
VSMPRIEX	263																	•					
VSMSPWEX	264		•	•	•	•		•															

TABLE 45. User Exits

User Exit Name	pg #	REPORTS	ARCHIVE/BACKUP	BACKUPCC	RESTORE	RECOVER	VBACKUP	VRECOVER	LISTD/V	IXMAINT	MERGE	REBUILD	IXUPDATE	FILES MGT	RELEASE	SEQ. MIGRATE	PDS COMPRESS	MOVE/COPY	SECURITY	TSO	BILLING	ISPF	XCOPY
VSNEWNEX	264				•	•																	
XCPARCEX	265																						•
XCPDSNEX	265																						•

User Exit Descriptions

ALLOEXIT — Restore/Recover Data Set Allocation

The module named by the sysparm entry is invoked at allocation time during restore or recover processing for non-VSAM data sets. The format-1 DSCB for the data set being restored is passed to the exit to allow the following fields to be updated prior to its allocation:

- CREATE DATE
- EXPIRE DATE
- LAST USED DATE
- LAST MOD DATE
- OPEN COUNT

Use sysparm ALLOEXIT to specify the module name to be invoked for this exit.

TABLE 46. Parameters Passed to ALLOEXIT

Type	Size	Description
INPUT	XL140	Data Set Identifier - Format-1 DSCB for data set to be processed

Note: Not available when using the DFSMSdss Data Mover interface.

ALOCATEX — Data Set Allocation Preprocessing

The module name supplied for this exit will receive control from any CA-Disk routine that allocates non-VSAM data sets on DASD. It receives control prior to allocation (SVC 32 or SVC 99).

The user can request that CA-Disk reject the allocation request or continue processing. The format-1 DSCB to be allocated is also provided to the exit for examination and changes.

Use sysparm ALOCATEX to specify the module name to be invoked for this exit.

TABLE 47. Parameters Passed to ALOCATEX

Type	Size	Description		
IN/OUT	XL256	DATA SET IDENTIFIER - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.		
IN/OUT	XL140	VOLUME IDENTIFIER - Format-4 DSCB for data set. Positions 1-6 contain VOLUME LIST & Count - First 4 bytes is the address of the Volume List, last 2 bytes is a halfword containing the number of volumes in the list.		
IN/OUT	XL4	ALLOCATION AMOUNTS - This parameter contains 2 values indicating how much space to allocate. HW - Number of tracks to be allocated HW - Number of directory blocks (if PDS).		
INPUT	XL2	DEVICE TYPE - Device type as documented for UCBTYP field in UCB control block		
OUTPUT	CL100	If user to reject request by setting return code to 2, optionally a message can be placed here. It will be printed by CA-Disk. If this field is spaces, no message will be printed.		
OUTPUT	HW	Specify return code:	H'1'	Continue Normal Processing
			H'2'	Reject Allocation Request

Note: When using the DFSMSdss Data Mover interface, CA-Disk initially allocates the data set allowing your exit to be called during BACKUP. However, since DFSMSdss allocates the data set for RESTORE, your user exit will not be invoked during RESTORE processing.

ALOPOSEX — Data Set Allocation Post-Processing

The module name supplied for this exit will receive control from any CA-Disk routine that allocates non-VSAM data sets on DASD. It receives control after SVC99 allocation.

This exit allows the user to retain the format-1 DSCB data required by their applications in the 13-byte System Code field and the 2-byte Reserved field at offset +103. These are the only two fields where data can be altered and then moved to the actual DSCB. Specify the correct return code to activate update processing.

Use sysparm ALOPOSEX to specify the module name to be invoked for this exit.

TABLE 48. Parameters Passed to ALOPOSEX

Type	Size	Description
IN/OUT	XL256	Source Data Set Identifier - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.
INPUT	XL140	Source Volume Identifier - Format-4 DSCB for data set. VOLUME SERIAL in positions 1 - 6.
IN/OUT	XL256	TARGET data set IDENTIFIER - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.
INPUT	XL140	TARGET VOLUME IDENTIFIER - Format-4 DSCB for data set. VOLUME SERIAL in positions 1 - 6.
INPUT	XL2	DEVICE TYPE - Device type as documented for UCBTYP field in UCB control block

TABLE 48. Parameters Passed to ALOPOSEX

Type	Size	Description	
OUTPUT	HW	RETURN CODE - Specify return code as indicated to have both the System Code field and the Reserved field of the format-1 DSCB updated with user data.	
		H'1'	Perform Update Processing
		OTHER	Continue Normal Processing

Note: When using the DFSMSdss Data Mover interface, CA-Disk initially allocates the data set allowing your exit to be called during BACKUP. However, since DFSMSdss allocates the data set for RESTORE, your user exit will not be invoked during RESTORE processing.

ARCDSNEX — Archive Data Set Name Generation Exit

The module name supplied by this exit is invoked at archive data set name generation time. This exit allows you to alter the CA-Disk archive data set name to meet special user requirements.

When this user exit is invoked, a unique data set name is created and passed as input. You must make sure that the data set name created by the exit is unique (even across shared operating systems), and that the name selects valid archive pools.

Use sysparm ARCDSNEX to specify the module name to be invoked for this exit.

TABLE 49. Parameters Passed to ARCDSNEX

Type	Size	Description
IN/OUT	CL44	UNIQUE ARCHIVE DATA SET NAME
INPUT	CL8	JOBNAME
INPUT	CL4	SYSTEM ID as identified to SMF

ARCEODEX — Archive Tape End Of File (CLOSE) Exit

The module name supplied for this exit will receive control after each archive tape is closed. It is used for informational purposes only.

The user is passed the DDNAME in order to determine if it is a close for the primary or the copy tape, and the JFCB in order to see the actual tape volumes used. The data set name and volume serials in the JFCB can be useful for interfacing to tape management systems that read the JFCB at job termination time only, rather than hooking directly into tape open and close routines. It can also be useful for generating tape labels, or for detecting when a multiple volume archive tape was created.

Use sysparm ARCEODEX to specify the module name to be invoked for this exit.

TABLE 50. Parameters Passed to ARCEODEX

Type	Size	Description		
INPUT	H	OPTION - Type of CLOSE call	H'1'	Regular tape close
			H'2'	Final tape close -job is ending
INPUT	CL8	DDNAME	CL8'ARCHIVE0'	Close for primary tape
			CL8'ARCHIVEC'	Close for copy tape

TABLE 50. Parameters Passed to ARCEODEX

Type	Size	Description
INPUT	XL176	JFCB - The JFCB for the named DDNAME after the tape close has been done
INPUT	XL110	ARCHVOLS - Image of the ARCHVOLS record as it will be stored in the ARCHVOLS subfile

ARCHIVEX — Process Archive Request

The module name supplied for this exit will receive control from the CA-Disk archive function after it has retrieved the format-1 DSCB and verified the data set exists and is OK to process. The user can request that CA-Disk reject the data set for processing or continue on to process the archive request.

Use sysparm ARCHIVEX to specify the module name to be invoked for this exit.

TABLE 51. Parameters Passed to ARCHIVEX

Type	Size	Description			
INPUT	CL8	COMMAND IDENTIFIER - CL8'ARCHIVE'			
INPUT	CL1	ENVIRONMENT	'B'	BATCH	
			'T'	TSO	
OUTPUT	HW	Specify return code	H'1'	Continue Normal Processing	
			H'2'	Terminate Processing For Data Set	
INPUT	XL256	DATA SET IDENTIFIER - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.			
INPUT	CL44	NEW NAME - If not spaces, NEWNAME for data set			
INPUT	XL140	VOLUME IDENTIFIER - Format-4 DSCB. VOLUME SERIAL in positions 1 - 6.			
IN/OUT	XL3	EXPIRATION DATE - XL3'yydddd' (BINARY)			
IN/OUT	CL4	DISPOSITION INDICATORS	CL1	S	SCRATCH
				N	NOSCRATCH
			CL1	U	UNCAT
				N	NOCAT
				R	RECAT
			CL1	E	Explicit ARCHIVE
			CL1	Y	SIMULATE
				NOT Y	LIVE
INPUT	HW	MEMBER COUNT - If MEMBERS specified			
INPUT	nXL8	MEMBER LIST - List of PDS member names			
INPUT	CL8	Password from PASSWORD command parameter			
INPUT	CL1	RESET parameter	Y	YES	
			N	NO	

ARESPREX — Auto-Restore Screening Exit

The module name supplied for this exit will receive control during the auto-restore function of CA-Disk. It is invoked by the auto-restore system task after CA-Disk has verified that an index record exists for the data set being restored. It is invoked prior to allocation of the archive device, so if this exit rejects the restore request, no device will be allocated.

The purpose of this exit is to let you screen the auto-restore request and, if desired, reject it. You can also put a logging facility in this exit to track the number and type of auto-restores being processed, or check the device type in the ARCHVOLS record to see if the data set was archived to tape or disk. The only parameter that can be modified by the exit is the result code. Although this exit is invoked by a system task, it is not required that it be reentrant. Take care in coding the exit to ensure that it will not cause an abend. Although the system task will recover and post the waiting task, the restore will not be processed. Your exit should issue an ESTAE macro to set up a recovery environment of its own.

Use sysparm ARESPREX to specify the module name to be invoked for this exit.

TABLE 52. Parameters Passed to ARESPREX

Type	Size	Description		
INPUT	CL44	DATA SET NAME		
INPUT	CL6	VOLUME from which data set was archived		
INPUT	CL6	PSEUDO VOLUME (if the catalog management hook is installed) or TARGET VOLUME (if the s213 hook is installed)		
INPUT	XL256	DSNINDEX RECORD		
INPUT	HW	ADDRESS SPACE ID OF ORIGINATING TASK		
INPUT	CL8	JOB NAME OR TSO ID OF ORIGINATING TASK		
OUTPUT	HW	Specify return code	H’0’	Continue Normal Processing
			H’1’	Defer (queue) the Request
			H’8’	Terminate Processing For Request
INPUT	XL110	ARCHVOLS RECORD		
INPUT	XL167	Auto-restore control block (ARCB) as described by the ARCB macro in the CA-Disk macro library		

ARPOOLEX — Auto-Restore DASD Pool Exit

The module name supplied for this exit will receive control during the auto-restore function of CA-Disk. It is called only when the auto-restore is driven from the CA-Disk catalog management exit, and not the S213 exit. This exit allows the user to inspect the DASD pool that CA-Disk has assigned for the restore (if any). If the pool name is blanks on input to the exit, either no pools were defined, or none of the pool definitions matched either the data set name or source volume of the data set.

The exit can either modify the pool name passed on input, or it can pass back an explicit volume to which the data set should be restored. If an explicit volume is returned, the pool name will be ignored. The volume on input will always be blank. If a pool name is returned, it must be a valid pool name defined in the DASDPOOL member of your parmlib data set.

For more information on the use of DASD pools in the auto-restore feature, review the topic “Setting Up DASD Pools for Auto-Restore” on page 70 in the Installation Guide.

Although this exit is invoked by a system task, the module need not be re-entrant. However, care should be taken in coding the exit so that it will not cause an abnormal termination.

Use sysparm ARPOOLEX to specify the module name to be invoked for this exit.

TABLE 53. Parameters Passed to ARPOOLEX

Type	Size	Description
INPUT	CL44	DATA SET NAME
INPUT	CL6	VOLUME DATA SET WAS ARCHIVED FROM
INPUT	XL256	DSN ARCHIVE INDEX RECORD
INPUT	XL110	DSN ARCHIVE VOLUME RECORD
IN/OUT	CL8	POOL NAME (must be valid dasdpool)
OUTPUT	CL6	EXPLICIT VOLUME TO RESTORE TO

CCINITEX — BACKUPCC Processing Exit

The module name supplied for this exit will receive control from the CA-Disk BACKUPCC function, and is invoked for each data set selected for Concurrent Copy processing. When this exit is active, the value of sysparm ONCCFAIL is ignored, and the return code(s) from the exit will control processing. Since the exit is LOADed by CA-Disk, it can be coded as a separate module, or as an alias in LPALIB or any LNKLST data set, or it can be located in a JOBLIB or STEPLIB data set.

The following items characterizes this exit:

- The parameter list passed to the exit is resident below the 16MB line
- All of the areas whose addresses are passed in the parameter list reside below the 16MB line
- The module can be coded in any AMODE and any RMODE
- The exit is called via a BASSM 14,15 instruction
- Return to CA-Disk should always be made via a BSM 0,14 instruction

At entry to the exit, register 1 points to a 1-word parameter list. The list contains the address of the Concurrent Copy Initialization User Exit Parameter List (CCIXL). This parameter list is mapped by macro ADSCCIXL.

The parameter list contains, among other things, a flag bit that indicates the setting of sysparm ONCCFAIL. You can utilize this flag in setting the return code. A return code of:

zero	Indicates that processing is to be continued.
non-zero	Indicates that processing is to be canceled. A non-zero return code is honored even if concurrent copy initialization has been successful.

Use sysparm CCINITEX to specify the module name to be invoked for this exit.

TABLE 54. Parameters Passed to CCINITEX

Type	Size	Description		
OUTPUT	X’80’	Default action if CC INIT fails	0	Cancel
			1	Continue
INPUT	HW	Return code from CC initialization	00	Successful
			04	Failed

TABLE 54. Parameters Passed to CCINITEX

Type	Size	Description		
INPUT	HW	Reason code if CXLRETC=4	00	No reason provided
			04	At least part of data set is on hardware lacking cc support
			08	Hardware limits exceeded
			12	Other hardware error
			16	Host limits exceeded
			20	System data mover not available
			24	Other host error
			28	Data Set type not supported for concurrent copy
INPUT	CL6	First volume serial of data set		
INPUT	CL44	Data Set name		

DARCHIEX — Process Deferred Archive Request

The module name supplied for this exit will receive control from the CA-Disk DARCHIVE function after it has retrieved the DARCHIVE COMMAND and before the command is placed on the DARCHIVE request queue.

The user can request that CA-Disk reject the data set for processing or continue on to queue the DARCHIVE request.

Use sysparm DARCHIEX to specify the module name to be invoked for this exit.

TABLE 55. Parameters Passed to DARCHIEX

Type	Size	Description			
INPUT	CL8	COMMAND IDENTIFIER - CL8'DARCHIVE'			
INPUT	CL1	Environment	'B'	BATCH	
			'T'	TSO/ISPF	
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing	
			H'2'	Terminate Processing For Data Set	
INPUT	CL44	DATA SET NAME			
INPUT	CL44	NEW NAME - If not spaces, NEWNAME for data set			
INPUT	CL6	VOLUME - What DASD volume contains data set			
IN/OUT	XL3	EXPIRATION DATE - XL3'yy0ddd' (BINARY)			
IN/OUT	HW	RETENTION PERIOD - Number of days to retain DSN			
IN/OUT	CL4	Disposition Indicators	CL1	S	SCRATCH
				N	NOSCRATCH
			CL1	U	UNCAT
				N	NOCAT
				R	RECAT
			CL1	(unused)	
			CL1	Y	SIMULATE
				NOT Y	LIVE
INPUT	HW	MEMBER COUNT - Not used at this time			

TABLE 55. Parameters Passed to DARCHIEX

Type	Size	Description		
INPUT	nXL8	MEMBER LIST - Not used at this time		
INPUT	CL8	Password from PASSWORD command parameter		
INPUT	CL1	RESET parameter	Y	YES
			N	NO

DBRCPREX — DBRC Control Statement Screening

The module name supplied for this exit will receive control from the DBRC interface after the control statements have been formatted and prior to writing them to the data set identified by the //DBRCCARD dd statement.

The user can request that CA-Disk not write the control statements, modify the control statements, or continue processing.

Use sysparm DBRCPREX to specify the module name to be invoked for this exit.

TABLE 56. Parameters Passed to DBRCPREX

Type	Size	Description		
INPUT	nCL80	DBRC CONTROL STATEMENTS		
OUTPUT	HW	Specify Return Code	H'0'	Continue Normal Processing
			H'4'	Control Statements MODIFIED
			GT H'4'	Do Not Write Control Statements

DCCMPEXn — Data Compression

The module name supplied for this exit will be given control to compress archive records. Technique number (n) is specified by sysparm DCCOMPTC. Only technique numbers 8 through F can be specified for user exits.

Use sysparm DCCMPEXn to specify the module name to be invoked for this exit.

TABLE 57. Parameters Passed to DCCMPEXn

Type	Size	Description
INPUT	FW	INPUT RECORD LENGTH - length of the record in parameter 2
INPUT	XL	ARCHIVE RECORD - data record to be compressed
OUTPUT	FW	OUTPUT RECORD LENGTH - place the length of compressed record here
OUTPUT	XL	OUTPUT RECORD BUFFER - A 45,056-byte buffer to receive the compressed record

Note: Your exit must be able to process records with input length of zero.

DCDCPEXn — Data Decompression

The module name supplied for this exit will be given control to decompress archive records. The exit is passed the record and the length of the record. The exit must decompress the record into the buffer being passed, and return the length of the decompressed record.

Use sysparm DCDCPEXn to specify the module name to be invoked for this exit.

TABLE 58. Parameters Passed to DCDCPEXn

Type	Size	Description
INPUT	FW	INPUT RECORD LENGTH - length of compressed record
INPUT	XL	COMPRESSED RECORD
OUTPUT	FW	OUTPUT RECORD LENGTH - place the length of the decompressed record here
OUTPUT	XL	OUTPUT RECORD BUFFER - A 45,056-byte buffer to receive the decompressed record

Note: Your exit must be able to process records with input length of zero.

DCDSNDEX — Screen Data Sets for Data Compression

The module name supplied for this exit is given control before doing data compression for a data set, to determine:

1. should compression be done on this data set
2. what technique to use for compression

Use sysparm DCDSNDEX to specify the module name to be invoked for this exit.

TABLE 59. Parameters Passed to DCDSNDEX

Type	Size	Description		
INPUT	CL44	DATA SET NAME		
IN/OUT	C	TECHNIQUE NUMBER - change this to indicate the data compression technique used (C’0’ - C’F’)		
IN/OUT	C	DATA COMPRESSION INDICATOR - change this to whether or not data compression is to be done	C’Y’	DO data compression
			C’N’	NO data compression

DEFARCEX — Screen Disposal of Deferred Archive Requests

During dispose processing of deferred (queued) archive requests, the module named for this exit is given control after an entry is retrieved from the queue, but before it has been selected for disposal (just before the data set is processed). The return code indicates if CA-Disk should archive the data set or flag this command and retrieve the next one. If the command is flagged as rejected, it will not be reviewed again, and will be dropped from the ARCHCMDS file when the hold period (specified in sysparm DARCKEEP) has expired.

Use sysparm DEFARCEX to specify the module name to be invoked for this exit.

TABLE 60. Parameters Passed to DEFARCEX

Type	Size	Description
INPUT	CL44	DATA SET NAME
INPUT	CL6	VOLUME SERIAL NUMBER

TABLE 60. Parameters Passed to DEFARCEX

Type	Size	Description			
INPUT	CL2	Disposition ACTION	CL1	S	SCRATCH
				N	NOSCRATCH
			CL1	U	UNCATALOG
				N	NOCATALOG
				R	RECATALOG
OUTPUT	HW	Specify Return Code	H’1’	Continue Normal Processing	
			H’2’	Reject Processing For Command	
INPUT	XL3	Expiration Date		—	XL3’yydddd’ (Binary)

DERASEEX — Process Erase TSO Deferred Processing Request

The module name supplied for this exit will receive control from the CA-Disk DERASE function after it has retrieved the DERASE command and before the specified request is deleted.

The user can request that CA-Disk reject the data set for request or continue on to delete the request.

Use sysparm DERASEEX to specify the module name to be invoked for this exit.

TABLE 61. Parameters Passed to DERASEEX

Type	Size	Description		
INPUT	CL8	COMMAND IDENTIFIER - CL8’DERASE’		
INPUT	CL1	Environment	‘T’	TSO
			‘B’	Batch
OUTPUT	HW	Specify Return Code	H’1’	Continue Normal Processing
			H’2’	Terminate Processing For Request
INPUT	CL44	DATA SET NAME - DSN for record to be deleted		
INPUT	CL1	TYPE TO BE DELETED	A	DARCHIVE
			R	DRESTORE

DINXUFEX — Update DSNINDEX User Field

The module name supplied for this exit will receive control from CA-Disk before creating or updating the DSNINDEX record in the Files Data Set. It is invoked during processing for the ARCHIVE, BACKUP, VBACKUP, RESTORE, RECOVER and VRECOVER functions. For each data set being processed by one of these functions, this exit is invoked, just before the DSNINDEX record is written to the Files Data Set.

CA-Disk allows only one field in the DSNINDEX record to be updated by this exit; the 25-byte DSNUSERF user field. CA-Disk ignores any other changed fields passed back in the DSNINDEX record.

Note: User's who install PTF SS02159 MUST linkedit this user exit using AMODE=31.

Use sysparm DINXUFEX to specify the module name to be invoked for this exit.

TABLE 62. Parameters Passed to DINXUFEX

Type	Size	Description		
IN/OUT	XL256	DSNINDEX RECORD - This is the archive data set index record from the files data set. DSNUSERF in positions 231 - 256 is the only field for which updates will be honored. Turn to "Subfile Record Format — (DSNINDEX)" on page 462 for a complete description of this record.		
INPUT	CL8	CA-Disk Function -This is the function processing when the exit is invoked.	Valid Values:	CL8'ARCHIVE'
				CL8'BACKUP'
				CL8'VBACKUP'
				CL8'RESTORE'
				CL8'RECOVER'
				CL8'VRECOVER'

DRESTOEX — Screen Deferred Restore Request

The module name supplied for this exit will receive control from the CA-Disk DRESTORE function after it has retrieved a DRESTORE command and before the command is placed on the DRESTORE request queue.

The user can request that CA-Disk reject the data set for processing, or continue on to queue the DRESTORE request.

Use sysparm DRESTOEX to specify the module name to be invoked for this exit.

TABLE 63. Parameters Passed to DRESTOEX

Type	Size	Description			
INPUT	CL8	Command Identifier	CL8'DRESTORE'		
INPUT	CL1	Environment	'B'	Batch	
			'T'	TSO/ISPF	
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing	
			H'2'	Data Set Processing Terminated	
INPUT	CL44	DATA SET NAME			
INPUT	CL44	NEW NAME - If not spaces, NEWNAME for data set			
IN/OUT	CL6	VOLUME - What DASD volume contains data set			
IN/OUT	XL3	EXPIRATION DATE - XL3'yydddd' (BINARY)			
IN/OUT	HW	Retention Period - Number of days to retain DSN			
IN/OUT	CL4	Disposition Indicators	CL1	S	SCRATCH specified
			CL1	N	NOCAT specified
				R	RECAT specified
			CL1	C	CREATE specified
			XL1	X'01'	ERASE specified
INPUT	HW	MEMBER COUNT - Not used at this time			
INPUT	nXL8	MEMBER LIST - Not used at this time			
INPUT	CL14	Data from TIMEDATE parameter			

TABLE 63. Parameters Passed to DRESTOEX

Type	Size	Description
IN/OUT	CL8	DASDPOOL name from POOL parameter
INPUT	CL8	Password from PASSWORD command parameter

DSBUSREX — Screen Billing Record Inserts

This module is given control during DASD billing just prior to inserting or updating a data set billing record in the files data set. Its intended use is to allow the user code field to be reset.

Use sysparm DSBUSREX to specify the module name to be invoked for this exit.

TABLE 64. Parameters Passed to DSBUSREX

Type	Size	Description		
INPUT	CL2	Type Of Function	'SI'	Insert record into file
			'SR'	Replace existing record
IN/OUT	XL	Dasd Billing Record - Turn to "Subfile Record Format — (DASDSPCB)" on page 460 for a complete description of the fields in this record		
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Ignore This Insert / Replace Request

DSCBCVEX — Format-1 DSCB Conversion Exit Location

The module name supplied for this exit will receive control after a new user data set has been allocated on DASD (for example, during Restore, Move/Copy, and PDS Compress processing) and by the DSCB Update Utility. This exit can update or move values of CA-Disk maintained fields kept in the format-1 DSCB.

A sample conversion exit has been provided in member ADSEX016 of the installation library. This sample exit will convert format-1 DSCBs maintained with an Open SVC from a release prior to 8.1 to the Release 8.1 offsets.

By default, the supplied conversion exit will retain the last using job name, move the last modified date, reset the open count, and update the SVC MODE. This will only do conversion if the current SVC mode is 1, 2 or 3 (indicating that it has not be converted before).

Conversion is done using the old default offsets. If you have modified the Open SVC and maintained fields at different offsets, you must modify this exit to reflect your changes. Also, if you want to maintain the creating job name instead of the last using job name, you must update the exit to reflect this.

You should not begin using this conversion exit until you have the new Open SVC installed. Since the old Open SVC does not know of the new offsets, these fields can become corrupted after they are converted.

Once you have determined that this exit will work as desired, specify sysparm DSCBCVEX with the value of ADSEX016 or whatever you have named your conversion exit. This value will cause conversion to take place as CA-Disk reallocates data sets during its processing.

TABLE 65. Parameters Passed to DSCBCVEX

Type	Size	Description		
IN/OUT	XL140	Data Set Identifier. Format-1 DSCB to be converted. This format-1 DSCB is as it was prior to the current CA-Disk processing. You must update this format-1 DSCB to reflect the proper values at the new offsets.		
IN/OUT	XL140	Data Set Identifier. Format-1 DSCB to be converted. This format-1 DSCB also needs to be converted. This format-1 DSCB also contains other updates made by CA-Disk function currently processing the data set.		
INPUT	H	Count Of Fields To Be Changed. This was input to the calling module. It identifies the number of other fields to be updated at the same time as the fields being converted.		
INPUT	nCL8	Field Name List. This is a list of the field names to be updated at the same time as the fields being converted by this exit. The values of these fields are already in the format-1 DSCB identified in parameter 2.		
OUTPUT	H	Return Code	H'1'	Continue Normal Processing
			Other	Indicates Errors

DSCLSCEX — DSCL Screening Exit

The module name supplied for this exit will receive control when processing any DSCL command. Control is passed after the screening tests have been applied, and directly prior to processing the DSCL action command. The screening tests are based on the DSCL selection criteria, after the volume and data set name filtering has been completed. The user can request that CA-Disk reject or process the request, regardless of what value was found in the result code parameter.

When the DS1CPOIT bit in the VTOC is set to 1 (on), the user exit DSCLSCEX allows the action commands ARCHIVE, MOVE, EXPIRE, RELEASE, or DELETE to take place only if an installation determined number of days has passed since the data set was last referenced. The default value of 0 allows the data set to be moved, regardless of the value in the last used date. For related information, please review the ARCDMEXT section of the DFSMS/MVS V1R3 Installation Exits manual.

Use system parameter DSCLSCEX to specify the module name to be invoked for this exit

Note: Computer Associates provides a module that we use at this exit point to process CHECKPOINTED DATA SETS. If you want to use this exit, the source is available for you to modify. Contact Technical Support for more information.

DSDISPEX — Post Data Set Disposition Action

TABLE 66. Parameters Passed to DSCLSCEX

Type	Size	Description		
IN/OUT	HW	The value on input informs the exit what CA-Disk action will take place if unmodified by the exit. The value on output (if changed) instructs CA-Disk to take the new action.	H'0'	Continue Normal Processing
			H'8'	Terminate Processing For Request
INPUT	XL256	Data Set Identifier. Format-1 DSCB and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.		
INPUT	XL140	Volume Identifier. Format-4 DSCB for data set. Volume serial is in position 1 to 6. If the data set to be processed is VSAM, only the volume serial will be passed.		
INPUT	CL44	Catalog Name - If VSAM, the catalog name in which the component is defined (if catalog scan)		
INPUT	CL8	Function Code	'ARCHIVE'	archive the data set
			'BACKUP'	backup the data set
			'COPY'	copy the data set
			'DELETE'	delete the data set
			'EXPIRE'	expire the data set
			'MOVE'	move the data set
			'RELEASE'	release space from the data set
			'REPORT'	report on the data set
			'VBACKUP'	backup a volume
INPUT	CL8	Disposition Value of DISP= DSCL ARCHIVE action command parameter.	'DELETE'	delete data set
			'KEEP'	do not scratch or recatalog set
			'RECAT'	recatalog data set to the CA-Disk pseudo volser
OUTPUT	F	This optional parameter indicates the number of days since last referenced for the data set to be eligible for processing. The default value is '0'.		

A post-disposition exit, this exit is given control during backup and archive processing, after dispose processing is complete.

Use sysparm DSDISPEX to specify the module name to be invoked for this exit.

TABLE 67. Parameters Passed to DSDISPEX

Type	Size	Description		
INPUT	XL256	DATA SET IDENTIFIER - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.		
INPUT	CL4	DISPOSITION FIELD - Disposition field in the TASC format, found in RETVOLSPEC table, indicates the action CA-Disk performed on the data set.		
INPUT	XL140	VOLUME IDENTIFIER - Format-4 DSCB for data set. Volume serial is in positions 1 to 6.		
INPUT	XL1	SIMULATION indicator	X'01'	If in SIMULATE mode
			X'00'	otherwise it will contain this

DSNDELEX — Screen Index Maintenance Deletes (Prior Exit)

The module name supplied for this exit will receive control from the CA-Disk index maintenance functions VOLDELETE, DSNDELETE and PURGE prior to making a decision to delete an index record for an archived data set from the DSNINDEX. The exit routine will receive control for every index record scanned, prior to any CA-Disk decisions to determine if it should be kept or deleted.

The user can request that CA-Disk continue to decide what to do with the record, to delete it, or to keep it in the DSNINDEX.

The Common File Handler Interface (described in "Common File Handler Interface" on page 33), should not be used from this exit to update a record. The update you perform can be lost when you return control and the original function rewrites the record as initially passed to the exit.

Use sysparm DSNDELEX to specify the module name to be invoked for this exit.

TABLE 68. Parameters Passed to DSNDELEX

Type	Size	Description		
INPUT	XL256	DSNINDEX RECORD - This is the archive data set index record from the files data set. DSNAME is in positions 1 - 44. Turn to "Subfile Record Format — (DSNINDEX)" on page 462 for a complete description of this record.		
INPUT	XL1	SIMULATION indicator	X'01'	If in SIMULATE mode
			X'00'	otherwise it will contain this
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Keep Record, Do Not Delete It
			H'3'	Delete Record

DSNDLPEX — Screen Index Maintenance Deletes (After Exit)

The module name supplied for this exit will receive control from the CA-Disk index maintenance functions VOLDELETE, DSNDELETE and PURGE, the CA-Disk index update function DELETE, the ISPF function LIST DSNINDEX AND CHANGE EXPIRATION DATE, and the CA-Disk function MERGE after CA-Disk has determined whether an index record is to be deleted, but prior to actually deleting it from the DSNINDEX. The user can request that CA-Disk continue to take the indicated action against the data set or override the decision made by CA-Disk. The action to be taken by CA-Disk is indicated in the third parameter RESULT.

The Common File Handler Interface (described in "Common File Handler Interface" on page 33), should not be used from this exit to update a record. The update you perform can be lost when you return control and the original function rewrites the record as initially passed to the exit.

Use sysparm DSNDLPEX to specify the module name to be invoked for this exit.

TABLE 69. Parameters Passed to DSNDLPEX

Type	Size	Description		
INPUT	XL256	DSNINDEX Record - This is the archive data set index record from the files data set. DSNAME is in positions 1 - 44. Turn to "Subfile Record Format — (DSNINDEX)" on page 462 for a complete description of this record.		
INPUT	XL1	Simulation indicator	X'01'	If in simulate mode
			X'00'	otherwise it will contain this

TABLE 69. Parameters Passed to DSNDLPEX

Type	Size	Description		
IN/OUT	HW	RESULT - CA-Disk Input is H'1' or less indicating delete the DSNINDEX record, and H'2' to keep the record. If the user wishes CA-Disk to reject the delete request and keep the DSNINDEX record, place a value of 2 in this parameter and return to CA-Disk; otherwise place a H'1' in this parameter. If the "Common File Handler Interface" on page 33 is used in this exit, use the return codes that specify repositioning required.	H'0'	Delete the DSNINDEX record, and reposition it
			H'1'	Delete the DSNINDEX record
			H'2'	Keep the DSNINDEX record
			H'3'	Keep and reposition the DSNINDEX record

FMSSAFEX — FMS After Selection Exit

The module name supplied for this exit receives control after standard FMS selection processing is complete for the DSNINDEX record. This exit is passed a copy of the DSNINDEX record selected for recovery, and a Y/N flag byte.

The initial setting of the flag is either C'Y' or C'N', indicating whether or not DSCL processing selected the data set. By setting this flag to C'Y', the data set will be selected regardless of any further DSCL statements. This flag can also be set to C'N', forcing a bypass of the data set regardless of any further DSCL statements.

Use sysparm FMSSAFEX to specify the module name to be invoked for this exit.

TABLE 70. Parameters Passed to FMSSAFEX

Type	Size	Description		
INPUT	CL256	Copy of the DSNINDEX record		
IN/OUT	CL1	Flag Byte Return	C'Y'	Select the data set
			C'N'	Bypass the data set

FMSSPREX — FMS Prior Selection Exit

The module name supplied for this exit receives control prior to FMS performing any selection processing for the DSNINDEX record. This exit is passed a copy of the DSNINDEX record selected for recovery, and a Y/N flag byte.

The initial setting of the flag is a hex value of zero (X'00'), indicating normal selection processing will continue. By setting this flag to C'Y', the data set will be selected regardless of any further DSCL statements. This flag can also be set to C'N', forcing a bypass of the data set regardless of any further DSCL statements.

Use sysparm FMSSPREX to specify the module name to be invoked for this exit.

TABLE 71. Parameters Passed to FMSSPREX

Type	Size	Description		
INPUT	CL256	Copy of the DSNINDEX record		
IN/OUT	CL1	Flag Byte Return	C'Y'	Select the data set
			C'N'	Bypass the data set

GDGCATEX — GDG Data Set Catalog Action

The module name supplied for this exit will receive control during functions RESTORE, RECOVER, MOVE, COPY, PDS COMPRESS, and IXCATLG prior to any catalog action for GDG data sets. A sample coding of this exit can be found in member GDGCATEX in the DMSASM library.

Default processing will not force a GDG catalog entry out of a full GDG sphere, but instead will leave the processed GDS in a deferred or uncataloged status. This exit gives the user the ability to either force the catalog action, or allow normal processing to take it course.

The GDG Status parameter, described in the following table, indicates the catalog status of the source data set (i.e., BACKUP, MOVE, or COPY).

Use sysparm GDGCATEX to specify the module name to be invoked for this exit.

TABLE 72. Parameters Passed to GDGCATEX

Type	Size	Description		
INPUT	CL44	Data set name or NEWNAME (if specified)		
INPUT	CL1	GDG Status	H	Active (normal for non-SMS GDS, whether originally cataloged or not)
			N	Deferred
			M	Rolled Off
OUTPUT	CL1	Catalog Action requested	C	Unconditionally catalog (or Roll-in) the GDS
			‘ ‘	Continue with normal processing
OUTPUT	H	Results:	0	Process according to Catalog Action setting
			1+	Bypass any catalog action. M3729 will be issued.

Note: When using the DFSMSdss Data Mover interface, your user exit will not be invoked during RESTORE processing because DFSMSdss allocates the data set for RESTORE.

ICHEKAEX — RACF ICHEINTY Processing Post-ICHEINTY Exit

The module name supplied for this exit will receive control from all functions of CA-Disk that need to determine whether a data set has a discrete profile or a generic profile.

CA-Disk issues the IBM ICHEINTY macro to determine if a data set has a discrete and/or a generic profile. This exit is invoked after issuing this macro.

The user is given an opportunity in this exit to look at the data set name and volume after issuing the ICHEINTY macro. The user is also given an opportunity to review the return code passed back from the macro. The return codes issued by the ICHEINTY macro are documented in the IBM SPL RACF manual.

Use sysparm ICHEKAEX to specify the module name to be invoked for this exit.

TABLE 73. Parameters Passed to ICHEKAEX

Type	Size	Description	
INPUT	XL1	D	CA-Disk checks to see if the data set has a discrete profile.
		G	CA-Disk checks to see if the data set has a generic profile.
INPUT	XL44	Data set name to be checked.	

TABLE 73. Parameters Passed to ICHEKAEX

Type	Size	Description
INPUT	Xl6	VOLSER of data set to be checked. If parameter 1 is G, VOLSER is ignored.
INPUT	HW	RETURN CODE - returned from the ICHEINTY macro

Note: Not available when using the DFSMSdss Data Mover interface.

ICHEKPEX — RACF ICHEINTY Processing Pre-ICHEINTY Exit

The module name supplied for this exit will receive control from all functions of CA-Disk that need to determine whether a data set has a discrete profile or a generic profile.

CA-Disk issues the IBM ICHEINTY macro to determine if a data set has a discrete and/or a generic profile. This exit is invoked prior to issuing this macro.

The user can modify the data set name or the VOLSER. The user can also prevent the issuing of the macro by changing the return code to a value greater than zero. If the value is changed to greater than zero, this is equivalent to saying to CA-Disk that the data set does not have either a discrete or a generic profile, depending on what type of profile is being looked for.

Use sysparm ICHEKPEX to specify the module name to be invoked for this exit.

TABLE 74. Parameters Passed to ICHEKPEX

Type	Size	Description	
INPUT	XL1	D	CA-Disk checks to see if the data set has a discrete profile.
		G	CA-Disk checks to see if the data set has a generic profile.
INPUT	XL44	Data set name to be checked.	
INPUT	Xl6	VOLSER of data set to be checked. If parameter 1 is G, VOLSER is ignored.	
INPUT	HW	Return Code. Set to a value greater than zero if you want the ICHEINTY macro to not be issued	

Note: Not available when using the DFSMSdss Data Mover interface.

IXCTLGEX — User Exit for IXCATLG Utility

This module is invoked after all selection testing has been performed by the IXCATLG procedure. It can be used either to monitor those data sets selected by the maintenance program or to change the selection flag to either include or exclude the data set.

When this exit is invoked, the selection flag has already been set to the value CA-Disk has determined is correct. If the value is Y, the data set will be cataloged to the CA-Disk pseudo-volume as specified by sysparm RECATVOL, to enable it as a candidate to be auto-restored. If the value is N, the data set will not be processed. This exit can change the selection flag to either a Y or an N, regardless of which input value was passed to the exit.

Use sysparm IXCTLGEX to specify the module name to be invoked for this exit. (A sample exit is provided in the CA-Disk installation library as member IXCTLGEX.)

TABLE 75. Parameters Passed to IXCTLGEX

Type	Size	Description		
INPUT	XL256	DSNINDEX record		
IN/OUT	CL1	Selection flag	Y	data set has been selected for processing
			N	data set has been excluded from processing

IXMREVEX — Screen Index Maintenance Deletes (Post Exit)

Use sysparm IXMREVEX to specify the module name to be invoked for this exit. This exit receives control from the DSNDELETE, PURGE, and VOLDELETE commands after CA-Disk has determined what action to take on the DSNINDEX record, but prior to actually taking the action. The action to be taken by CA-Disk is indicated in the third parameter RETURN CODE. The user can request that CA-Disk continue to take the indicated action against the data set, or you can override the action entirely.

Caution: "Common File Handler Interface" on page 33 should **NOT** be used from this exit to update a record. If you use this interface to update a record, the update you perform can be lost when returning control and the original function rewrites the record initially passed to the exit.

The DSNDLPEX user exit must not be specified at the same time as this exit. If both are specified, the job will abend with a U100 accompanied by an appropriate message.

TABLE 76. Parameters Passed to IXMREVEX

Type	Size	Description		
INPUT	XL256	DSNINDEX record		
INPUT	XL1	Simulation Flag	X'01'	Simulation mode
			X'00'	Live mode
IN/OUT	HW	Return Code	H'0'	Record will be kept — Input
			H'1'	Record will be deleted — Input
			H'2'	Record will be kept — Input
			H'3'	Record will be updated — Input
			H'1'	Delete DSNINDEX record — Output
			H'2'	Keep DSNINDEX record — Output
		Note: Do not set this field unless you want to alter its disposition. An output return code of other than its original input return code or 1 or 2 will result in an abend.		
INPUT	CL16	WHYFLAGS		
INPUT	XL256	Catalog locate work area		
INPUT	XL212	Management Class		

Additional Notes:

1. Parm 4 are the whyflags as seen in the DD SMSPRINT output. A complete list of these flags are documented under the topic "SMSPRINT and WHYFLAGS Diagnostics" on page 331 in the User's Guide.
2. If a Catalog Locate was performed on the data set, then its workarea is contained in parm 5. Offset 6 contains the data set's cataloged volser.

3. If the data set has a Management Class, then the CA-Disk mapping is contained in parm 6 as mapped out by the SMSMC mapping macro.

LISTDMEX — Screen TSO LISTDMS Command

The module name supplied for this exit will receive control from the CA-Disk LISTDMS function after it has retrieved the LISTDMS command and before it is processed.

The user can request that CA-Disk reject the request or continue processing it. See "LISTDVEX — List Archive Index (LISTD/LISTV) Exit" on page 225 for related information.

Use sysparm LISTDMEX to specify the module name to be invoked for this exit.

TABLE 77. Parameters Passed to LISTDMEX

Type	Size	Description			
INPUT	CL8	Command Identifier - CL8'LISTDMS'			
INPUT	CL1	Environment	'T'	TSO (always)	
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing	
			H'2'	Terminate Processing For Request	
INPUT	CL44	Data Set Name - DSN for record to be listed			
INPUT	CL3	Data to be listed	N	DON'T LIST	
			Y	CL1	List Data Set Status
				CL1	List Data Set Allocation Data
				CL1	List Data Set Archive Data

LISTDVEX — List Archive Index (LISTD/LISTV) Exit

The module name supplied for this exit will receive control after each ARCHVOLS or DSNINDEX entry has been selected for printing. The exit, if used, is also active under the TSO LISTDMS function.

The user is passed the file name (ARCHVOLS or DSNINDEX), the index record itself, and a one-character switch that can be set to suppress the printing of the record.

Use sysparm LISTDVEX to specify the module name to be invoked for this exit.

TABLE 78. Parameters Passed to LISTDVEX

Type	Size	Description			
INPUT	CL8	Filename	CL8'ARCHVOLS'		
			CL8'DSNINDEX'		
INPUT	XLnnn	Index Record For Named File - Turn to "Subfile Record Format — (ARCHVOLS)" on page 459 for field lengths within the ARCHVOLS subfile, or "Subfile Record Format — (DSNINDEX)" on page 462 for field lengths within the DSNINDEX subfile.			
IN/OUT	CL1	Print Switch	CL1	'Y'	Display the index entry
			CL1	'N'	Do NOT display the entry

LISTREEX — Screen TSO LISTREQ Command

The module name supplied for this exit will receive control from the CA-Disk LISTREQ function after it has retrieved the LISTREQ command and before it is processed.

The user can request that CA-Disk reject the request or continue processing it.

Use sysparm LISTREEX to specify the module name to be invoked for this exit.

TABLE 79. Parameters Passed to LISTREEX

Type	Size	Description		
INPUT	CL8	Command Identifier - CL8'LISTREQ'		
INPUT	CL1	Environment	'T'	TSO (always)
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Terminate Processing For Request
INPUT	Cl44	Data Set Name - DSN for record to be listed		
INPUT	CL1	Type To Be Listed	A	DARCHIVE
			R	DRESTORE

MERDSNEX — Data Set Being Deleted Through Merge

The module name supplied for this exit will receive control from the CA-Disk MERGE function after an archive DSNINDEX record has been deleted from the files data set.

No return code is accepted by this exit. The MERGE process cannot be altered.

Use sysparm MERDSNEX to specify the module name to be invoked for this exit.

TABLE 80. Parameters Passed to MERDSNEX

Type	Size	Description		
INPUT	XL256	DSNINDEX RECORD - This is the archive data set index record from the files data set. Data set name is in positions 1 to 44. See "Subfile Record Format — (DSNINDEX)" on page 462 for a complete description of this record.		
INPUT	XL1	Simulation Indicator	X'01'	If in SIMULATE mode
			X'00'	otherwise it will contain this

MERVOLEX — Archive Volume Being Deleted By Merge

The module name supplied for this exit will receive control from the CA-Disk MERGE function after an ARCHVOLS record has been deleted from the files data set. (If a multivolume ARCHVOLS entry was deleted, only the first ARCHVOLS record in the chain is passed to this exit. Use exit "TAPEDLEX — Archive Volume Being Deleted from Archives" on page 249 instead of this one if you need to get the list of all ARCHVOLS that were in the chain.)

No return code is accepted by this exit. The MERGE process cannot be altered.

Use sysparm MERVOLEX to specify the module name to be invoked for this exit.

TABLE 81. Parameters Passed to MERVOLEX

Type	Size	Description		
INPUT	XL110	ARCHVOLS RECORD - This is the archive VOLUME index record from the files data set. Volume is in positions 1 - 6. See for a complete description of this record.		
INPUT	XL1	Simulation Indicator	X’01’	If in SIMULATE mode
			X’00’	otherwise it will contain this

MIGDISEX — Screen Sequential Migrate Disposition Request

The module name supplied for this exit will receive control from sequential migrate after it has copied a data set to tape, but before it has scratched/recataloged it. The exit can examine disposition fields and optionally reject the disposition request.

Use sysparm MIGDISEX to specify the module name to be invoked for this exit.

TABLE 82. Parameters Passed to MIGDISEX

Type	Size	Description		
INPUT	XL1	Simulate Mode Indicator - X’01’ implies a Simulate execution of CA-Disk. X’00’ implies “Live”		
INPUT	XL147	Disposition Record - This record contains information about the data set copied to tape. Its format is as follows:		
			AL2	RECORD DESCRIPTOR WORD
		+002	XL2	(RESERVED)
		+004	CL44	DATA SET NAME
		+048	CL30	1 to 5 OUTPUT TAPE VOLSERS
		+078	CL30	VOLSERS OF COPY TAPES
		+108	XL4	UCBTYPE
		+108	XL1	DS1DSIND FMT 1 field (contains RACF & PASSWORD bits)
		+109	XL3	(RESERVED)
		+112	HW	NUMBER OF TAPE VOLUMES USED
		+114	HW	DATA SET SEQUENCE NUMBER ON TAPE
		+116	XL3	(RESERVED)
		+119	XL3	EXPIRATION DATE XL3’YYDDDD’
		+122	CL8	(RESERVED)
		+130	XL2	DATA SET LRECL
		+132	XL2	DATA SET BLKSIZE
		+134	XL2	TRACKS ALLOCATED
		+136	XL2	DEVICE TYPE OF DASD
		+138	XL1	DATA SET RECFM
		+139	XL1	DATA SET KEY LENGTH
		+140	XL1	DATA SET OPTCD
		+141	CL4	REASON DATA SET WAS MIGRATED
		+145	HW	NUMBER OF VOLUMES OF DATA SET MIGRATED
		+147	CL120	VOLUMES(S) MIGRATED FROM

TABLE 82. Parameters Passed to MIGDISEX

Type	Size	Description		
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Do Not Take Disposition For Data Set

MIGGDGEX — Return GDG Limit to Sequential Migrate

This module is given control after sequential data set migration has failed to obtain a generation limit from the GDGDASDG table for a GDG data set. The user can supply the GDG limit for the data set.

Use sysparm MIGGDGEX to specify the module name to be invoked for this exit.

TABLE 83. Parameters Passed to MIGGDGEX

Type	Size	Description		
INPUT	CL44	INPUT CL44 DATA SET NAME		
OUTPUT	HW	GDG LIMIT COUNT to use for the data set		
OUTPUT	HW	Specify Return Code	H'1'	Generation Limit Returned
			H'2'	Use Default Generation Limit

MIGREXIT — Sequential Migration Disposition Screening

This module is invoked after a data set has been selected for sequential data set migration, but before a disposition control record has been built.

The user can reject migration of the data set or alter DSCB information for a data set to be migrated. Use sysparm MIGREXIT to specify the module name to be invoked for this exit.

TABLE 84. Parameters Passed to MIGREXIT

Type	Size	Description		
IN/OUT	XL256	Data Set Identifier - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area. Only the expiration date can be modified; all other changes will be ignored.		
INPUT	L140	Volume Identifier - Format-4 DSCB for data set. VOLSER is in positions 1 to 6.		
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Terminate Processing For data set

RAADDAEX — RACF RACDEF-ADDVOL Post Processing Exit

The module name supplied for this exit will receive control from the CA-Disk module responsible for adding volumes to RACF profiles. It receives control after the RACDEF-ADDVOL macro has been issued.

The exit can inspect parameters passed to RACF and change the return code returned from it. Use sysparm RAADDAEX to specify the module name to be invoked for this exit.

TABLE 85. Parameters Passed to RAADDAEX

Type	Size	Description
INPUT	CL44	Profile Name That Was Modified
INPUT	CL6	OLD VOLUME - Existing volume in the profile data after which the new volume was added.

TABLE 85. Parameters Passed to RAADDAEX

Type	Size	Description		
INPUT	CL6	NEW VOLUME - New volume serial associated with the RACF profile.		
INPUT	CL1	Data set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set
IN/OUT	HW	The RETURN CODE from the RACDEF (ADDVOL) is placed in this parameter. It can be changed by the exit. Use one of the following valid values:		
		H'0'	Volume Added	
		H'4'	Resource Previously Defined	
		H'8'	Failed By Racf Exit	
		H'16'	Critical Error	

Note: When using the DFSMSdss Data Mover interface, CA-Disk initially allocates the data set allowing your exit to be called during BACKUP. However, since DFSMSdss allocates the data set for RESTORE, your user exit will not be invoked during RESTORE processing.

RAADDPEX — RACF RACDEF-ADDVOL Processing Pre-RACHECK Exit

The module name supplied for this exit will receive control from the CA-Disk module responsible for adding volumes to RACF profiles. It receives control before the RACDEF-ADDVOL macro is issued.

The exit can change RACDEF-ADDVOL parameters, or cause the RACDEF-ADDVOL to be bypassed and return its own return code. Use sysparm RAADDPEX to specify the module name to be invoked for this exit.

TABLE 86. Parameters Passed to RAADDPEX

Type	Size	Description		
INPUT	CL44	Profile Name To Be Modified		
INPUT	CL6	OLD VOLUME - Existing volume in the profile data after which the new volume is to be added.		
INPUT	CL6	NEW VOLUME - New volume serial to be associated with the RACF profile.		
INPUT	CL1	Data set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set
OUTPUT	HW	If your exit performs the RACDEF ADDVOL, set the appropriate return code, as indicated below. If CA-Disk is to process the RACDEF ADDVOL function, do not set this return code.		
		H'0'	Volume Added	
		H'4'	Resource Previously Defined	
		H'8'	Failed By Racf Exit	
		H'16'	Critical Error	

Note: When using the DFSMSdss Data Mover interface, CA-Disk initially allocates the data set allowing your exit to be called during BACKUP. However, since DFSMSdss allocates the data set for RESTORE, your user exit will not be invoked during RESTORE processing.

RACFDAEX — Decode CA-Disk RACF Encoded Profile Name (Post-decode)

The module name supplied for this exit will receive control from the CA-Disk RACF processing when a request has been made to decode a CA-Disk RACF encoded profile name. Control is passed to the user after CA-Disk has decoded the profile name.

The user can elect to change one of the fields from the decoded name or leave them as they are. Use sysparm RACFDAEX to specify the module name to be invoked for this exit.

TABLE 87. Parameters Passed to RACFDAEX

Type	Size	Description		
INPUT	XL44	CA-Disk RACF ENCODED PROFILE NAME — Format of this name is described in "Format of CA-Disk-Saved Profiles" on page 352		
IN/OUT	XL3	ARCHIVE DATE - Date from decoded profile name. XL3'YYDDDD' (BINARY).		
IN/OUT	XL2	ARCHIVE TIME - Time from decoded profile name. XL2'HHMM' (BINARY unsigned).		
IN/OUT	CL44	DSNAME - Data set name from decoded profile.		
OUTPUT	HW	Specify Return Code	H'1'	Continue Processing, Successful
			H'2'	Error, Reject Decode Request

RACFDPEX — Decode CA-Disk RACF Encoded Profile Name (Pre-decode)

The module name supplied for this exit will receive control from the CA-Disk RACF processing when a request has been made to decode a CA-Disk RACF encoded profile name. Control is passed to the user prior to decoding the profile name.

The user can elect to decode the name, reject the decode request, or indicate to continue processing without change. This exit allows users to implement their encoding technique for CA-Disk RACF profile names.

Use sysparm RACFDPEX to specify the module name to be invoked for this exit.

TABLE 88. Parameters Passed to RACFDPEX

Type	Size	Description		
INPUT	XL44	CA-Disk RACF ENCODED PROFILE NAME — Format of this name is described in "Format of CA-Disk-Saved Profiles" on page 352		
OUTPUT	XL3	ARCHIVE DATE - Date from decoded profile name. XL3'YYDDDD' (BINARY).		
OUTPUT	XL2	ARCHIVE TIME - Time from decoded profile name. XL2'HHMM' (BINARY unsigned).		
OUTPUT	CL44	DSNAME - Data set name from decoded profile.		
OUTPUT	HW	Specify Return Code	H'1'	User Exit Decoded The Encoded Profile Name
			H'2'	Continue With CA-Disk Decode Of Profile Name
			H'3'	Error, Reject Decode Request

RACFEAEX — Encode CA-Disk RACF Profile Name (Post-encode)

The module name supplied for this exit will receive control from CA-Disk RACF processing when a request has been made to encode a CA-Disk data set name into a CA-Disk RACF profile encoded name. Control is passed to the exit after the name has been encoded.

The user can change the encoded name or leave it as it is. Use sysparm RACFEAEX to specify the module name to be invoked for this exit.

TABLE 89. Parameters Passed to RACFEAEX

Type	Size	Description		
INPUT	CL44	DSNAME - Encoded data set name.		
INPUT	XL3	ARCHIVE DATE - Date used in encoding the name. XL3'YYDDDD' (BINARY).		
INPUT	XL2	ARCHIVE TIME - Time used in encoding the name. XL2'HHMM' (BINARY unsigned).		
IN/OUT	XL44	CA-Disk RACF ENCODED PROFILE NAME		
OUTPUT	HW	Specify Return Code	H'1'	Continue Processing, Successful
			H'2'	Error, Reject Encode Request

RACFEPEX — Encode CA-Disk RACF Profile Name (Pre-encode)

The module name supplied for this exit will receive control from CA-Disk RACF processing when a request has been made to encode a CA-Disk data set name into a CA-Disk RACF profile encoded name. Control is passed to the exit prior to encoding the profile name.

The user can elect to encode the name, reject the encode request, or indicate to continue processing without change. This exit allows users to implement their own encoding technique for CA-Disk RACF profile names.

Use sysparm RACFEPEX to specify the module name to be invoked for this exit.

TABLE 90. Parameters Passed to RACFEPEX

Type	Size	Description		
INPUT	CL44	DSNAME - Data set name to be encoded.		
INPUT	XL3	ARCHIVE DATE - Date for use in encoding the name. XL3'YYDDDD' (Binary).		
INPUT	XL2	ARCHIVE TIME - Time for use in encoding the name. XL2'HHMM' (Binary unsigned).		
OUTPUT	XL44	CA-Disk RACF ENCODED PROFILE NAME		
OUTPUT	HW	Specify Return Code	H'1'	User Exit Encoded the Encoded PROFILE NAME
			H'2'	Continue With CA-Disk Encode Of Profile Name
			H'3'	Error, Reject Encode Request

RACFSMEX — Process RACF Protected DSN with Sequential Migrate

The module name supplied for this exit will receive control from the CA-Disk sequential migrate function when it needs to determine if a RACF-protected data set can be processed. When the exit receives control, CA-Disk has determined how to process the data set.

The user exit can elect to override the decision or accept it. Use sysparm RACFSMEX to specify the module name to be invoked for this exit.

TABLE 91. Parameters Passed to RACFSMEX

Type	Size	Description		
INPUT	XL256	DATA SET IDENTIFIER - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.		
INPUT	HW	COUNT OF VOLUMES DSN RESIDES ON		
INPUT	nCL6	LIST OF VOLUMES DSN RESIDES ON		
IN/OUT	HW	Decide To Process Data set	H'1'	Process Racf-protected Data set
			H'2'	Reject Data set From Processing

RACHKAEX — RACF RACHECK Processing Post-RACHECK Exit

The module name supplied for this exit will receive control from the CA-Disk module responsible for RACHECK processing. It receives control after the RACHECK macro has been issued.

The exit can inspect the return code from the RACHECK macro and/or change it. Use sysparm RACHKAEX to specify the module name to be invoked for this exit.

TABLE 92. Parameters Passed to RACHKAEX

Type	Size	Description		
INPUT	CL44	Data set Name Checked		
INPUT	HW	Volume Count - A count of the volumes on which the data set resides. This count indicates how many volumes are contained in the data for parameter 3.		
INPUT	nCL6	List Of Volumes On Which The Data Set Resides		
INPUT	CL1	Requested Access Level	'U'	Update access requested
			'R'	Read access requested
INPUT	CL1	Data Set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set
IN/OUT	HW	The return code from the RACHECK macro is placed in this area. If desired it can be changed to one of the following valid values:		
		H'0'	OK to Process Data Set	
		H'8'	User is not Authorized	
		H'4'	Resource Not Defined	
		H'16'	Critical Error	

RACHKPEX — RACF RACHECK Processing Pre-RACHECK Exit

The module name supplied for this exit will receive control from the CA-Disk module responsible for RACHECK processing. It receives control before the RACHECK macro is issued.

The exit can change RACHECK parameters, or cause the RACHECK to be bypassed and return its own return code. Use sysparm RACHKPEX to specify the module name to be invoked for this exit.

TABLE 93. Parameters Passed to RACHKPEX

Type	Size	Description		
INPUT	CL44	Data Set Name To Be Checked		
INPUT	HW	Volume Count - A count of the number of volumes on which the data set resides. This count indicates how many volumes are contained in the data for parameter 3.		
INPUT	nCL6	List Of Volumes On Which The Data Set Resides		
INPUT	CL1	Requested Access Level	'U'	Update access requested
			'R'	Read access requested
INPUT	CL1	Data Set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set
OUTPUT	HW	If the exit desires to perform the RACF check and to have CA-Disk bypass the RACHECK, place a return code in this parameter. To cause CA-Disk to continue processing, leave this parameter with the value as it was on entry.		
		H'0'	OK TO PROCESS DATA SET	
		H'8'	User is not Authorized	
		H'4'	Resource Not Defined	
		H'16'	Critical Error	

RADEFAEX — RACF RACDEF Processing Post-RACDEF Exit

The module name supplied for this exit will receive control from the CA-Disk module responsible for RACDEF processing. It receives control after the RACDEF macro has been issued.

The exit can inspect the parameters passed and returned from RACDEF processing. Use sysparm RADEFAEX to specify the module name to be invoked for this exit.

TABLE 94. Parameters Passed to RADEFAEX

Type	Size	Description		
INPUT	CL44	Old Data Set Name - The data set used as the model to generate the RACF profile for the NEWNAME data set name.		
INPUT	HW	Old Volume Count - A count of volumes on which the data set resides. This indicates how many volumes are contained in data for parameter 3.		
INPUT	nCL6	List Of Volumes On Which Old Data Set Resides		
INPUT	CL44	New Data set Name - This is the data set that received the RACF profile.		
INPUT	HW	New Volume Count - A count of volumes on which the data set resides. This indicates how many volumes are contained in data for parameter 6.		
INPUT	nCL6	List Of Volumes On Which New Data Set Resides		
INPUT	CL1	Data Set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set

TABLE 94. Parameters Passed to RADEFAEX

Type	Size	Description	
OUTPUT	HW	The return code from the RACDEF macro is placed here. It can be changed by the exit. Use one of the following valid values:	
		H'0'	RACDEF Processed OK
		H'8'	RACF Installation Exit Failed
		H'4'	Resource Previously Defined
		H'16'	Old Data Set not Defined
		H'20'	Critical Error

RADEFPEX — RACF RACDEF Processing Pre-RACDEF Exit

The module name supplied for this exit will receive control from the CA-Disk module responsible for RACDEF processing. It receives control before the RACDEF macro is issued.

Note: If you use the IBM RACF name conversion exit to change nodes within a data set name, you must also use this exit to restore the name to its original form and return it as the first parameter defined below. The exit can change RACDEF parameters, or cause the RACDEF to be bypassed and return its own return code.

Use sysparm RADEFPEX to specify the module name to be invoked for this exit.

TABLE 95. Parameters Passed to RADEFPEX

Type	Size	Description		
IN/OUT	CL44	Old Data Set Name - This is the data set to be used as the model to generate the RACF profile for the NEWNAME data set name.		
INPUT	HW	Old Volume Count - A count of volumes on which the data set resides. This indicates how many volumes are contained in data for parameter 3.		
INPUT	nCL6	List Of Volumes On Which Old Data Set Resides		
INPUT	CL44	New Data Set Name - This is the data set to be given a new RACF profile		
INPUT	HW	New Volume Count - A count of volumes on which the data set resides. This indicates how many volumes are contained in data for parameter 6.		
INPUT	nCL6	List Of Volumes On Which New Data Set Resides		
INPUT	CL1	Data Set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set
OUTPUT	HW	If the exit desires to perform the RACF RACDEF and have CA-Disk bypass the RACDEF, place a return code in this parameter. To cause CA-Disk to continue processing, leave this parameter with the value as it was on entry.		
		H'0'	RACDEF Processed OK	
		H'4'	Resource Previously Defined	
		H'8'	RACF Installation Exit Failed	
		H'16'	Old Data Set not Defined	
		H'20'	Critical Error	

RADELAEX — RACF RACDEF (Delete) Processing Post-Processing

The module name supplied for this exit will receive control from the CA-Disk module responsible for deleting RACF profiles. It receives control after the RACDEF macro has been issued.

The exit can inspect parameters passed to RACDEF and change the return code from the RACDEF macro. Use sysparm RADELAEX to specify the module name to be invoked for this exit.

TABLE 96. Parameters Passed to RADELAEX

Type	Size	Description		
INPUT	CL44	Profile Name Deleted		
INPUT	HW	Volume Count - A count of volumes on which the data set resides. This count indicates how many volumes are contained in the data for parameter 3.		
INPUT	nCL6	List Of Volumes On Which Data Set Resides		
INPUT	CL1	Data Set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set
IN/OUT	HW	The RETURN CODE from RACDEF is placed in this parameter. It can be changed to any of the following values if desired.		
		H'0'	Profile Deleted	
		H'4'	Resource not Defined	
		H'8'	Failed by RACF Exit	
		H'16'	Critical Error	

RADELPEX — RACF RACDEF (Delete) Processing Pre-RACHECK Exit

The module name supplied for this exit will receive control from the CA-Disk module responsible for deleting RACF profiles. It receives control before the RACDEF macro is issued.

The exit can change RACDEF parameters, or cause the RACDEF to be bypassed and return its own return code. Use sysparm RADELPEX to specify the module name to be invoked for this exit.

TABLE 97. Parameters Passed to RADELPEX

Type	Size	Description		
INPUT	CL44	Profile Name To Be Deleted		
INPUT	HW	Volume Count - A count of the number of volumes on which the data set resides. This count indicates how many volumes are contained in the data for parameter 3.		
INPUT	nCL6	List Of Volumes On Which Data Set Resides		
INPUT	CL1	Data Set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set
OUTPUT	HW	If the exit desires to perform the RACDEF and to have CA-Disk bypass the RACDEF, place a return code in this parameter. To cause CA-Disk to continue processing, leave this parameter with the value as it was on entry.		
		H'0'	Profile Deleted	
		H'4'	Resource not Defined	
		H'8'	Failed by RACF Exit	
		H'16'	Critical Error	

RECPRIEX — Screen Recover Requests

The module name supplied for this exit will receive control from the RECOVER proc before selecting data sets to recover. Different selection criteria can be applied to the RECOVER function based on the fields in the DSNINDEX record. Use sysparm RECPRIEX to specify the module name to be invoked for this exit.

TABLE 98. Parameters Passed to RECPRIEX

Type	Size	Description		
INPUT	XL256	DSNINDEX record for the data set to be recovered.		
OUTPUT	HW	Specify return code	H’0’	Continue Processing Data Set Through Normal Selection Criteria (command parms)
			H’1’	Select this Data Set and Bypass the Normal Selection Criteria (command parms)
			H’2’	Do not Select this Data Set, Bypass Normal Selection Criteria (command parms)

RESAFTEX — Post Restore Exit

The module name supplied for this exit will receive control from the RESTORE, DEFERRED RESTORE and RECOVER after restoring the data set or cluster. Use sysparm RESAFTEX to specify the module name to be invoked for this exit.

Note: User's who install PTF SS02159 MUST linkedit this user exit using AMODE=31.

TABLE 99. Parameters Passed to RESAFTEX

Type	Size	Description				
INPUT	XL256	DSNINDEX record for the data set to be restored				
INPUT	XL140	Format-1 DSCB for restored data set				
INPUT	HW	Return code of restore	H’1’	Data Set Restored Okay		
			H’2’	Data Set Restored Okay, But Disposition Failed		
			H’4’	Data Set Restore Failed		
INPUT	XL4	Disposition indicators	CL1	Scratch	U	no scratch
					S	scratch
			CL1	Catalog	C	catalog
					N	no catalog
					R	recatalog
			CL1	Create Date	N	no reset of create date
					C	reset of create date
			XL1	Erase	x’00’	no erase
					x’01’	erase
INPUT	HW	Return code of disposition processing				
INPUT	CL8	Function issuing restore				
INPUT	HW	Count of volumes supplied in VOLUME parameter				
INPUT	nCL6	Volume(s) supplied in VOLUME parameter or blank				
INPUT	CL1	Auto-restore indicator	N	not an auto-restore		
			Y	is an auto-restore		

TABLE 99. Parameters Passed to RESAFTEX

Type	Size	Description		
INPUT	CL1	TSO task indicator	N	not a TSO task
			Y	is a TSO task
INPUT	CL8	Jobname		

RESCRNEX — Screen Restore Requests

RESPRIEX — Screen Restore Requests

The module name supplied for these two exits receive control from RESTORE, Auto-Restore, Dispose, TSO RESTORE, and RECOVER prior to restoring the data set or cluster. The user can request that CA-Disk reject the restore request or continue processing.

The difference between these two exits has to do with the time in which control is passed to the exit:

- **RESCRNEX** — receives control prior to the archive volume being mounted on the tape or cartridge unit
- **RESPRIEX** — receives control after the archive volume is already mounted

Note: User's who install PTF SS02159 MUST linkedit these user exits using AMODE=31.

TABLE 100. Parameters Passed to RESCRNEX and RESPRIEX

Type	Size	Description				
INPUT	XL256	DSNINDEX record for the data set to be restored				
OUTPUT	CL50	Message Field - Place message here when return code is set to 2 to reject request. The message will be printed with CA-Disk output.				
OUTPUT	HW	Specify return code	H’1’	Continue Normal Processing		
			H’2’	Terminate Processing for Restore		
INPUT	CL8	Function issuing restore				
IN/OUT	CL44	New name for data set				
IN/OUT	HW	Count of volumes supplied in VOLUME parameter				
IN/OUT	nCL6	Volume(s) supplied in VOLUME parameter or blank. Current CA-Disk processing limits the volume list to one volume. Do not enter more than one.				
IN/OUT	CL8	Pool name specified on command or blank				
IN/OUT	XL4	Disposition indicators	CL1	Scratch	U	no scratch
					S	scratch
			CL1	Catalog	C	catalog
					N	no catalog
					R	recatalog
			CL1	Create date	N	no reset of create date
					C	reset of create date
			XL1	Erase	x’00’	no erase
					x’01’	erase

TABLE 100. Parameters Passed to RESCRNEX and RESPRIEX

Type	Size	Description		
INPUT	CL1	Auto-restore indicator	N	not an auto-restore
			Y	is an auto-restore
INPUT	CL1	TSO task indicator	N	not a TSO task
			Y	is a TSO task
INPUT	CL8	Jobname		
INPUT	CL8	Password supplied in PASSWORD parameter		

RETAFTEX — Retention Control Data Set Screening (Post-validate)

The module name supplied for this exit will receive control from the CA-Disk retention control function after it has retrieved the format-1 DSCB and qualified it for processing. The disposition of the data set is shown in field (PARM 3).

The user can request that CA-Disk reject the data set for processing or continue on to process the data set as normal. Use sysparm RETAFTEX to specify the module name to be invoked for this exit.

TABLE 101. Parameters Passed to RETAFTEX

Type	Size	Description	
INPUT	XL256	Data Set Identifier - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.	
INPUT	XL140	Volume Identifier - Format-4 DSCB for data set. Volume serial is in positions 1 to 6.	
IN/OUT	XL14	The DISPOSITION field to be used to process the data set. Portions of this field have been taken from the CA-Disk RETVOLSPEC table.	
		1-4	TASC FIELD from RETVOLSPEC table entry
		1	Retention selection criteria that caused the data set to be selected for processing. Modification of this field will not affect any CA-Disk processing or output contents.
		2-4	The ASC values from the RETVOLSPEC table entry causing the data set to be processed. These values can be modified with valid ASC values to change the way in which CA-Disk will process the data set.
		5	RESERVED
		6-8	XL3'yydddd' Current Date (BINARY)
		9-10	RESERVED
		11-14	Archive Reason Codes - The four characters of reason codes that indicate on the RETAIN report why the data set was selected for processing. This field can be modified to reflect different values on the RETAIN report.

RETGDGEX — Return GDG Limit to Implicit Archival for a DSN

This module is given control after implicit archival has failed to obtain a generation limit from the GDGDASDG table for a GDG data set.

The user can supply the GDG limit for the data set. Use sysparm RETGDGEX to specify the module name to be invoked for this exit.

TABLE 102. Parameters Passed to RETGDGEX

Type	Size	Description		
INPUT	CL44	DATA SET NAME		
OUTPUT	HW	GDG LIMIT COUNT to use for the data set		
OUTPUT	HW	Specify Return Code	H'1'	Generation Limit Returned
			H'2'	Use Default Generation Limit

RETPRIEX — Retention Control DSN Screening (Pre-validation)

The module name supplied for this exit will receive control from the CA-Disk retention control function after it has retrieved the format-1 DSCB from disk and prior to qualifying it for processing. The user can request that CA-Disk reject the data set for processing, continue on to qualify the data set for processing, and/or change the disposition action CA-Disk is to use for the data set.

Use sysparm RETPRIEX to specify the module name to be invoked for this exit.

TABLE 103. Parameters Passed to RETPRIEX

Type	Size	Description		
INPUT	XL256	Data Set Identifier - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.		
INPUT	XL140	Volume Identifier - Format-4 DSCB for data set. Volume serial is in positions 1 to 6.		
OUTPUT	XL14	DISPOSITION field. If the RETURN CODE is set to 3, positions 2 through 4 must contain valid ASC values to be used in place of the values normally taken from the RETVOLSPEC table.		
		1-4	TASC FIELD (like RETVOLSPEC table)	
		1	T value of TASC field. This value is set to N upon entry to the exit. Modification of this field will not affect any CA-Disk processing or output contents.	
		2-4	This field is initialized to C'NNN' upon entry to the exit. If the exit routine sets the value of parameter 4 to H'3', this field must be filled with valid ASC values to determine how to process the data set.	
		5	RESERVED	
		6-8	XL3'yydddd' Current Date (BINARY)	
		9-10	RESERVED (DO NOT CHANGE)	
		11-14	ARCHIVE REASON CODES - The 4 characters of reason codes that indicate on the RETAIN report why the data set was selected for processing. This field can be modified to reflect different values on the RETAIN report.	
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Terminate Processing For Data Set
			H'3'	Initiate Disposition Per Disposition Field

RLSEAFEX — Idle Space Release Screening Exit

The module name supplied for this exit is given control after a data set has been selected for processing by idle space release.

The user can request that CA-Disk either continue normal processing or bypass the data set. When VSAM data sets are being processed, the entire cluster will be bypassed if this exit determines that either the data or index component was bypassed.

Use sysparm RLSEAFEX to specify the module name to be invoked for this exit; the EXIT= parameter on the command will override this default value.

TABLE 104. Parameters Passed to RLSEAFEX

Type	Size	Description		
INPUT	CL256	Data Set Identifier - Format-1 DSCB and appendage for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.		
INPUT	XL256	Volume Identifier - Format-4 DSCB and appendage for data set to be processed. Turn to "Format-4 Fields" on page 500 for a description of this area. The volume serial is in positions 1-6.		
INPUT	XL1	Simulation Flag	X'01'	simulate mode
			X'00'	live mode
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Bypass Processing for Data Set
INPUT	CL44	VSAM Cluster Name - For non-VSAM data sets, this field is filled with blanks. For VSAM data sets, it contains the cluster name for the component passed in the format-1 DSCB and the format-4 DSCB.		

RPTDSBEX — Process Data Set Billing Report Records

The module name supplied for this exit will receive control from the CA-Disk billing EXTEND processing. It is passed the report record (see "Report Record Format — (RPTNDSNDSB)" on page 410).

When this exit is specified, the standard CA-Disk report record is not written. Use sysparm RPTDSBEX to specify the module name to be invoked for this exit.

TABLE 105. Parameters Passed to RPTDSBEX

Type	Size	Description
INPUT	H'1'	OPTION - always a halfword of 1.
INPUT	CL6	CL6 VOLUME
INPUT	CL44	DATA SET NAME
INPUT	XL159	Report Record - See "Report Record Format — (RPTNDSNDSB)" on page 410.

RPTDSNEX — Screen Data Sets to be Reported On

The module name supplied for this exit will receive control from the CA-Disk reports function prior to generating report records for any data set.

The user can request that CA-Disk reject the data set for processing, or continue on to report on the data set. Use sysparm RPTDSNEX to specify the module name to be invoked for this exit.

TABLE 106. Parameters Passed to RPTDSNEX

Type	Size	Description		
INPUT	XL256	Data Set Identifier - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.		
INPUT	XL140	Volume Identifier - Format-4 DSCB for data set. Volume serial is in positions 1 to 6.		
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Terminate Processing for Data Set

RPTDSUEX — Data Set Utilization Report Record Screening

The module name supplied for this exit will receive control from the Data Set Utilization report generator. It is passed each record generated for the report.

The user can request that CA-Disk reject the record for printing or continue processing as normal. Use sysparm RPTDSUEX to specify the module name to be invoked for this exit.

TABLE 107. Parameters Passed to RPTDSUEX

Type	Size	Description		
INPUT	XL140	Volume Identifier - Format-4 DSCB for data set. Volume serial is in positions 1 to 6.		
INPUT	XL256	Data Set Identifier - Format-1 DSCB and appendage for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.		
IN/OUT	XL	DSU Report Record. Turn to "Report Record Format — (RPTNDSUTIL)" on page 417 for a complete description of this record.		
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Do not Print Line for this Record

RPTMVDEX — Multiple Volume Dictionary Record Screening

The module name supplied for this exit will receive control from the MVD report generator. It is passed each record generated for the report.

The user can request that CA-Disk reject the record for printing or continue processing as normal. Use sysparm RPTMVDEX to specify the module name to be invoked for this exit.

TABLE 108. Parameters Passed to RPTMVDEX

Type	Size	Description
INPUT	XL140	Volume Identifier - Format-4 DSCB for data set. Volume serial is in positions 1 to 6.
INPUT	XL256	Data Set Identifier - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.
IN/OUT	XL	MVD Report Record. Turn to "Report Record Format — (RPTNMVDICT)" on page 429 for a complete description of this record.

TABLE 108. Parameters Passed to RPTMVDEX

Type	Size	Description		
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Do Not Print Line for this Record

SCRTCHEX — Screen CA-Disk Scratch Data Set Request

The module name supplied for this exit will receive control from any CA-Disk function requesting to scratch a user's data set. It is invoked before scratch processing has started.

The user can request that CA-Disk reject the request to scratch the data set, or continue to process it. Use sysparm SCRTCHEX to specify the module name to be invoked for this exit.

TABLE 109. Parameters Passed to SCRTCHEX

Type	Size	Description		
INPUT	CL44	Data Set Name - Name of data set to be scratched.		
INPUT	HW	Volume Count - Count of volumes on which data set to be scratched resides.		
INPUT	nCL6	Volume List - A list of 6-character volume serials that contain the data set to be scratched. Parameter 2 contains a count of the number of volumes in this list.		
OUTPUT	CL100	Message - If desired, the exit routine can place a message here to be printed. The message will be printed if the return code parameter contains a 2 and the message area is non-blank.		
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Reject Request to Scratch Data Set

SECURAEX — Security Validation Exit, Post-CA-Disk Decision

The module name supplied for this exit will receive control whenever CA-Disk security routines are called to validate access to a protected resource. This is the post-processing exit. It will receive control after CA-Disk has invoked the user security system or CA-Disk security module and has made a decision as to whether the data set can be processed by CA-Disk.

The exit routine has a chance to examine the decision made by CA-Disk and, optionally, override the decision. Note that if CA-Disk has rejected the data set for processing and messages are to be written to the user, the messages have already been written at this point. To intercept security before messages have been issued, see the SECURPEX (pre-processing) exit.

The parameters to this exit are exactly the same as to the pre-processing exit (SECURPEX) except that the return code field has been set to reflect the decision CA-Disk has made regarding processing of the data set. See the documentation for SECURPEX for exit module parameter descriptions. The user exit can change the return code to any valid value before returning control to CA-Disk.

Use sysparm SECURAEX to specify the module name to be invoked for this exit.

SECURPEX — Security Validation Exit, Pre-CA-Disk Decision

The module name supplied for this exit will receive control whenever CA-Disk security routines are called to validate access to a protected resource. This is the pre-processing exit. It will receive control before CA-Disk invokes the user security system or CA-Disk security module to make a decision as to whether the data set can be processed by CA-Disk.

The exit routine has a chance to make the decision itself by setting the return code to one of the valid values. If the exit decides to leave the decision to CA-Disk, it can return without changing the return code.

Note: When security checking is being made for a data set that exists in the CA-Disk archives, the volume serial passed will be "*ARCH*". This is not a real DASD volume. It has special meaning to CA-Disk.

Use sysparm SECURPEX to specify the module name to be invoked for this exit.

TABLE 110. Parameters Passed to SECURPEX

Type	Size	Description	
INPUT	HW	Resource Type - The first parameter to this exit indicates the resource type for which security is being checked. It is in a halfword binary count format. The following list indicates the resource types protected and the value corresponding to that type:	
		H'1'	Data Set Name
		H'2'	reserved
		H'3'	DASD Volume Serial
		H'4'	VSAM Catalog Name
		H'5'	CA-Disk Command
		H'6'	VSAM Component (cluster) Name
		H'7'	CA-Disk Subfile Of The Files Data Set
		H'8'	Member Of CA-Disk Parmlib Data Set
Parameters 2 through 4 passed to the exit are always the same regardless of the resource type for which they are being called. They are:			
INPUT	CL8	Access Request - This field contains an character identifier of the type of access being requested for the resource. The numbers following indicate what RESOURCES are passed the ACCESS TYPE. The following is a list of valid ACCESS TYPES:	
		L8'INPUT '	1,3,4,6
		CL8'OUTPUT '	1,4,5,6,7,8
		CL8'ALLOCATE'	1,6
		CL8'SCRATCH '	1,6
		CL8'CATALOG '	1,6
		CL8'RENAMEF '	1,6
		CL8'RENAMET '	1,6
		CL8'IN/OUT '	1,6
INPUT	CL1	Message Indicator - This 1-character field indicates if CA-Disk has been requested to issue a message if the request to process the resource is rejected. Value Y indicates a message is to be issued. Value N indicates no messages are to be issued.	

TABLE 110. Parameters Passed to SECURPEX

Type	Size	Description	
OUTPUT	HW	Return Code - This field indicates whether the request to process is to be accepted or rejected. The value is in halfword binary count format. It is initialized to zero. If not modified, CA-Disk will continue security checking for the resource and make the decision to accept or reject it. If the user exit routine makes the decision, this field is set to reflect it. The valid values for this field are:	
		H'0'	Let CA-Disk make the decision (SECURPEX ONLY)
		H'1'	Let user process resource
		H'8'	Reject request to process

Additional parameters for SECURPEX are Resource Type = 1 (Data Set Name) parameters below:

TABLE 111. Parameters 5 - 8 (DSNAME) for SECURPEX

Type	Size	Description
INPUT	XL256	Data Set Identifier - Format-1 DSCB and CA-Disk appendage for the data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area. **Note**: Reading the directory of a PDS requires security clearance that has not been performed yet. Therefore the PDS directory fields in the appendage area have not been filled at the time of this exit.
INPUT	HW	Volume Count - The number of volumes on which this data set resides.
INPUT	nCL6	List Of Volumes - A list of volumes on which the data set resides. The number of entries in this list depends on the count from parameter 6.
INPUT	CL1	Password Indicator - A 1-character flag whose value indicates if password-protected data sets are to be processed. A Y indicates the user has requested to process password- protected data sets. An N indicates the user has not specifically requested for process password-protected data sets. If the PASSWORD parameter is available on the CA-Disk command, this indicates if it has been supplied or not.

Additional parameters for SECURPEX are Resource Type = 3 (DASD Volume Serial) described below:

TABLE 112. Resource Type = 3 (VOLSER) for SECURPEX

Type	Size	Description
INPUT	CL6	Volume Serial Name - The 1- to 6-character volume serial that the user is requesting to process.

Additional parameters for SECURPEX are Resource Type = 4 (VSAM Catalog Name) described below:

TABLE 113. Resource Type = 4 (VSAM Catalog Name)

Type	Size	Description
INPUT	CL44	VSAM Catalog Name - The 1- to 44-character data set name of the VSAM catalog.
INPUT	CL8	Catalog Password - If the user has supplied the PASSWORD for the catalog in the CA-Disk command, it will be present here.
INPUT	CL6	Catalog Volser - The volume serial name on which the VSAM catalog resides.

Additional parameters for SECURPEX are Resource Type = 5 (CA-Disk Command) described below:

TABLE 114. Resource Type = 5 (Commands) for SECURPEX

Type	Size	Description
INPUT	CL12	CA-Disk COMMAND - The 1- to 12-character name of the CA-Disk command the user is attempting to use. This is the full command name regardless of how it was entered.

Additional parameters for SECURPEX are Resource Type = 6 (VSAM Component) described below:

TABLE 115. Resource Type = 6 (VSAM Component)

Type	Size	Description	
INPUT	CL44	VSAM Component Name - The 1- to 44-character data set name of the VSAM component to be accessed.	
INPUT	CL8	VSAM Component Type - This field indicates the type of component for which the request applies. The valid values are:	CL8'CLUSTER '
			CL8'DATA '
			CL8'INDEX '
			CL8'PATH '
			CL8'AIX '
INPUT	CL8	VSAM Password - The VSAM password specified for the component name.	
INPUT	XL2	VSAM Security Flags - This is the VSAM security flags from the VSAM CATALOG record for the component. See the IBM CATALOG MANAGEMENT PLM for a complete description of these flags.	
INPUT	HW	Volume Count - A count of 1. This value never changes for the exit.	
INPUT	HW	Catalog Volume Serial - The DASD volume serial that contains the VSAM catalog that describes this component.	

Additional parameters for SECURPEX are Resource Type = 7 (CA-Disk Subfile) described below:

TABLE 116. Resource Type = 7 (Subfile) for SECURPEX

Type	Size	Description
INPUT	CL8	Subfile Name - The 8-character name of the subfile in the CA-Disk files data set for which access is being requested.
INPUT	CL8	Member Name - The 8-character name of the member of the parmlib data set for which access is being requested.

SPACECEX — Space Conversion Exit (Non-VSAM)

The module name supplied for this exit will receive control from any CA-Disk function requesting allocation of DASD space for a non-VSAM data set.

To have the blocksize redriven for a new allocation, pass in a blocksize of “0” at the time of RESTORE/RECOVER, MOVE or COPY.

The user can inspect or modify the number of tracks that CA-Disk has chosen to allocate for the data set. Use sysparm SPACECEX to specify the module name to be invoked for this exit.

TABLE 117. Parameters Passed to SPACECEX

Type	Size	Description
IN/OUT	HW	New tracks to allocate for the data set
INPUT	CL44	Data Set name being allocated
INPUT	CL6	Volume on which allocation will occur
INPUT	FW	Original Tracks for the data set on its original device type
INPUT	XL2	Original Source Device Type as expressed in the UCBTYPE field
INPUT	XL2	Target Device Type as expressed in the UCBTYPE field
INPUT	FW	Tracks Per Cylinder on SOURCE device type
INPUT	FW	Tracks Per Cylinder on TARGET device type
INPUT	FW	Original Blocksize
INPUT	FW	New Blocksize
INPUT	XL2	DSORG as defined in the VTOC entry
INPUT	XL1	RECFM as defined in the VTOC entry
INPUT	HW	KEY LENGTH for a key-sequenced data set
INPUT	XL1	SCALO field as defined in the VTOC entry
INPUT	XL1	OPTCD field as defined in the VTOC entry

SPFHDGEX — ISPF Report Heading Exit

The module name supplied for this exit will receive control after CA-Disk has formatted the top two heading lines for an ISPF report, but before it has printed them.

The user can change either or both of the report headings to suit installation standards. If the headings are not modified, CA-Disk will print the headings it has built. Use sysparm SPFHDGEX to specify the module name to be invoked for this exit.

TABLE 118. Parameters Passed to SPFHDGEX

Type	Size	Description
INPUT	CL60	Report title user specified (do NOT modify)
IN/OUT	CL132	Top heading of ISPF report
IN/OUT	CL132	Second heading of ISPF report

SPFORDEX — ISPF Online Reporting Report Definition Authorization

A specific exit can be invoked to send CA-Disk messages through the SAMS subsystem. The module name supplied for this exit will receive control if a user has selected a report definition option in ISPF online reporting.

The exit routine is passed the TSO user ID of the user requesting the option and passes back a return code to indicate if the user is authorized or not. Use sysparm SPFORDEX to specify the module name to be invoked for this exit.

TABLE 119. Parameters Passed to SPFORDEX

Type	Size	Description		
INPUT	CL8	TSO user ID		
OUT	H	Return code	H'1'	user is authorized
			other	user is not authorized

SPFORSEX — ISPF Online Reporting Authorization

The module name supplied for this exit will receive control if a user has chosen to execute an online report.

The exit routine is passed the TSO user ID of the user requesting the report and passes back a return code to indicate if the user is authorized or not. Use sysparm SPFORSEX to specify the module name to be invoked for this exit.

TABLE 120. Parameters Passed to SPFORSEX

Type	Size	Description		
INPUT	CL8	TSO user ID		
OUT	H	Return code	H'1'	user is authorized
			other	user is not authorized

SYSOUTEX — SYSOUT Exit

The module name supplied for this exit will receive control for every line of printed output being directed to CMDPRINT, MSGPRINT or SYSPRINT.

The user can instruct CA-Disk to either print or bypass the line of output. Suppression of heading lines can yield unpredictable results, however, in that line count and paging are affected. You also must not attempt to use any of the standard CA-Disk techniques to issue an additional message from your exit. This recursiveness will be detected, and the message will be written to the operator instead of its designated location. You can, of course, redirect the line of output to any new target, such as a disk data set or a special operator console.

Install the SYSOUTEX using the source code shown in USERMOD.

TABLE 121. Parameters Passed to SYSOUTEX

Type	Size	Description
INPUT	CL8	DDNAME (CMDPRINT, MSGPRINT or SYSPRINT)
INPUT	CLxx	Print Line

TABLE 121. Parameters Passed to SYSOUTEX

Type	Size	Description		
INPUT	CL1	Carriage Control for the print line:	C’ ‘	single space
			C’0’	double space
			C’-’	triple space
			C’1’	new page, first line
INPUT	HW	LENGTH of the PRINT LINE		
INPUT	XL2	TYPE of PRINT LINE:	X’0000’	regular data line for printing
			C’H’-C’H9’1’	a heading line
			C’F1’-C’F9’	a footing line
OUTPUT	HW	Specify The Return Code	H’1’	continue (print the line)
			H’4’	skip the print line

TAOPPOEX — Archive Tape Processing Exit (Post-Open)

The module name supplied for this exit will receive control from CA-Disk functions that process archive tape data. It is invoked immediately after an archive tape volume has been opened.

The user exit allows the user to inspect and/or change the system control blocks related to the open. Use sysparm TAOPPOEX to specify the module name to be invoked for this exit.

TABLE 122. Parameters Passed to TAOPPOEX

Type	Size	Description		
INPUT	CL8	Function - Indicates the CA-Disk function invoking the exit (that is, archive, restore, merge)		
INPUT	CL6	Type Of Open	“input”	indicating read
			“output”	indicating write
INPUT	CL1	Type Of Tape	M	indicates primary archive tape being processed
			C	indicates copy archive tape being processed
INPUT	XL	DCB - Data Control Block used for the open		
INPUT	XL	JFCB - Job File Control Block used for the open		

TAOPPREX — Archive Tape Processing Exit (Pre-Open)

The module name supplied for this exit will receive control from CA-Disk functions that process archive tape data. It is invoked prior to opening an archive tape volume.

The user exit allows the user to inspect and/or change the system control blocks related to the open. Use sysparm TAOPPREX to specify the module name to be invoked for this exit.

TABLE 123. Parameters Passed to TAOPPREX

Type	Size	Description		
INPUT	CL8	Function - Indicates the CA-Disk function invoking the exit (that is, archive, restore, merge)		
INPUT	CL6	Type Of Open	“input”	indicating read
			“output”	indicating write
INPUT	CL1	Type Of Tape	M	indicates primary archive tape being processed
			C	indicates copy archive tape being processed

TABLE 123. Parameters Passed to TAOPPREX

Type	Size	Description
INPUT	XL	DCB - Data Control Block used for the open
INPUT	XL	DCB - Data Control Block to be used when opening the tape volume. It is set up to open when the exit is called

TAPCTLEX — Tape Management System Control Interface Exit

This exit is identical to TMSCTLEX. The module name supplied for this exit enables users to run more than one tape management system during a conversion process. See user exit TMSCTLEX for a full description, and a complete list of input and output parameters available.

TAPEDLEX — Archive Volume Being Deleted from Archives

The module name supplied for this exit will receive control from any CA-Disk function that is deleting TAPE/CARTRIDGE archive volumes from the archive index. This exit is bypassed in SIMULATE mode.

The archive volume record passed to this exit is always sequence #1. Parm 2 indicates the total number of archvols records in the list and Parm 3 is the address of the list of arckeys for this archive data set. (For example, if a multivolume archive data set is being deleted, parm 2 will be GT H'1' and parm 3 will have more than one arckey in the list.)

No return code is accepted by this exit. CA-Disk processing cannot be altered. Use sysparm TAPEDLEX to specify the module name to be invoked for this exit.

TABLE 124. Parameters Passed to TAPEDLEX

Type	Size	Description
INPUT	XL110	Archvols Record - This is the archive volume index record from the files data set. ARCKEY is in positions 1 - 6. The true volser is in position 12 - 17. Turn to "Subfile Record Format — (ARCHVOLS)" on page 459 for a complete description of this record.
INPUT	H	List Count - Number of archive keys in the list that follows (maximum of 35 for any given multivolume archive data set).
INPUT	XL4	List Address - The address of the list of archive keys for the specified archive data set. (The number of keys in the list is specified in parm 2).

TMSCTLEX — Tape Management System Control Interface Exit

The module name supplied for this exit receives control from any CA-Disk function that is updating the status of an existing archive tape volume. This exit can then make the appropriate updates in whatever tape management data base is used to control the tapes. When CA-Disk changes the status of a tape volume, only the files data set is updated. This exit must provide the interface to the tape management system.

Note: User Exit TAPCTLEX enables support for sites that use more than one tape management system.

There are two options that correspond to the two reasons for an update:

1. CA-Disk no longer needs the tape. The tape can be returned to scratch status within the tape management system data base. This option is invoked when CA-Disk is going to expire or delete an archive volume.

2. The expiration date for a tape must be updated. This option is invoked when CA-Disk is going to alter the expiration date of an archive volume. When option 2 is invoked, a seventh parameter is passed. This parameter is the new expiration date that CA-Disk is going to reset for the archive volume. This parameter does not exist for option 1.

No messages are issued based on the return code passed back from this exit. The exit must provide these messages. The exit must also ensure the integrity of the tape management system data base and ensure that tapes do not prematurely expire.

Use sysparm TMSCTLEX to specify the module name to be invoked for this exit.

TABLE 125. Parameters Passed to TMSCTLEX

Type	Size	Description		
INPUT	H	OPTION:	1	Expire Tape
			2	Reset Expiration Date
INPUT	CL44	Tape DSN		
INPUT	H	Number of tape volumes Will always be 1		
INPUT	CL6	Tape volume		
OUTPUT	H	Return Code:	1	Update successful
			2 or more	update failed
OUTPUT	H	Reason Code:	Not used at this time. Can be set to any HW value.	
INPUT	PL4	New EXPDT - (option 2 only)		

Although you can choose to write this exit yourself, these exit programs are supplied as part of the CA-Disk system.

1. ADSTH014 — the interface to the CA1 External Data Manager (EDM) which is supported in Release 4.8 or higher of CA1. Using this program is the preferred method of interfacing with CA1. Implementation instructions can be found in the topic "Method 1 — Controlling Tapes Via the EDM" on page 86 in the Installation Guide.
2. ADSTH015 — to support the EDM interface developed for EPIC.
3. ADSTH016 — to support the EDM interface developed for RMM.
4. ADSTH017 — to support the EDM interface developed for release 5.0.4 of TLMS. This is documented in "ADSTH017 — Interface for TLMS" on page 252.
5. ADSTH025 — a direct interface for Release 4.7 and below of CA1. This is documented below.

Some tape management systems vendors have developed their own EDM interface to our software, such as Auto Media, formerly known as Zara. If your tape management system is not listed here, contact the tape management system vendor to discover if an EDM exit is available.

ADSTH025 — Direct Interface for CA1

For expiration date control within a CA1 (release 4.7 and below) environment, we recommend that you install the direct interface that CA-Disk provides to CA1. Review the information presented below, as well as the topic "Method 2 — Controlling Tapes by Expiration Date" on page 88 in the Installation Guide, and then decide if the direct interface to CA1 is applicable and desirable in your installation. If it is not, you should change the expiration date

being assigned via sysparm DYNEXPDT (turn to "DYNEXPDTE99365" on page 143 for a description of this sysparm) or your JCL to specify either a true expiration date or the value 99000, which can then be used to place the tapes under catalog control. Also see "Assigning Tape Expiration Dates" on page 45 for other applicable rules.

Note: If CA1 release 5.0 or above is installed, do not use the CA-Disk/CA1 direct interface. CA1 macro changes have made this interface incompatible.

The CA-Disk direct interface to CA1 uses a central module designed to handle the interchange and update of records in the TMC (Tape Management Catalog). All precautions are taken to ensure that updating is thorough and complete. However, if you use a CA1 release that is higher than 4.7 and specify sysparm DYNEXPDT to a value other than the default, you should be aware of the following limitations when using this interface:

1. Like CA1 Release 4.7 and below, this interface does not support expiration dates beyond the year 2000.
2. This interface does no special processing for expiration dates which fall within the ranges of cycle or frequency control.
3. This interface does not read or write exception or journal records.

When CA-Disk determines that an archive or backup tape can be returned to scratch status, it can inform CA1 in two different ways:

1. CA-Disk can retrieve the TMC record for that tape and set its expiration date to the current date. The tape is made a scratch when the CA1 scratch and clean jobs are run.
2. CA-Disk can take the same action as above, but also turn on the scratch bit indicator flag. Turning on this bit makes the tape an immediate scratch. This is particularly useful when also using the CA-Disk tape pool support.

Use of the direct interface to CA1 is not required. It simply provides the advantage of immediate updates to the CA1 tape management catalog when CA-Disk releases a tape volume from the archives or increases the expiration date of an archived data set. It has no effect during data set archival/backup processing since standard CA1 hooks in tape OPEN get the expiration date from system control blocks at that time.

Installing the CA-Disk Direct Interface to CA1:

1. Assemble module ADSTH012. The module is located in the installation library, and is written in Assembler Language. Use a standard assembly procedure, ensuring that the CA-Disk macro library is concatenated before all other maclibs in the SYSLIB statements. Also make sure that the CA1 macro library is in the SYSLIB statements but after the CA-Disk macro library.

   If you have installed CA-Disk with SMP/E, this assembly should be done using SMP/E in the form of a usermod. Installation library member USERMODA can accomplish this for you. If you did not install CA-Disk with SMP/E, sample assemble/link JCL for assembling this module is located in installation library member INSTTMS.

   We recommend using Assembler H to assemble CA-Disk modules. However, we will try to allow source members in the installation library to be assembled with Assembler F. If you use Assembler F, you must add

   ```
   ,AMODE=,SPLEVEL=
   ```

to the end of the PROC macro statement in your assemble source. If Assembler F cleanly assembles a CA-Disk module (condition code 0—no assembly errors) the module is correctly assembled. The problems with Assembler F are in the form of assembly errors (for example, “IF STATEMENT OUT OF ORDER. . .“) not experienced with Assembler H.

Note: Since the CA1 macros can change with each release of CA1, you can need to assemble and link the CA-Disk module ADSTH012 when you install a new release of CA1.

If a problem arises because some of the needed CA1 modules cannot be found, simply put a PRINT GEN statement in module ADSTH012 and reassemble it. The assembly listing with the expanded macros will identify the CA1 modules that must be found to complete a successful installation.

2. Verify that the module is in the CA-Disk load library and there are no unresolved references in the link-edit.

Activating the CA-Disk **Direct Interface to CA1:**

1. Make the CA1 load library available to CA-Disk. Either put the CA1 load modules in a LINKLIST library, or put the CA1 load library in the STEPLIB for all CA-Disk backup, archive, restore and IXMAINT tasks.
2. Review the following sysparms in the Sysparms section:

   TPMGMTEX — on page 184
   TMSCTLEX — on page 184
   UCC1SCRT — on page 186
   UCC1EXPR — on page 186
   UCC1BYPS — on page 186
   UCC1EXTN — on page 186

3. Turn on the CA1 support by specifying sysparm TMSCTLEX with a value of ADSTH025.
4. Decide which of the two methods of scratching tapes CA-Disk is to use. CA-Disk defaults to just setting the expiration date to the current date. If you want CA-Disk to also set the scratch bit, causing CA1 to immediately treat it as a scratch tape, specify sysparm UCC1SCRT with a value of Y.

CA-Disk/**CA1 Interfacing Considerations -** We recommend that you run the DSNDELETE command of IXMAINT (turn to “DSNDELETE Command and Parameters” on page 322 in the User’s Guide for details) nightly before the CA1 scratch and clean functions are run. This helps to ensure that CA1 is always current with CA-Disk.

If a job abends while writing a tape and you are not using the EDM Interface as your CA-Disk interface, CA1 defaults to considering that output tape a scratch. To CA-Disk, however, the partial tape is a good tape and must be kept. You should take special precautions to prevent CA1 from marking CA-Disk output tapes as scratch tapes after an abend. This can be done manually, or you can make use of one of the exits within the CA1 system. (A CA-Disk user has supplied a sample for this exit in member SLI035 of the user mod library. You should consult your local CA1 support staff or Computer Associates directly, if necessary, if you have further questions regarding their exit.)

ADSTH017 — Interface for TLMS - This module provides support for the EDM interface developed for release 5.0.4 of TLMS. To enable the support, sysparms TMSCTLEX and DYNEXPDT must be set to the values

ADSTH017 and E99365 respectively. The TLMS documentation supplied by CA must also be performed. Known requirements from CA for this support INCLUDE (but cannot be limited to) the following:

1. The TLMSIPO member in the TLMS PPOPTION library needs to be updated with the following parameters:

Parameters	Explanation
`EDM=DISK`	This is the association to the tape owner
`DSN=SAMSDISK.TAPE.NAME`	This is the association to the tape dsname specified in sysparm ARC0NAME, ARCCNAME, MERP1NAM, etc.
`JOBNAME=SAMDISK1`	This is the optional jobname that will determine the eligibility to access the EDM
`PGM=ADSMI002`	According to CA, this is the “scratching program name, and IXMAINT, IXUPDATE, and MERGE all run under this program name
`POOLID=DISKPOOL`	This is an optional poolname to restrict this EDM to a TLMS range or pool of tapes

2. To update existing CA-Disk tapes so that they fall under TLMS' control, use the UPV batch command as documented in the TLMS manual. An example of release 5.0.4 is:

   ```
   UPV NNNNNN,OWNER=DISK
   ```

   where NNNNNN is the tape number being updated to EDM control and Disk is the value specified as the EDM ID in the TLMSIPO member)

3. Replace the IBM ARCTVEXT exit with the TLMS ARCTVEXT exit.

4. Set the INPUT=YES subparameter CHAIN=NO in the TLMSIPO member.

Note: This information is provided for your convenience. If there are any questions implementing CA-TLMS parameters, customers should review their TLMS manual or contact CA support directly.

TPMGMTEX — Tape Management Interface Exit

The module name supplied for this exit will receive control from any CA-Disk function that is updating the status of an existing archive volume. Typical types of updates are if CA-Disk is going to expire or delete archive volumes or if CA-Disk is going to update the expiration date of an archive volume.

There are two options possible for this exit. Option 1 is invoked when CA-Disk is going to expire or delete an archive volume. Option 2 is invoked when CA-Disk is going to alter the expiration date of an archive volume. When option 2 is invoked, a seventh parameter is given. This parameter is the new expiration date that CA-Disk is going to reset for the archive volume. This parameter does not exist for option 1.

If you have sysparm TMSCTLEX specified with “ADSTH014”, (that is you are using the External Data Manager facility of CA1 as your tape management interface), you should be aware that there is no facility available to change the expiration date within the CA1 database, therefore no processing will be done by CA-Disk. This user exit will still be called with this option, however, to support other tape management interfaces that can exist.

Use sysparm TPMGMTEX to specify the module name to be invoked for this exit.

TABLE 126. Parameters Passed to TPMGMTEX

Type	Size	Description		
INPUT	H	OPTION:	1	Expire Tape
			2	Reset Expiration Date
INPUT	CL44	Tape DSN		

TABLE 126. Parameters Passed to TPMGMTEX

Type	Size	Description		
INPUT	H	Number of tape volumes. Will always be 1.		
INPUT	CL6	Tape volume		
OUTPUT	H	Return Code	1	OK to update
			2 or more	stop update
OUTPUT	H	Reason Code	Not used at this time. Can be set to any HW value.	
IN/OUT	PL4	New EXPDT - (option 2 only).		

TSCHKAEX — CA-TOP Secret Post-RACHECK Exit

The module name supplied for this exit will receive control from the CA-Disk module responsible for RACHECK processing. It receives control after the RACHECK macro has been issued.

The exit can inspect the return code from the RACHECK macro and/or change it. Use sysparm TSCHKAEX to specify the module name to be invoked for this exit.

TABLE 127. Parameters Passed to TSCHKAEX

Type	Size	Description		
INPUT	CL44	Data Set Name Checked		
INPUT	HW	Volume Count - A count of the number of volumes on which the data set resides. This count indicates how many volumes are contained in the data for parameter 3.		
INPUT	nCL6	List Of Volumes On Which Data Set Resides		
INPUT	CL1	Requested Access Level	'U'	Update access requested
			'R'	Read access requested
INPUT	CL1	Data Set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set
IN/OUT	HW	The return code from the RACHECK macro is placed in this area. If desired it can be changed to one of the following valid values.		
		H'0'	OK to Process Data Set	
		H'4'	Resource not Defined	
		H'8'	User is not Authorized	
		H'16'	Critical Error	

TSCHKPEX — CA-TOP Secret Pre-REACHECK Exit

The module name supplied for this exit will receive control from the CA-Disk module responsible for RACHECK processing. It receives control before the RACHECK macro is issued.

The exit can change RACHECK parameters, or cause the RACHECK to be bypassed and return its own return code. Use sysparm TSCHKPEX to specify the module name to be invoked for this exit.

TABLE 128. Parameters Passed to TSCHKPEX

Type	Size	Description		
INPUT	CL44	Data Set Name To Be Checked		
INPUT	HW	Volume Count - A count of the number of volumes on which the data set resides. This count indicates how many volumes are contained in the data for parameter 3.		
INPUT	nCL6	List Of Volumes On Which Data Set Resides		
INPUT	CL1	Requested Access Level	'U'	Update access requested
			'R'	Read access requested
INPUT	CL1	Data Set Type	'V'	Request for VSAM cluster
			'N'	Request for non-VSAM data set
OUTPUT	HW	If the exit desires to perform the CA-Top Secret check and to have CA-Disk bypass the RACHECK, place a return code in this parameter. To cause CA-Disk to continue processing, leave this parameter with the value as it was on entry.		
		H'0'	OK to Process Data Set	
		H'4'	Resource not Defined	
		H'8'	User is not Authorized	
		H'16'	Critical Error	

UNLOADEX — Process Unloaded Files Data Set Record

The module name supplied for this exit will receive control after a record has been selected for unload in the files unload process. The entire record is passed to the module and the user can modify any field desired. If the length of any record is altered, the user must ensure that the FILEDEFN table and files data set entries conform to the change. The exit can also be used to selectively eliminate records by setting a return code of 2.

Note: When record length is referenced, it means the file-defined record length plus decimal 12; for example, if a DSNINDEX record is passed and its file-defined length is 256, the length passed is 268.

TABLE 129. Parameters Passed to UNLOADEX

Type	Size	Description		
IN/OUT	CL8	Subfile Name		
IN/OUT	CL256	Files Record (maximum length of 256 bytes)		
IN/OUT	HW	Record Length (can be modified)		
OUTPUT	HW	Return Code	H'1'	Unload the record
			H'2'	Bypass the record

USERSENQ — ENQ/DEQ Exit

The module name supplied for this exit will receive control for each ENQ, DEQ, RESERVE or RELEASE to be issued. Its intended use is to allow the user to monitor, modify or replace the normal CA-Disk ENQ processing. It does not affect processing specified by the CA-Disk sysparms SPFENQQN, SPFENQTY or SPFENQWT. (To perform its intended task, the CA-Disk auto-restore function internally assigns and uses the module name "DMSNOENQ" for this user exit.)

A return code set by the exit module indicates what further action CA-Disk is to take. Use sysparm USERSENQ to specify the module name to be invoked for this exit.

TABLE 130. Parameters Passed to USERSENQ

Type	Size	Description		
IN/OUT	CL2	Type Of Enq Request	CL2'E '	ENQ Exclusive
			CL2'S '	ENQ Shared
			CL2'C '	Change Shared to Exclusive
			CL2'D '	DEQ
			CL2'RE'	RESERVE and ENQ Exclusive
			CL2'RS'	RESERVE and ENQ Shared
			CL2'RC'	RESERVE and Change Shared to Exclusive
			CL2'RD'	RELEASE and DEQ
			CL2'TE'	Tape ENQ
			CL2'TD'	Tape DEQ
INPUT	CL44	DSNAME (the resource name)		
INPUT	CL6	VOLUME (for RESERVE requests)		
OUTPUT	H	Return Code	H'0'	Skip normal ENQ processing. DO NOT perform ENQ/DEQ processing.
			H'4'	Continue with normal CA-Disk ENQ/DEQ processing per the request type in parameter 1
			H'5'	Reject the ENQ/DEQ request

USERVENQ — VSAM DISP=SHR/OLD Exit

The module name supplied for this exit will receive control for each VSAM cluster to be allocated in a task. Its intended use is to allow the user to monitor or replace the DISP=SHR or DISP=OLD value.

A return code set by the exit module indicates what further action CA-Disk is to take. The USERVENQ user exit is not called for empty VSAM clusters. CA-Disk obtains all pertinent information from the catalog thereby removing the necessity of an enqueue.

Use sysparm USERVENQ to specify the module name to be invoked for this exit.

TABLE 131. Parameters Passed to USERVENQ

Type	Size	Description		
IN/OUT	CL2	Type Of Enq Request	CL2'E '	ENQ Exclusive (DISP=OLD)
			CL2'S '	ENQ Shared (DISP=SHR)
INPUT	CL44	DSNAME (the resource name)		
INPUT	CL6	VOLUME		

TABLE 131. Parameters Passed to USERVENQ

Type	Size	Description		
OUTPUT	H	Return Code Value	H'0'	Continue with normal CA-Disk processing. Use the ENQ type returned in the first parameter.
			H'5'	Skip normal CA-Disk processing. Bypass processing of the cluster.

USRCODEX — User Code Substitution Exit

This module is invoked by DASD billing just prior to a table lookup to obtain a user code for a given data set name. The data set name is in a work area and can be modified by the exit module. If the data set name is modified, it will be used to search the USERCODTBL.

Use sysparm USRCODEX to specify the module name to be invoked for this exit.

TABLE 132. Parameters Passed to USRCODEX

Type	Size	Description		
INPUT	CL44	DSNAME For Data Set		
INPUT	CL12	Usercode Value		
OUTPUT	HW	Specify Return Code	H'1'	Obtain Code From Table and Continue
			H'2'	Use User Code Returned by Exit Module

VCCOPYEX — Move/Copy Screening

This module is invoked just prior to the copying a data set. The exit is the equivalent of the VCDMIGEX exit except it is invoked for the COPY command instead of the MIGRATE command. The exit can override the migration attempt.

Use sysparm VCCOPYEX to specify the module name to be invoked for this exit.

TABLE 133. Parameters Passed to VCCOPYEX

Type	Size	Description
INPUT	XL256	Data Set Identifier - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.
INPUT	XL140	Volume Identifier - Format-4 DSCB for data set. Volume serial is in positions 1 - 6.

TABLE 133. Parameters Passed to VCCOPYEX

Type	Size	Description		
INPUT	XL16	Option Indicators - 16 one-character fields that indicate the options active at the time the exit is invoked. If the byte contains X'00', the option is not active. If it contains X'01', the option is active.	XL1	SIMULATE
			XL1	NOCATALOG
			XL1	NOSCRATCH
			XL1	FILL
			XL1	REDUCE
			XL1	PREALLOC
			XL1	CYLINDER
			XL1	PASSWORD
			XL1	ABSOLUTE
			XL1	TRACK
			XL1	CATALOGALL
			XL1	CONTIG
			XL1	CREATE
			XL1	reserved for future expansion
			XL1	reserved for future expansion
			XL1	reserved for future expansion
		The VCFLAGS macro is available in the CA-Disk macro library to map these flags		
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Terminate Processing For Data Set

VCDMIGEX — Move/Copy Screening

This module is invoked just prior to the relocation of a data set. All selection screening by CA-Disk has been done and the exit is given a chance to override the migration.

Use sysparm VCDMIGEX to specify the module name to be invoked for this exit.

TABLE 134. Parameters Passed to VCDMIGEX

Type	Size	Description
INPUT	XL256	Data Set Identifier - Format-1 DSCB, and appendage, for data set to be processed. Turn to "Format-1 Fields" on page 499 for a description of this area.
INPUT	XL140	Volume Identifier - Format-4 DSCB for data set. Volume serial is in positions 1 - 6.

TABLE 134. Parameters Passed to VCDMIGEX

Type	Size	Description		
INPUT	XL16	Option Indicators - 16 one-character fields that indicate the options active at the time the exit is invoked. If the byte contains X'00', the option is not active. If it contains X'01', the option is active.	XL1	SIMULATE
			XL1	NOCATALOG
			XL1	NOSCRATCH
			XL1	FILL
			XL1	REDUCE
			XL1	PREALLOC
			XL1	CYLINDER
			XL1	PASSWORD
			XL1	ABSOLUTE
			XL1	TRACK
			XL1	CATALOGALL
			XL1	CONTIG
			XL1	CREATE
			XL1	reserved for future expansion
			XL1	reserved for future expansion
			XL1	reserved for future expansion
		The VCFLAGS macro is available in the CA-Disk macro library to map these flags		
OUTPUT	HW	Specify Return Code	H'1'	Continue Normal Processing
			H'2'	Terminate Processing For Data Set

VSACESEX — Change VSAM Read/Write Processing Options

The module name supplied for this exit will receive control prior to every open of a VSAM data set. The purpose of the exit is to allow you to change the method CA-Disk will use to read VSAM clusters. It can also specify whether the format of the archived cluster should be in logical records or in control interval image copy format (ESDS clusters only). The values passed to the exit are those that CA-Disk has selected, based on sysparms in effect. If not modified, the values passed will be used when CA-Disk processes the cluster.

This exit would be used typically when your installation has either a systems or application program that formats VSAM entry-sequenced clusters (ESDS) in a non-standard format. Examples of this would be DB2 data bases and some data sets created by the MSA product line. These data sets must be archived in a control interval image copy format, so that they can be restored in the same manner. If you can recognize these data sets based upon data set naming conventions, this exit can be used to select image format for their archival, while all other data sets would use logical record format. The only advantage of using logical record format is that the control interval size of the data set can be changed when restoring the data set, whereas a control interval image copy must be restored to the same control interval size. If this flexibility is not required, you can specify sysparm VSARCFMT with a value of C to force all ESDS clusters to be archived in image copy format, and this exit would not be required.

Note: CA-Disk will automatically attempt to detect DB2 data bases by looking for the characters "DSNDBC" in the second-level node of the data set name. If it finds these characters, it will automatically change its archive format to control interval image copy. If all of your DB2 data bases follow this naming standard, this exit would not be required.

Use sysparm VSACESEX to specify the module name to be invoked for this exit.

TABLE 135. Parameters Passed to VSACESEX

Type	Size	Description		
INPUT	CL44	Data Set Name		
INPUT	CL1	Cluster Type	'K'	Key-sequenced data set
			'E'	Entry-sequenced data set
			'R'	Relative record data set
IN/OUT	CL1	VSAM Read Access Type	'C'	Read the cluster with control interval processing
			'L'	Read the cluster with logical record processing
IN/OUT	CL1	Archive Format Of Data	'L'	Logical record format
			'C'	Control interval image copy format
		Notes:	Format type C is honored only for ESDS clusters.	
			If you specify format type C but specify access type L (parameter 3), CA-Disk will change the access type to C.	
OUTPUT	HW	Result Code	0	Process the data set
			Non-zero	bypass the data set

VSALLOEX — Change VSAM Space Allocation Amounts

The module name supplied for this exit will receive control prior to all defines of VSAM components. The purpose of the exit is to allow the user to inspect the allocation amounts to be used in the DEFINE and change these values if desired. The exit not only receives the original allocation amounts used for the DEFINE, but also the new allocation amount that CA-Disk has calculated. It will be invoked for each component of the cluster.

The exit can either change the new allocation amount calculated by CA-Disk, or leave it alone, in which case that value will be used as the allocation amount. There is no RETURN CODE from this exit.

Note: This exit will not receive control if the allocation amounts for the cluster are specified on the RESTORE command. Its also not available when using the DFSMSdss Data Mover interface.

Use sysparm VSALLOEX to specify the module name to be invoked for this exit.

TABLE 136. Parameters Passed to VSALLOEX

Type	Size	Description	
INPUT	XL7	ORIGINAL ALLOCATION AMOUNT	
		XL3	Primary Allocation Amount
		XL3	Secondary Allocation Amount
		X	Allocation Type
		`1... ....`	Allocation Is In Tracks
		`11.. ....`	Allocation Is In Cylinders

TABLE 136. Parameters Passed to VSALLOEX

Type	Size	Description		
IN/OUT	XL7	New Allocation Amount (calculated By CA-Disk)		
		XL3	Primary Allocation Amount	
		XL3	Secondary Allocation Amount	
		X	Allocation Type	
		`1... ....`	Allocation Is In Tracks	
		`11.. ....`	Allocation Is In Cylinders	
INPUT	F	Total Of All Allocated Extents In Original Data Set (total Is Always In Tracks)		
INPUT	F	Total Amount Of Used Space (total Number Of Tracks Actually Required By Original Data Set)		
INPUT	XL4	Original Device Type		
INPUT	XL4	New Device Type		
INPUT	F	High Used RBA Of Original Data Set		
INPUT	F	High Allocated RBA Of Original Data Set		
INPUT	F	Physical Block Size		
INPUT	F	Control Interval Size		
INPUT	CL1	Component Type	'D'	Data Component
			'I'	Index Component
INPUT	CL44	Data Set Name		
INPUT	H	Number Of Old Volumes		
INPUT	20CL6	List Of Old Volumes		
INPUT	H	Number Of New Volumes		
INPUT	20CL6	List Of New Volumes		

VSATTREX — VSAM Bufferspace and CI Size Exit

The module name supplied for this exit will receive control for each VSAM cluster to be defined in a task. Its intended use is to allow the user to monitor or replace the BUFFERSPACE, CISZ, and certain data set attributes prior to any DEFINE issued by CA-Disk.

Note: Not available when using the DFSMSdss Data Mover interface.

Use sysparm VSATTREX to specify the module name to be invoked for this exit.

TABLE 137. Parameters Passed to VSATTREX

Type	Size	Description		
INPUT	CL44	Data Set Name		
INPUT	CL1	Component Type	CL1'D'	Data Component
			CL1'I'	Index Component
IN/OUT	F	Buffer Space Amount		
		Note: Value can be changed to specific amount on output, or can be set to zero to let AMS calculate new value.		

TABLE 137. Parameters Passed to VSATTREX

Type	Size	Description	
IN/OUT	F	Control Interval Size	
		Note: Value can be changed to specific amount on output, or can be set to zero to let AMS calculate new value.	
IN/OUT	XL2	Data Set Attribute Bytes	
		XL1	Data Set Attribute Byte (ATTR1)
		`1... ....`	Use Speed Option When Loading Data Set
		`0... ....`	Use Recover Option When Loading Data Set
		`.1.. ....`	Define The Data Set As Unique
		`.0.. ....`	Define The Data Set As Suballocated
		`..1. ....`	Define The Cluster As Reusable
		`..0. ....`	Define The Cluster As Not Reusable
		`...X XXXX`	User Exit Should Not Modify These Flags
		XL1	Data Set Sharing Attributes (ATTR2)
		`00.. ....`	Data Set can be shared by read users or by one update/output user
		`01.. ....`	Data Set can be shared by read users and by one update/output user
		`10.. ....`	Data Set Can Be Fully Shared
		`11.. ....`	Data Set Can Be Fully Shared With VSAM's Assistance
		Data Set Sharing Attributes Across Systems, As follows:	
		`..00 ....`	RESERVED
		`..01 ....`	RESERVED
		`..10 ....`	Data Set Can Be Fully Shared
		`..11 ....`	Data Set Can Be Fully Shared With VSAM's Assistance
		`.... XXXX`	User Exit Should Not Modify These Fields

VSMAFTEX — Implicit VSAM Data Set Screening (Post-Validate)

The module name supplied for this exit will receive control from the implicit VSAM support after it has retrieved all information from catalog management regarding the cluster, and also after all screening tests are applied. If the VSMPRIEX user exit is also specified, this exit gets control after the VSMPRIEX exit has screened the request, regardless of the result returned by that exit. The result code passed as input represents the action that CA-Disk will take. This exit will also get control when processing requests that scan volumes using the SCAN command. As of Release 8.0, this is limited to only the ISPF reporting function. When a data set name is supplied as a result of a SCAN REAL command, the catalog name passed as the fourth parameter will be nulls.

The user can request that CA-Disk reject or process the request, regardless of what value was found in the result code parameter. Use sysparm VSMAFTEX to specify the module name to be invoked for this exit.

TABLE 138. Parameters Passed to VSMAFTEX

Type	Size	Description	
IN/OUT	HW	Action Code. This parameter is both an input and an output parameter. The value on input informs the exit what action CA-Disk will take if unmodified by the exit. The value on output (if changed) instructs CA-Disk to take the new action.	
		H'0'	Data Set will be processed
		H'8'	Data Set will be bypassed

TABLE 138. Parameters Passed to VSMAFTEX

Type	Size	Description		
INPUT	CL8	Function Code	'REPORT'	report on the component
			'DELETE'	delete the component
			'BACKUP'	backup the component
			'ARCHIVE'	archive the component
INPUT	CL44	Data Set Name		
INPUT	CL6	Volser - primary volume of data component		
INPUT	CL44	Catalog Name - the catalog name in which the component is defined (if catalog scan)		
INPUT	CL1	DSNTYPE	'C'	Cluster component
			'I'	Index component
			'D'	Data component
			' '	Component type unknown
INPUT	CL6	Volser - volume serial number of this component		

VSMPRIEX — Implicit Data Set Screening (Pre-Validate)

The module name supplied for this exit will receive control from the implicit VSAM support after it has retrieved all information from catalog management regarding the cluster, but before any screening tests are applied. If the VSMAFTEX user exit is also specified, it will get control after the VSMPRIEX exit has screened the request, regardless of the result returned by this exit.

This exit will also get control when processing requests that scan volumes using the SCAN command. This is limited to only the ISPF reporting function. When a data set name is supplied as a result of a SCAN REAL command, the catalog name passed as the fourth parameter will be nulls.

The user can request that CA-Disk continue with normal processing, reject the request, or unconditionally process the request. Use sysparm VSMPRIEX to specify the module name to be invoked for this exit.

TABLE 139. Parameters Passed to VSMPRIEX

Type	Size	Description		
OUTPUT	HW	Result Code. This parameter instructs CA-Disk as to what action to take. Its value on input is always 1, so if this module does not modify its value, normal processing will be carried out.		
		H'1'	Continue normal processing. This means that if any SELECT and/or EXCLUDE commands were specified, those tests will be applied after this exit returns control to CA-Disk.	
		H'2'	Terminate processing for the data set. Note that if the VSMAFTEX is also specified, that exit will still get control and can reverse the rejection request. Any SELECT and/or EXCLUDE commands are ignored.	
		H'3'	Unconditionally select the data set for processing. Note that if VSMAFTEX is also specified, that exit will still get control and can reverse the selection request.Any SELECT and/or EXCLUDE commands will be ignored.	
INPUT	CL8	Function Code	'REPORT'	report on the component
			'DELETE'	delete the component
			'BACKUP'	backup the component
			'ARCHIVE'	archive the component
INPUT	CL44	Data Set Name		

TABLE 139. Parameters Passed to VSMPRIEX

Type	Size	Description		
INPUT	CL6	Volser - primary volume of data component		
INPUT	CL44	Catalog Name - the catalog name in which the component is defined (if catalog scan)		
INPUT	CL1	DSNTYPE	'C'	Cluster component
			'I'	Index component
			'D'	Data component
			' '	Component type unknown
INPUT	CL6	Volser - volume serial number of this components		

VSMSPWEX — VSAM Master Password Exit

The module name supplied for this exit will receive control after the CA-Disk master password is extracted. Its intended use is to allow the user to monitor or replace the CA-Disk master password with a user-supplied password.

Use sysparm VSMSPWEX to specify the module name to be invoked for this exit.

TABLE 140. Parameters Passed to VSMSPWEX

Type	Size	Description
INPUT	CL8	Function Being Executed
IN/OUT	CL8	CA-Disk Master Password

VSNEWNEX — VSAM New Name Exit

The module name supplied for this exit will receive control prior to every DEFINE of a VSAM component issued by CA-Disk. It is invoked whether or not a NEWNAME parameter was specified on the input command. Its intended use is to allow the user to monitor or replace the data set name for each component.

Use sysparm VSNEWNEX to specify the module name to be invoked for this exit.

TABLE 141. Parameters Passed to VSNEWNEX

Type	Size	Description		
INPUT	CL1	Component Type	C'C'	Cluster Component
			C'D'	Data Component
			C'I'	Index Component
INPUT	CL44	Base Component's Original Data Set Name		
INPUT	CL44	Base Component's New Name		
INPUT	CL44	Current Component's Data Set Name		
IN/OUT	CL44	New Data Set Name For Component		
IN/OUT	H	Return Code. Note that this is both an input and output parameter. If the value on input is greater that one, CA-Disk will let AMS generate a data set name for the component. If the value is less than or equal to one, the name passed as input will be used in the DEFINE. This value can be reset by the exit to either let AMS generate a name, or to indicate that the exit is supplying a name.		
		'1'	Use data set name supplied	
		'4'	Let AMS generate a component name	

XCPARCEX — XCOPY Archive Volume Selection Exit

This module is called during XCOPY processing to allow selection processing based on criteria other than that available through SELECT and XCOPY parameters.

This exit is called during volume level selection for all archive volumes in the input files data set. The user can use any and all data in the ARCHVOLS record provided to determine if the volume should be selected for XCOPY processing. The ARCHVOLS record passed to the user exit is a copy of the ARCHVOLS record used during the process and is discarded on return from the user exit. The Flag byte passed to the exit can contain a 'Y' or 'N' depending on whether the volume has been selected for processing via the selection criteria specified on the SELECT and XCOPY command lines. In this way the exit can know if the volume was to be selected and override processing if desired.

TABLE 142. Parameters Passed to XCPARCEX

Type	Size	Description	
INPUT	XL110	Input ARCHVOLS Record	
IN/OUT	CL1	Flag, C 'Y'	Select volume
		Flag, C 'N'	Do not select volume

XCPDSNEX — XCOPY Data Set Level Selection Exit

This module is called during XCOPY processing to allow selection processing based on criteria other than that available through SELECT and XCOPY parameters.

This exit is called for all data sets that exist on the Archive Volumes that have previously been selected for processing. The user can use any and all data in the ARCHVOLS and DSNINDEX records provided, to determine if the data set should be selected for XCOPY processing. The ARCHVOLS and DSNINDEX records passed to the user exit are copies of records used during the process and are discarded on return from the user exit. The Flag byte passed to the exit can contain a 'Y' or 'N' depending on whether the data set has been selected for processing via the selection criteria specified on the SELECT and XCOPY command lines. In this way the exit can know if the volume was to be selected and override processing if desired.

TABLE 143. Parameters Passed to XCPDSNEX

Type	Size	Description	
INPUT	XL110	Input ARCHVOLS Record	
INPUT	XL256	Input DSNINDEX Record	
IN/OUT	CL1	Flag, C 'Y'	Select data set
		Flag, C 'N'	Do not select data set

CHAPTER 6

Message Formatting

Module ADSUT028 is available to allow user-written programs (for example, a user exit) to write messages to the same //MSGPRINT dd statement used by CA-Disk. The user program merely formats the message text as desired into a 115-character print line (with trailing blanks for messages shorter than 115 characters). Module ADSUT028 is then called, passing the name of the user's program, a message number to be assigned to the message, and the message text to be printed.

The module must be invoked through the CA-Disk module intercept interface (MI). The example shows how to do this. To use this interface, the CA-Disk load library must be available when you link-edit your program, as documented in “Assembly and Link-Edit of CA-Disk Modules” on page 299.

The module and its interface are as follows:

Module Name — ADSUT028

Module function - Print message to MSGPRINT dd statement:

TABLE 144. Parameters Available to Module ADSUT028

Parm #	Type	Size	Description
1	Input	CL8	Name of program issuing the message
2	Input	CL3	Message number to be used (Will be prefixed with “U”)
3	Input	CL115	Message text

Example Use of ADSUT028

Assembler language call -

```
CALL MI, (DMSPGM,USERPGM,MSGNBR,MSGTEXT),VL
```

Assembler constants for call -

```
DMSPGM    DC CL8'ADSUT028'
0USERPGM  DC CL8'xxxxxxxx'
MSGNBR    DC CL3'001'
MSGTEXT   DC CL115'message text'
```

Message Formatting

All printed messages are prefixed with a 13-character identifier, consisting of the 8-character name of the module issuing the message, an asterisk or a space, and the 4-character message number.

Some of the messages issued by CA-Disk modules are constructed by a parameter-driven module that uses a combination of passed parameters and values in an appropriate entry in the DMSMSGES member of the parmlib data set. Prior to Release 7.0, the module called to format these messages was ADSDM190. In Release 7.0, all calls to DM190 were changed to call ADSDM590 instead, using a slightly different parameter structure.

If you have written programs to call DM190, your messages will now be prefixed with USRDM190*nnnn (where nnnn is your message number). An additional message will then be issued as follows: ADSDM190 9999 IF MI CONTROL STACK IS ACTIVE, MODULE ISSUING PRIOR MESSAGE WAS mmmmmmmm. To eliminate these extra messages you must change your programs to call DM590 instead of DM190, and insert the 8-character name of your module as the first parameter being passed to DM590. For example, assume your program named USRPGM03 has the following call:

```
CALL MI,(DM190,MSG5021,PARM1,PARM2),VL
```

It should be changed to:

```
CALL MI,(DM590,USRPGM03,MSG5021,PARM1,PARM2),VL
```

with the following definitions needed as well:

```
DM590     DC CL8'ADSDM590'
USRPGM03  DC CL8'USRPGM03'
```

Each entry in the DMSMSGES member of the parmlib data set consists of a message number, post-message construction action code, and a series of one or more field and literal specifications. The format of each entry, which is variable in length, is described below.

TABLE 145. Format of the DMSMSGES Member of PARMLIB

Columns	Description		
1-4	Message Number - A four-digit zoned decimal number that uniquely identifies the particular message		
5	Action Code - A one-character designation of action to be taken after the message is constructed and printed. It can be one of the following:	C	Continue by returning to the calling module
		T	Terminate Without A Dump and user code 16
		D	Terminate With A Dump and user code 20

TABLE 145. Format of the DMSMSGES Member of PARMLIB

Columns	Description		
6-n	Field / Literal - One or more variable-length entries can appear, with the first character of each item describing the type of data to be inserted next in the message:	C	Designates a CHARACTER field of a length specified by the next one or two digits. The next parameter points to the value to be moved to the message.
		D	Indicates that a DATE in the (ydd) discontinuous binary format is to be processed as the next parameter
		X	Specifies a HEXADECIMAL field of length specified by the next digit. The value to be unpacked and translated is at the location pointed to by the next parameter. It can be 1 to 4 bytes in length.
		F	Specifies a FULLWORD field. The value to be unpacked and translated is at the location pointed to by the next parameter.
		H	Specifies a HALFWORD field. The value to be unpacked and translated is at the location pointed to by the next parameter.
		/	Indicates that a LITERAL begins with the next character and continues up to a terminating slash or the end of the member entry

Whenever an error is encountered by the message processing module, it will print an error message consisting of the prefix (ADSDM590) followed by appropriate error text. It will then print the completed portion of the message it was directed to construct. Depending on the nature of the error, this can be only the identification of the calling module.

CHAPTER 7

Problem Analysis/Aids

Computer Associates takes pride in providing leading edge products using state of the art software technology. Providing high quality technical support is of equal importance. In an effort to maintain this standard for an increasing customer base, we request your assistance in performing as many of the following diagnostic steps as are relevant prior to contacting the Technical Support Center.

- Read the documentation carefully.
- Reconstruct the events and write them down.
- Run some simple test cases.
- If the job abended, find the name of the data set that CA-Disk was processing at the end of the Module Control Stack (see "Module Intercept Control Stack" on page 272 for details). If you are running an implicit function (that is, DMS), try an explicit run (that is, ARCHIVE) against the offending data set. If this abends, bypass the data set in your implicit run until you find out what is special about the offending data set.

If a problem arises that requires technical programming or analysis support, contact us at the following telephone number.

```
(800) 889-0226
```

We recommend that you not ask for a specific support representative unless you are already working with that individual on your particular problem.

When You Call In

When you call in, you will need to provide the following information:

- Your organization name
- Your name
- A phone number where you can be reached
- The CASE number or PI tracking system number, if one was assigned to the problem previously by a Customer Service Representative.

For new problems, you will need to provide:

- The release of CA-Disk that you are running
- The machine type on which you are running (that is, IBM 3090, Siemens)
- The operating system and release on which you are running (that is, MVS 5.2)
- If the request is for a CA-Disk Open SVC zap, the PTF UZ number that you are running under
- The function(s) of CA-Disk that you are running
- If this is a security-related problem, the security package type, release, options installed, and the authority level of the user
- Can you recreate the problem?
- Has the problem occurred before?
- Has the function ever worked correctly?
- For how long?
- Under what conditions?
- Any changes made to your system between the time that CA-Disk worked and it didn't work
- The exact sequence of events that led to the error
- The JCL from all jobs involved.
- The module name, message numbers and complete text of any CA-Disk messages that were issued.
- If you are running auto-restore, examine the message output from the DMSAR started task. If you are not getting any message output, set the //MSGPRINT DD SYSOUT= statement in the DMSAR proc to be printed or routed to your favorite spooling package. Debugging of auto-restore cannot be accomplished without the MSGPRINT output.

- The message numbers and complete text of any system messages that were issued (that is, from the JES2 log).

- The console log from around the time of the error, if it contains messages that do not appear on the JES2 log. For instance, I/O errors or bad data sets can result in IEA000I messages, and ISPF 806 abends can result in other IEAxxxi messages that will appear on the console log and not on the JES2 log.

- If there is a dump, the abend code and module intercept control stack information. Locate the CA-Disk control stack in user subpool zero, and identify the failing module. See "Module Intercept Control Stack" on page 272 for details.

  Note: Because CA-Disk uses GETMAIN areas for much of its processing, an ABEND-AID dump is almost never sufficient to troubleshoot problems in CA-Disk. Therefore, CA-Disk uses the ABNLDUMP dd statement to get a complete SYSUDUMP in addition to an ABEND-AID dump.

  SVC dumps have an IEA911E message on the JES log, or the second word of the PSW in the dump points to a location not contained in the SYSUDUMP. If you have such an SVC dump, print it using the LPAMAP, LOGDATA, and PRINT CURRENT commands of the program AMDPRDMP. Also use the LISTLPA command of the program AMBLIST to get the locations of the system modules (XA users must follow other procedures.)

  The programs AMDPRDMP and AMBLIST are described in the IBM manual OS/VS2 MVS System Programming Library: Service Aids. See your systems programmer for details.

  If you encounter a problem while running CA-Disk Stand-Alone Restore (DMSSAR), vary a printer online and retry the restore. If you go into a wait state, record the contents of the PSW. If a dump is produced, let the entire dump print. If you need to contact the Technical Support Center for assistance, please try to call from a phone near the console you are using for the restore because the support representative can need you to display certain portions of memory or other information at the console.

- A list of all active system parameters. You can specify sysparm SYSPARMS with a value of S to get this list.

- A list of all user exits that you have installed in CA-Disk.

- A list of all user exits in other products at your site, if they are related to the CA-Disk function that you are trying to run (that is, RACF user exits for problems with the CA-Disk/RACF interface).

- Results from tests that you have run.

- Any other related output.

- Any other information that can help to solve the problem.

- What exactly do you want the Technical Support Center to do?

  Incomplete information can delay or prevent solutions to problems. If we do not receive a response within four weeks for a request for more information, we will consider the problem closed. If it is not possible to provide the information within four weeks, let us know so we can keep the problem open.

  When you receive a fix to a problem, please confirm, good or bad, the results of the fix to the support rep. If we do not receive a response within two weeks, we will consider the problem closed.

This policy is to help us give you better service by responding to your problems in a timely manner. If you have any suggestions for changes to this policy, please contact the Technical Support Center. We always welcome suggestions to provide better service.

Sending Information to the Technical Support Center

In some cases the CA-Disk Technical Support Center representatives can request that you send information regarding a problem directly to them. Please document the problem completely before sending it, including:

- Your organization name and address.
- Your name.
- A telephone number where you can be reached.
- The PI problem tracking system number assigned by the support rep. Write this PI number on ALL documents and on the OUTSIDE of the package.
- Other information from the list above. The list itself can be photocopied and included with the package.
- The importance of the problem, so that we can assign it a priority.

  If you have to send a dump in to the Technical Support Center, please provide the dump on paper (or if it is large, on tape). Dumps on microfiche will be accepted only if your installation cannot provide it on paper or on tape. Organizations that cannot ship dumps or other listings due to security restrictions can or cannot receive assistance over the phone. Contact you local Technical Support organization for further instructions.

Diagnostic Tools

CA-Disk is composed of numerous callable modules, each of which performs a separate unique function. CA-Disk components are driven by executing a control module which, in turn, calls all the lower-level modules necessary to complete the desired task.

Module Intercept Control Stack

Calls between all modules are done through a common interface routine, MI. This routine not only provides an interface between modules, but also provides the base for detailed debugging facilities. Three major functions are provided by the MI interface.

For each call, MI is passed the name of an executable load module. MI first determines if the module is already in storage. If it is, linkage is established, and control is passed to it. If it is not in storage, MI issues a load for the module and then passes control to it. Upon return to MI from the subroutine, MI returns control to the original caller.

During this process of interfacing between driver and subroutine modules, MI also keeps track of all modules currently active. Information about all active modules is kept in a table referred to as the module intercept control stack. The control stack is not maintained when MI fast path is used, which will be discussed shortly.

At the time of the first call through the interface, a control block is dynamically constructed in subpool zero. To facilitate finding it in a memory dump and making use of its contents, it is allocated on a 32-byte boundary and is enclosed with graphic headings.

This control area includes a complete description of the active portion of the hierarchic control structure of the executing program. It collects in a structured format information that is scattered through the user partition or region.

Finding the Module Intercept Control Stack

The module intercept control stack can be found in any CA-Disk abend SYSUDUMP listing. Instructions for getting a dump during abends of ISPF functions are supplied in the CA-Disk installation library in the member SPFTRACE. Use the following steps to find the module intercept control stack:

1. Find the INDEX for the dump. It is the last page of the dump.
2. Go to the page for entry “USER SUBPOOLS”.
3. Within the next few pages you should see the STACK in the right-hand portion of the listing. It will look like the example shown.

Example of Module Intercept Control Stack

The following example shows how the module intercept control stack will appear in CA-Disk SYSUDUMP listings:

```
00000 00000000    *...............................*
00000 00000000    *...............................*
00000 00000000    *...............................*
0E2E3 C1C3D240    * MODULE INTERCEPT CONTROL STACK *
5D740 40D7D340    *LVL MODULE RS SA RT EP PL *
00000 00000000    * 0 EXTERNAL ...................*
95128 00000000    * 1 ADSMI048 ...................*
A4BA0 0009565C    * 2 ADSDM180 ...................*
B6498 000A5880    * 3.ADSDM181 ...................*
B27A0 000B6A00    * 4 ADSDM008 ...................*
A1678 000A4014    * 5          ...................*
                    .    .     .
                    .    .     .
                    .    .     .
                    .    .     .
00000 00000000    *49          ...................*
44040 40404040    *      0000033 CALLS ISSUED      *
7E7E7 E7E7E7E7    *XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*
04040 40404040    *                                *
04040 40404040    *      DMS DIAGNOSTIC AREA       *
04040 40404040    *                                *
440C9 E2406040    * DSNAME OR F1DSCB AT ABEND IS . *
04040 40404040    *ADS.DMS.TEST.DSN01              *
04040 40404040    *                                *
04040 40404040    *                                *
440C9 E2406040    * VOLUME OR F4DSCB AT ABEND IS . *
04040 40404040    *WORK01                          *
04040 40404040    *                                *
04040 40404040    *                                *
5C140 40404040    *     END OF DIAGNOSTIC AREA     *
7E7E7 E7E7E7E7    *XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*
00000 00000000    *...............................*
00000 00000000    *...............................*
```

A brief description of each of the fields comprising a stack entry is described below:

TABLE 146. Module Intercept Control Stack Fields

Field Name	Description
Control Level (lvl)	The hierarchic level of function with the operating system being considered level 0. Up to 26 control levels can be accommodated
Module Name (module)	The eight-character name of the module most recently invoked at that level in the stack. A flag of hex FF will appear in front of the module name that is active at the time of the dump. This will print as a period on the graphic side of the dump listing.
Save Area (sa)	The second fullword following module name contains the address of the SAVEAREA of that module. This field is established at the time the module calls a lower-level module by way of the MTF INTERFACE. Until the first call, it is zero.
Return Address (rt)	The third fullword following the module name contains the return address from the most recent call to a subordinate module via the MTF INTERFACE. As with the save area pointer, it will be zero until the first call.

TABLE 146. Module Intercept Control Stack Fields

Field Name	Description
Entry Point (ep)	The fourth fullword following the module name is the address of the entry point for the module. It is placed there by the MTF INTERFACE immediately before giving control to the module. This value can be subtracted from the return address to determine the displacement of the most recent call made by a module (CLIST).
Parameter List (pl)	The fifth fullword following the module name, and the last value on each line of the dump listing, is the address of the parameter list passed to the module. It will point to the address of the parameter following the name of the module being invoked.

At the bottom of the STACK is the NUMBER OF CALLS issued prior to abnormal termination. It can be used in conjunction with the Module Trace Facility to obtain a trace of module names and parameter values for a reasonable number of calls prior to the failure.

MI Fast Path

Introduced in Release 7.0 was a streamlined path through MI that we call “MI fast path”. As the name implies, the purpose of this facility is to reduce the overhead associated with tracing and control stack maintenance. Although this can improve system performance up to 30 percent (depending on the function being executed), it also eliminates the debugging capabilities of MI. The fast path is keyed off the dd statements supplied in the JCL. If any dd statements start with “MTF”, the slow path through MI will be executed, which functions exactly as versions prior to Release 7.0. However, if no “MTF” dd statement is found, the fast path is taken.

How does fast path work, and how does it affect debugging? Basically, the first call that any module issues to another goes through the standard MI interface. Prior to passing control to the called module, however, MI replaces the VCON to MI with an address to an offset inside a branching table. This causes all subsequent calls to be routed directly to the called module (with four intervening instructions) without MI seeing the call. This causes the control stack to be maintained only on the first call from one module to another, which is useless for debugging purposes. We therefore overlay the control stack heading line (LVL MODULE RS, etc.) with a literal “** CONTROL STACK UNRELIABLE **”. This also causes the “number of calls issued” value to be considerably less than the actual value.

There are occasions where this path cannot be used to pass program control. The primary one to be aware of is when the called module name can change dynamically throughout the course of execution, as in the following example:

```
IF (CLI,OPTION,EQ,C'1')
   MVC MODNAME,=CL8'ADSDM415'
ELSE
   MVC MODNAME,=CL8'ADSUT208'
ENDIF
   #CALL MODNAME,(PARM1,PARM2),VL
```

This causes a problem in fast path processing because the VCON that points to MI will be replaced with an address that points directly to the module being called. Unfortunately, this will cause all of the following calls through this code to be passed to the first module that was called, even if the module name changes. This can be circumvented in two ways. The first is to include any dd statement that begins with MTF (such as MTFDUMMY). This will cause CA-Disk to use the slow path through MI for all program transfers. Although this will achieve the desired results, you will also be giving up the performance improvements in fast path.

This leads us to the second — and preferred — alternative of changing the linkage of MI to MI2. This tells the module intercept facility to not replace the VCON to MI for this particular call, but to go ahead and use fast path in

any other calls (assuming no MTF dd statements were specified). So the updated code for the call above would look like this:

```
#LINK MODNAME,(PARM1,PARM2),VL
```

The other area of concern can be if you are using the MI facility to transfer calls between your own modules. If you have any assembler routines such as the one above, they should be changed to call MI2. Also, if you are using macros that have a hard-coded “=V(MI)” to transfer control, these will quite possibly cause a problem.

Another area to watch out for is if you are using MI to transfer control between programs written in a high-level language, such as COBOL or even PL/1. The compilers for these languages generally produce only one VCON to MI, which can lead to the same problem as that described above where the module name in an assembler program can change dynamically. In these languages you should always transfer control through MI2, although specific code was put in the module intercept facility to detect calls originating from COBOL programs so that the address of MI would not be replaced.

DMS Diagnostic Area

Immediately below the module intercept control stack in CA-Disk dumps is the DMS diagnostic area. This area contains a data set name and volume serial pertaining to the data set being processed at the time CA-Disk abended. This data is maintained whether or not MI fast path being used.

Following the literal “DSNAME OR F1DSCB AT ABEND IS - “ is the name of the data set being processed. Following the literal “VOLUME OR F4DSCB AT ABEND IS - “ is the volume serial of the volume on which the data set resides.

Turn to the topic “Example of Module Intercept Control Stack” on page 274 for an example of the CA-Disk diagnostic area as it will appear in CA-Disk dumps.

Module Intercept Tracing

The third function performed by MI is the ability to generate a trace of all calls and returns of control between driver and subroutine modules. This facility is normally turned off during CA-Disk execution. It can be initiated if it is necessary to trace the interactions between modules to determine the cause of problems or errors.

This feature is controlled by providing special JCL dd statements. When MI detects these dd statements, all code necessary to perform the trace is invoked. This code is in the form of loadable modules and does not incur extra overhead if not used. Use of the trace facility requires roughly 30K extra memory.

The dd statements needed to invoke the MI trace facility are:

TABLE 147. MI Trace Facility DD Statements

DD	Description
MTFIN	An input stream (DD *) containing a single trace control statement (described later)
MTFINOPT	A second input stream. Control statements can be included to confine the trace to modules specified by the user. Omitting this dd causes all modules to be traced.
MTFINPRT	SYSOUT class that will contain images of control statements supplied in the MTFINOPT dd stream. This statement is required only if the MTFINOPT dd is also supplied.
MTFPRINT	SYSOUT class to which trace output should be directed
MTFPRT	SYSOUT class to which trace control statements should be directed

TABLE 147. MI Trace Facility DD Statements

DD	Description
MTFTABLE	This dd refers to a table used by the MI trace facility to format the trace output. Sample JCL in member MTFTJCL of the CA-Disk installation library is supplied to create the table. Use of this table creates a formatted trace with titles and parameter names.
MTFTIME	This dd statement produces a formatted report displaying module performance statistics. A sample is shown in Figure 13. When using this dd, sysparm "IXMQTIME0000" MUST be set to it's default value of 0000. Otherwise an 0C9-09 in module ADSMI101 will occur because both the sysparm and the dd statement use STIMERS, and MVS only allows one timer to be active at a time.

Sample JCL to trace batch runs is supplied in your SAMS.DISK.INSTALL member TRACEDD. A CLIST to allocate JCL to trace ISPF functions is supplied in member SPFTRACE of SAMS.DISK.DMSCLIB. Both provide the following JCL to activate a trace:

```
//MTFPRT   DD  SYSOUT=*
//MTFPRINT DD  SYSOUT=*
//MTFINPRT DD  SYSOUT=*
//MTFIN    DD  *
0000000/05000/00NNFB
Use this control card when using the MTFTABLE dd statement
0000000/05000/03YNVB
//MTFTABLE DD  DISP=SHR,DSN=SAMS.DISK.Rnn.MTFTABLE
//MTFTIME  DD  SYSOUT=*
//MTFINOPT DD  *
INCLUDE
MBR=ADS.....
```

These DD statements are strictly for diagnostic purposes and should be used under temporary circumstances. They produce large amounts of output making it nearly impossible to show output samples created by each DD

statement. However, as a simple example of the type of output they produce, the following report is output created when using the MTFTIME DD statement:

FIGURE 13. Sample Report produced by the MTFTIME DD Statement

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

 Display  Filter  View  Print  Options  Help
SDSF OUTPUT DISPLAY SBDLMFST JOB00042  DSID   102 LINE 1       COLUMNS 02- 133
COMMAND INPUT ===> _                                         SCROLL ===> CSR
11/12/98                        M O D U L E   D E V E L O P M E N T   S Y S T E M                        PAGE   1
VER 2.0                       M O D U L E   P E R F O R M A N C E   A N A L Y S I S                 14.00 05/13/96
MODULE       USE      CALLS     --------------ELAPSED TIME--------------      ----------------CPU TIME----------------
 NAME       COUNT    ISSUED      PATH       MAX       MIN       AVG   PCT      PATH       MAX       MIN       AVG   PCT
ADSDM107        1       416    11,963       249       249       249     2     1,072        50        50        50     5
GSMDM012        1                32        32        32        32              31        31        31        31     3
ADSUT077        1         1      445        53        53        53              57         2         2         2
ADSUT193        1         1      391        65        65        65     1        55         2         2         2
ADSTS048        1         1      326        36        36        36              52         2         2         2
ADSTS148      648       648        1        10                          1                   2                         9
ADSTS147      648       711                 13                          1                   2                        10
ADSTS149        3         3       84        85                 28       1        13         4                   1
ADSUT104        3
ADSUT035      710        16               158                          3                  29                        11
ADSDM085       21
ADSDM081      390       390       24        16                          1                   2                         6
ADSDM083      391                 24       104                 24      81                  23                        15
ADSDM108       13       125      116        98                  8       1        54         6                   1     1
ADSDM458        1         1    1,386        86        86        86     1       675         2         2         2
ADSDM175        1     1,307    1,299       336       336       336     3       672       111       111       111    10
ADSUT022      658       656               55                          2                   2                        10
ADSUT020        1         5       51        50        50        50              3         2         2         2
ADSUT108        4
 Aa   B TCPIP        R 4  C 21        18:01  11/12/98
```

MI Trace Control Statement Format

The following section describes the control statement image to be included in the MTFIN dd control statement stream. When this dd and control statement are omitted from the CA-Disk JCL stream, no processing will be performed. However, the MI control stack will be maintained in either case.

TABLE 148. Format of the MI Trace Control Statement

Cols	Description		
1-7	Call Skip Count: number of calls to ignore prior to activation of the trace — Default is zero		
9-13	Call Trace Count: number of calls to be listed during execution; the trace will deactivate after this count is exhausted — Default is 100		
15-16	Fixed Format Parameter Length: byte length of parameter values to be printed in hex and graphic format when the display option is “f” — Default is 4. Special values in this field cause parameter trace routines to attempt to automatically determine the length of each parameter to be traced. Any other value entered causes that specific number of characters of each parameter to be traced.	00	Attempt to automatically deter-mine the length of each parm
		01	Attempt to automatically deter-mine the length of each para-meter and dump all format-1 DSCB areas as 140 chars.
		02	Same as 01 except it dumps all format-4 DSCB areas also
		03	Attempt to determine the length of each parameter but set a minimum trace length of 16 characters. Also, dump all format-1 DSCB areas as a full 140 characters.
17	Module Title Display Option:	Y	Produce titles. If module title cards are supplied, the Default is Y
		N	no titles
18	Module Selectivity Option:	Y	Indicates if the tracing is to be confined to those modules that have been defined in the attribute data set. Note: The trace count is not confined to selected modules.
		N	The Default is No confinement
19	Module Parameter Display Option:	F	Designates fixed format
		V	Designates variable format. If V is specified and either no parameter definitions are supplied or a discrepancy exists between the actual number of parameters and the definition count, the default will apply.
		N	The Default is that No values are to be listed
20	Entry / Exit Display Option:	X	Indicates that parameter values are to be displayed only at exit time
		B	The Default is to display parameter values at both entry and exit time

MI Trace Options Control Statement

In addition to the trace control statement supplied in the MTFIN dd input stream, the user can also supply trace module selection criteria through a different dd statement control stream. This allows the user to considerably reduce the amount of trace output that will be generated. Control statements are supplied through the MTFINOPT dd statement. The dd statement and control statements are optional.

All control statement data for these statements is freeform format. Only one option can be specified per statement

image. Option name can begin in any column. The following control statements are supported:

TABLE 149. MI Trace Options and Definitions

Option	Definition
INCLUDE/ EXCLUDE	Either the word INCLUDE or EXCLUDE can be specified in a statement image to indicate that values specified on subsequent control statements are to be traced (include) or not traced (exclude). Only one include or exclude statement can be present in the input stream. This statement is REQUIRED.
MBR=	This statement allows the specification of a single member to be included or excluded in the trace output. More than one MBR= statement can be present to allow the selection of more than one member. If followed by ",I", only calls IN are traced (",O" for calls OUT, ",B" for both).
PREFIX=	This statement allows the specification of a member name prefix that will be used to validate members to be excluded or included in trace output. More than one can be supplied.
PATH=	This statement allows the user to specify a path of modules to be included or excluded from trace output. The member specified and all of its subordinate modules will be used for trace member validation.

MI Trace Output

Whenever the appropriate control statement specifications are satisfied, a TRACE LISTING is produced. In addition to the date and time, the following information is provided for each call and return.

TABLE 150. MI Trace Output Explanation

Item	Definition		
CTL LVL	Hierarchic control level of the module receiving control. The user control module is always at level 1		
ACTIVE MODULE	Member name of the module receiving control by either a call or return. This column provides a chronological list of control location.		
XFR	Type of control transfers	C	call
		R	return
CALRET MODULE	The member name of the calling/returning module that is yielding control to the active module.		
PAR CNT	The number of parameters being passed by the calling module or being returned by the returning module.		
MODULE TITLES/ PARAMETER VALUES	Descriptive titles of up to 50 characters are placed on this line for the two respective module names. Also, the parameter values are listed on succeeding lines with or without descriptive names and variable format.		

When the variable format parameter display is active, trailing blanks will be truncated from parameter values on the listing.

Archive Tape Analysis Utility

This utility is provided to aid in determining the validity of an archive tape volume. In the event that an archive tape volume becomes damaged, this utility can be used to help determine the extent of the problem. It will indicate what data sets are still valid and which ones have been lost or damaged. This information can then be used to determine if data sets can be recovered from the volume and, if so, which ones.

This utility operates independently from other functions. To perform its function it simply:

1. Scans the archive tape volume and retrieves information from the files data set that it finds in both the archive tape and the files data set
2. Compares the BLKID for each data set (if available) and indicates any discrepancies by:
   - placing a *D at the end of the printed line that contained the discrepancy
   - placing an * at the end of the printed line if the values compared equally
3. Interprets its contents and produces a printed report

Executing the Utility

Use the following JCL as an example to execute the utility:

FIGURE 14. Sample JCL for the Archive Tape Analysis Utility

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                      (ANALYSIS) - 01.01                Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000100 //JOBCARD  JOB 'ACCT,INFO','NNNNNNN',CLASS=X,MSGCLASS=T,NOTIFY=&USERID
 000200 //STEP1    EXEC PGM=ADSMI002,PARM='ADSDM455',REGION=1024K
 000300 //STEPLIB   DD  DSN=SAMS.DISK.LOADLIB,DISP=SHR
 000400 //ABNLDUMP  DD  DUMMY
 000500 //FILES     DD  DSN=SAMS.DISK.FILES,
 000600 //         DISP=SHR
 000700 //PARMLIB   DD  DSN=SAMS.DISK.PARMLIB,
 000800 //         DISP=SHR
 000900 //CMDPRINT  DD  SYSOUT=A                                - COMMANDS
 001000 //MSGPRINT  DD  SYSOUT=A                                - ERROR MESSAGES
 001100 //SYSPRINT  DD  SYSOUT=A                                - REPORT
 001200 //SYSUDUMP  DD  SYSOUT=A
 001300 //SYSUT1    DD  DSN=THE.ARCHIVE.TAPE.DATA SET.NAME,
 001400 //         UNIT=TAPE,DISP=OLD,
 001500 //         VOL=SER=VVVVVV
 001600 //SYSIN     DD  *
 001700 ANALYZE ....
 001800 /*
 ****** ***************************** Bottom of Data ***************************
 4 ▌                      Aa   A TCPIP        R 4  C 15        16:42   5/12/98
```

Modifying the SYSUT1 DD Statement

The SYSUT1 dd statement refers to the archive tape volume to be scanned. If the archive tape volume is a multivolume data set, include all volumes in the JCL. You must provide the correct name of the “archive tape” on the SYSUT1 dd statement.

If the beginning portion of your CA-Disk tape has been accidentally overwritten, the operating system will no longer recognize the CA-Disk archives as the first standard label file on the tape. If you have this condition, an attempt can still be made to process any remaining CA-Disk archive data, by adding the following parameter to the SYSUT1 dd statement:

```
// LABEL=(5,BLP),
```

Note: StorageTek Redwood devices are exempt from this facility.

This will cause the OPEN SVC to go past enough tape marks to get to the remaining portion (if any) of the CA-Disk archive data set. You can have to experiment with the number on this parameter, depending on the exact nature of the problem. If additional errors occur trying to read the tape (such as wrong length records), you can attempt to bypass these by adding the following DCB parameter to the same SYSUT1 dd statement:

```
// DCB=(BLKSIZE=32760,DSORG=PS,RECFM=U),
```

If CA-Disk is still unable to read this tape, analysis and recovery of any remaining data cannot be done. If you have other tape copy/salvage utilities in your installation, you can be able to use them to create a new tape from the damaged one, from which CA-Disk can then analyze and recover as much data as possible.

Command Input

The utility accepts a command as input. It allows the user to specify parameters to control tape processing. Only ANALYZE is required. The command format and parameter descriptions follow:

```
ANALYZE COUNT=nnn,ERRORMAX=nnn,DUMPMAX=nnn,EXITNAME=xxxxxxxx,FILENBR=nnn
```

COUNT= - The number of tape records to be scanned during tape processing is a 1 to 10 digit number. The maximum number of tape records is 2,147,483,647 which is also the default. When the designated number of records isread, processing will terminate. If the number of actual tape records is less than the designated number of records, the entire tape is processed. The number of actual tape records can be exceeded due to error processing (See ERRORMAX=). This parameter is optional.

ERRORMAX= - Specify a number between 0 and 32767 indicating the maximum number of errors to be encountered before processing is terminated. It defaults to a limit of 10 errors before termination. This parameter is optional.

DUMPMAX= - As errors with data blocks are encountered, the data blocks are dumped to the MSGPRINT DD SYSOUT stream. Specify a 1- to 5-digit number to limit the number of data blocks to be dumped. When this number is reached, processing will continue, but no more blocks will be dumped. It defaults to a limit of 10 blocks. This parameter is optional.

EXITNAME= - Specify a 1- to 8-character user exit module name to be invoked. This optional parameter can be used to cause the utility to pass logical records from the archive tape to a user's processing routine. When specified, the user exit module will be invoked once for each archive tape logical record processed (logical record formats are described in “Record Formats” on page 396). Compressed records will be decompressed before being passed to the exit.

FILENBR= - Specify the 1- to 5-digit file number with which to start tape analysis. It defaults to the first file. This parameter is optional.

Sample Report

In addition to the printed report, error messages will be generated for all detected error conditions.

FIGURE 15. Sample Archive Tape Verification Report

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

 SDSF OUTPUT DISPLAY ISPDLM1A JOB01459  DSID   104 LINE 1       COLUMNS 02- 133
 COMMAND INPUT ===> _                                          SCROLL ===> CSR
NOV  14, 1997                    A R C H I V E   T A P E   V E R I F I C A T I O N                                  PAGE    1
    ARCHIVE TAPE VOLUME = TRAINS, CREDT = 11/14/1997, EXPDT = 12/31/1999, DSNAME = ISPDLM1.DMSC.D1997318.T110844
 FILE                                          SOURCE    ARC      ARC     STARTING    STARTING    ENDING        TOTAL  PHYSICAL
 NBR. ------------- D S N A M E ------------- VOLUME    DATE     TIME    BLK-PHYS    BLK-REC     BLOCK NO      BLOCKS BLOCK-ID
    1 ISPDLM1.CMD.GEN.DSN                      SSL801 11/14/1997 1108          1           1           1           1 00000000
    2 ISPDLM1.DMS.FILES                        SSL807 11/14/1997 1108          2           2           3           2 00000000
ADSDM455 2177 ARCHIVE BLOCK NUMBER           7 CONTAINS AN ERROR, SEE ERROR LISTING FOR DETAILS
    3 ISPDLM1.DMS.PARMLIB                      SSL807 11/14/1997 1108          4           4          13          10 00000000
ADSDM455 2177 ARCHIVE BLOCK NUMBER          15 CONTAINS AN ERROR, SEE ERROR LISTING FOR DETAILS
    4 ISPDLM1.DMSORCMD.D08NOV95.T100656        MVSS02 11/14/1997 1108         14          14          14           1 00000000
ADSDM455 2113 FILE NUMBER MISSING, EXPECTED          5, FOUND            6
    6 ISPDLM1.DMSORCMD.D09OCT97.T192350        MVSS02 11/14/1997 1108         16          16          16           1 00000000
    7 ISPDLM1.DMSORCMD.D09OCT97.T193049        MVSS02 11/14/1997 1108         17          17          17           1 00000000
    8 ISPDLM1.DMSORCMD.D09OCT97.T193216        MVSS02 11/14/1997 1108         18          18          19           2 00000000
ADSDM455 2113 FILE NUMBER MISSING, EXPECTED          9, FOUND           10
   10 ISPDLM1.DMSORCMD.D10OCT97.T113724        MVSS02 11/14/1997 1108         20          20          20           1 00000000
   11 ISPDLM1.DMSORCMD.D29OCT97.T075435        MVSS02 11/14/1997 1108         21          21          21           1 00000000
   12 ISPDLM1.DMSORCMD.D29OCT97.T081617        MVSS02 11/14/1997 1108         22          22          22           1 00000000
   13 ISPDLM1.DMSORCMD.D29OCT97.T081743        MVSS02 11/14/1997 1108         23          23          23           1 00000000
   14 ISPDLM1.DMSRRPTS.D08NOV95.T162717        SY9SY2 11/14/1997 1108         24          24          24           1 00000000
   15 ISPDLM1.DMSRRPTS.D09OCT97.T192835        SSL804 11/14/1997 1108         25          25          25           1 00000000
   16 ISPDLM1.DMSRRPTS.D10OCT97.T113922        SSL807 11/14/1997 1108         26          26          26           1 00000000
   17 ISPDLM1.DMSRRPTS.D29OCT97.T075528        SSL803 11/14/1997 1108         27          27          27           1 00000000
   18 ISPDLM1.DMSRRPTS.D29OCT97.T081651        SSL803 11/14/1997 1108         28          28          28           1 00000000
   19 ISPDLM1.DMSRRPTS.D29OCT97.T081833        SSL805 11/14/1997 1108         29          29          29           1 00000000
 ⊞ ▌                        Aa   B TCPIP        R 2  C 21        11:16  11/14/97
```

User Exit — Archive Analysis

A user exit can optionally be invoked to provide user programs access to archive tape data. This can be useful in recovering data. The user exit for logical records is passed the following parameters:

TABLE 151. Archive Analysis User Exit

Type	Size	Description		
INPUT	HW	Call Type	When the user exit is called with this halfword containing a value of 1, a logical record from the archive tape is being passed to the exit module. When the halfword contains a value of 2, an end-of-file (not end-of-volume) condition has been encountered on the input tape. No record is being passed. This gives the exit a chance to perform end-of-file processing.	
INPUT	variable	Data Record	When parameter 1 contains a 1, this parameter contains a logical record from the archive tape. The contents should not be used in other cases. Do not use this parameter when parameter 1 is equal to 2.	
OUTPUT	HW	Return Code	This parameter indicates if the user exit routine wants processing to continue passing more records or to terminate processing.	
			H’1’	Continue Passing Records to the Exit
			H’2”	Terminate Processing, Close File and End

Archive Extract Utility

This utility extracts data records from CA-Disk archive volumes. It is normally used to recover data for a data set on damaged archive tapes. It scans the archive tape volume for the specified data set, and then copies the data records or blocks to a sequential data set. For sequential data sets, the output data set is the same format as before it was archived. It can be used directly by application programs. For other data set organizations, the extracted data must be processed by other utilities to reload and recreate the original data set.

This program actually functions as a user exit through the archive analysis utility. The program reviews logical records from the archives, and writes the selected ones to an output sequential file. The user provides the output sequential file to be written to. It must contain proper DCB attributes for the data set to be extracted. Partitioned data sets cannot be recovered by this utility. Only one data set can be extracted for each execution of the utility.

This program places in the output file data that can most easily be processed by the user. In many cases the data is extracted into the exact format from which it was archived. In other cases the data will be input to a reload utility to actually recover the data into a usable format.

Also note that when CA-Disk encounters data sets with records longer than 32,760 bytes, those records will be split up into records of 32,760 bytes, and message 2961 issued to let you know this occurred.

Output from the Extract Utility

Following are descriptions of the contents of the output file for the different data set organizations processed:

```
DSORG=PS and DA (PHYSICAL SEQUENTIAL and DIRECT ACCESS)
```

The data blocks from these files are written sequentially in the same order that they were originally read from disk. The DCB attributes are taken from the archived data set's VTOC information that was stored in the archives. For a DA file being extracted, you should also specify DCB=DSORG=DA on the //SYSUT2 dd statement.

Sequential data sets and DA files with fixed-length records should be usable directly as extracted, although for DA files you must also ensure that all special attributes needed are assigned to the data set before being used by application programs.

However, DA files with variable or undefined-length records (RECFM=V or U) will probably not be usable directly as extracted. These data sets usually have special formatting requirements. All records can be extracted, but the formatting needs are not known by the extract function.

If the data set was originally in the unmovable format, care must be taken to ensure it is allocated properly on the DASD device. These data sets must be extracted to the same device type from which they were archived.

```
DSORG=IS and AM (INDEXED SEQUENTIAL and VSAM)
```

The logical records archived from the original data set are written to the sequential output file. The output file is written with the DCB attributes specified on the JCL statement for the output file.

To recover the data set, the extracted logical records must be processed by the IEBDG and IDCAMS utilities for ISAM and VSAM respectively. These utilities will use the logical records to re-create the data set. If the ISAM/VSAM records are out of sequence, sort the records on their appropriate key fields before using IEBDG/IDCAMS to load them.

Executing the Extract Utility

To recover a data set, the user must know the name of the data set and the volume serial of the archive tape volume on which it resides. The EXTRACT command is to be placed in the command input stream of the archive tape analysis utility. Place it immediately after the ANALYZE command. (Turn to “Archive Tape Analysis Utility” on page 281 for a discussion on how to specify and process the ANALYZE command.) Specify ADSDM460 as the value to the EXITNAME parameter in the ANALYZE command to cause the EXTRACT utility to be invoked.

EXTRACT command

```
EXTRACT DSNAME=,TIME=
```

DSNAME= - This required parameter specifies the name assigned to the data set to be extracted.

TIME= - If there is a possibility that more than one data set resides on the archive tape with the data set name specified, specify this parameter to further qualify the data set. Specify the time the data set was archived. This value must be equal to the archive time displayed in the DSNINDEX record for the data set. Use the CA-Disk LISTD function to determine this information.

JCL Example

Use the following JCL as an example to execute the utility:

FIGURE 16. Sample JCL for the EXTRACT Utility

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
EDIT                        (EXTRACT) - 01.00                Columns 00001 00072
Command ===> _____________________________________________________ Scroll ===> CSR
****** ***************************** Top of Data ******************************
000100 //JOBCARD  JOB 'ACCT,INFO','NNNNNNN',CLASS=X,MSGCLASS=T,NOTIFY=&USERID
000200 //STEP1    EXEC PGM=ADSMI002,PARM='ADSDM455'
000300 //STEPLIB   DD  DSN=SAMS.DISK.LOADLIB,DISP=SHR
000400 //ABNLDUMP  DD  DUMMY
000500 //CMDPRINT  DD  SYSOUT=A                              -  COMMANDS
000600 //MSGPRINT  DD  SYSOUT=A                              -  ERROR MESSAGES
000700 //PARMLIB   DD  DSN=SAMS.DISK.PARMLIB,DISP=SHR
000800 //SYSPRINT  DD  SYSOUT=A                              -  REPORT
000900 //SYSUDUMP  DD  SYSOUT=A
001000 //SYSUT1    DD  DSN=ARCHIVE.TAPE.DATA.SET.NAME,
001100 //         UNIT=TAPE,VOL=SER=VVVVVV,DISP=OLD
001200 //SYSUT2    DD  DSN=DMS.TEST,DISP=(,CATLG),
001300 //         UNIT=DISK,VOL=SER=VVVVVV,
001400 //         SPACE=(TRK,(5,5))
001500 //SYSIN     DD  *
001600  ANALYZE EXITNAME=ADSDM460
001700  EXTRACT DSNAME=SAMS.DISK.TEST
001800 /*
****** ***************************** Bottom of Data ***************************
4 ■                        Aa   A TCPIP        R 4  C 15        16:56   5/12/98
```

Modifying the SYSUT1 and SYSUT2 DD Statements

The SYSUT1 dd statement refers to the archive tape volume to be scanned. If the archive tape volume is a multivolume data set, include all volumes in the JCL. You must provide the correct name of the “archive tape” on the SYSUT1 dd statement. If the beginning portion of the tape has been accidentally overwritten, turn to the topic “Modifying the SYSUT1 DD Statement” on page 282 for additional notes.

The SYSUT2 dd statement refers to the output data set to contain the extracted data. Specify data set name, volume, and allocation parameters for the output data set. You can use any data set name for the output data set on the SYSUT2 dd statement. For DA files, add DCB=DSORG=DA to the dd statement. For ISAM and VSAM, supply

```
DCB=(DSORG=PS,LRECL=ll,BLKSIZE=bb,RECFM=rr)
```

where ll, bb, and rr are appropriate for the logical records being extracted.

Archive Fix Utility

This utility is provided to aid in correcting errors that exist within archive tape volumes. It consists of a single job step that accepts input transactions and, based on their contents, copies all or specific archive data set files from the input archive tape volume to a new output volume. As the copy process continues, errors that are detected on the archive tape are corrected.

Of course, errors cannot always be corrected without loss of data because all the original data cannot be present or is not complete. In cases where the error is simply a bad file number, it can be corrected without data loss. If the error is a truncated or missing record, the rest of the block and any split record at the beginning of the next block will be bypassed. The file can or cannot be able to be restored by normal CA-Disk restore processing. It is then the user's responsibility to inspect the restored data set for errors and correct them manually. When records of an archived PDS member are corrected, messages are printed to indicate the members that can contain data errors.

It accepts as input a CA-Disk archive tape volume. It does not use the files data set for processing the input tape, so the tape need not be currently defined to CA-Disk. The SYSUT1 dd statement points to the archive volume to be scanned.

The output volume is referred to by the ARCHIVE0 and ARCHIVEC dd statements. They are coded exactly as they are for normal archive executions (see “Data Storage Command Language” on page 33 in the User’s Guide for details).

The input command specifies what files are to be copied from the input volume to the output volume. It can optionally specify a file number at which to start and a maximum number of records to process.

A user exit can optionally be invoked to provide user programs access to archive tape data at logical record level (EXITNAME). This can be useful in recovering data.

Note: The fix utility uses the standard CA-Disk archive modules to create the new output data set. This means that all applicable system parameters are used. If data compression is active, the new archive data set will be in compressed format, whether or not the input was. Conversely, the input can be in compressed format. But if compression is not active for this utility, the output will be in decompressed (expanded) format.

Note: For the same reasons as stated in the note above, a record is added to the ARCHVOLS subfile of the files data set for the new output volume(s) created. This new archive volume record for the output tape is not complete. To make this a complete entry, it must be purged and rebuilt into the subfile. This is done by using the “PURGE VOL=” and “REBUILD” commands. Caution: The old ARCHVOL entry must be purged before the rebuild is run.

Use the following JCL as an example to execute the utility:

FIGURE 17. Sample JCL for the Archive Fix Utility

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                      (FIXUTLY) - 01.00                Columns 00001 00072
 Command ===> ____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000100 //JOBCARD  JOB 'ACCT,INFO','NNNNNNN',CLASS=X,MSGCLASS=T,NOTIFY=&USERID
 000200 //STEP1    EXEC PGM=ADSMI002,PARM='ADSDM554'
 000300 //STEPLIB   DD  DSN=SAMS.DISK.LOADLIB,DISP=SHR
 000400 //ABNLDUMP  DD  DUMMY
 000500 //CMDPRINT  DD  SYSOUT=A                                - COMMANDS
 000600 //MSGPRINT  DD  SYSOUT=A                                - ERROR MESSAGES
 000700 //SYSPRINT  DD  SYSOUT=A                                - REPORT
 000800 //SYSUDUMP  DD  SYSOUT=A
 000900 //PARMLIB   DD  DSN=SAMS.DISK.PARMLIB,DISP=SHR
 001000 //FILES     DD  DSN=SAMS.DISK.FILES,DISP=SHR
 001100 //ARCHIVEC  DD  DUMMY
 001200 //ARCHIVE0  DD  DISP=(NEW,KEEP,DELETE),DSNAME=ARCHVOL.DSN,
 001300 //         DCB=(BLKSIZE=20000,DSORG=PS,RECFM=U),
 001400 //         LABEL=EXPDT=00000,UNIT=(TAPE,,DEFER)
 001500 //SYSUT1    DD  DSN=ARCHIVE.TAPE.DATA SET.NAME,
 001600 //         UNIT=TAPE,VOL=SER=VVVVVV,DISP=OLD
 001700 //SYSIN     DD  *
 001800  COPY .......
 ****** ***************************** Bottom of Data ***************************
 4 ■                      Aa   A TCPIP        R 13 C 9        17:01   5/12/98
```

Modifying the SYSUT1 DD Statement

The SYSUT1 dd statement refers to the archive tape volume to be scanned. If the archive tape volume is a multivolume data set, include all volumes in the JCL. You must provide the correct name of the “archive tape” on the SYSUT1 dd statement. If the beginning portion of the tape has been accidentally overwritten, turn to topic “Modifying the SYSUT1 DD Statement” on page 282 for additional notes.

The ARCHIVE0 and ARCHIVEC dd statements refer to the output archive volumes. When specifying an installation-defined DASD esoteric name for the UNIT= parameter, do not specify a data set name in the override JCL. Instead, use the POOLDEFS member in PARMLIB to specify a data set name for ARCNAME0.

Command Input

The utility accepts a command as input. It allows the user to specify parameters to control tape processing. The command format and parameter descriptions follow:

```
COPY COUNT=nnnnn,FILENBR=(nnnnn,nnnnn,...),ALL,EXITNAME=
```

COUNT= - Specify a 1- to 5-digit number to indicate the maximum number of tape records to be scanned. When this number of records has been read, processing will terminate. Due to error processing, this number can be exceeded. Default is 32,760. Maximum is 32,767.

FILENBR= - Specify up to 100 tape file numbers to be copied from the input volume.

ALL - If this simple parameter is supplied, all files on the input tape will be copied to the output file.

EXITNAME= - Specify a 1- to 8-character user exit module name to be invoked. This optional parameter can be used to cause the utility to pass logical records from the archive tape to a user's processing routine. When specified, the user exit module will be invoked once for each archive tape logical record processed (logical record formats are described in "Record Formats" on page 396). Compressed records will be decompressed before being passed to the exit.

Note: COUNT, FILENBR, or ALL must be specified.

User Exit — Fix Utility: Parameters Passed

The user exit for logical records is passed the following parameters:

TABLE 152. Archive Fix Utility User Exit

Type	Size	Description	
INPUT	HW	Call Type	When the user exit is called with this halfword containing a value of 1, a logical record from the archive tape is being passed to the exit module. When the halfword contains a value of 2, an end-of-file (not end-of-volume) condition has been encountered on the input tape. No record is being passed. This gives the exit a chance to perform end-of-file processing.
INPUT	variable	Data Record	When parameter 1 contains a 1, this parameter contains a logical record from the archive tape. The contents should not be used in other cases. Do not use this parameter when parameter 1 is equal to 2.
OUTPUT	HW	Return Code	This parameter indicates if the user exit routine wants processing to continue passing more records or to terminate processing.
			H'1' Continue Passing Records to the Exit
			H'2" Terminate Processing, Close File and End

Repair Archvols (REPARCH) Utility

This utility (REPARCH) provides the ability to analyze and synchronize the archive volume (ARCHVOLS) records and the tape pool (DMSPOOLS) records in the files data set. Its repair activities include:

- deleting corrupted ARCHVOLS records (broken chains)
- deleting unchained ARCHVOLS records
- synchronizing DMSPOOLS record status with ARCHVOLS records

It also provides for a user exit that can be used to modify (convert) the device type field in the ARCHVOLS records.

Processing Considerations

For estimating purposes, allow approximately 30 seconds to process 1,000 ARCHVOLS records. For integrity reasons, this utility always has exclusive control of the files data set during processing, even in simulate mode, which should be considered along with the time estimate before running the utility.

Output

This utility generates a report listing all deleted ARCHVOLS records and the reason for deletion, and a total of the ARCHVOLS records that were processed and deleted.

JCL — REPARCH Utility

Use the following JCL to execute this utility. (A sample copy is in member REPARCH in the install library.)

FIGURE 18. Sample JCL for the Repair Archvols Utility (REPARCH)

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                      (REPARCH) - 01.00                Columns 00001 00072
 Command ===> ____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000100 //JOBCARD  JOB 'ACCT,INFO','NNNNNNN',CLASS=X,MSGCLASS=T,NOTIFY=&USERID
 000200 //REPARCH  EXEC PGM=ADSMI002,PARM=ADSDM636,REGION=3072K
 000300 //STEPLIB   DD  DISP=SHR,
 000400 //         DSN=SAMS.DISK.LOADLIB
 000500 //ABNLDUMP  DD  DUMMY
 000600 //CMDPRINT  DD  SYSOUT=*
 000700 //FILES     DD  DISP=SHR,
 000800 //         DSN=SAMS.DISK.FILES
 000900 //MSGPRINT  DD  SYSOUT=*
 001000 //PARMLIB   DD  DISP=SHR,
 001100 //         DSN=SAMS.DISK.PARMLIB
 001200 //SYSPRINT  DD  SYSOUT=*
 001300 //SYSUDUMP  DD  SYSOUT=*
 001400 //SYSIN     DD  *
 001500 REPARCH SIMULATE,PURGE=,DEVTEXIT=,POOL=,TMS=
 001600 /*
 ****** **************************** Bottom of Data ****************************

 4 ■                      Aa   A TCPIP       R 4  C 15       17:06   5/12/98
```

REPARCH Command and Parameters

```
REPARCH SIMULATE,PURGE=,DEVTEXIT=,POOL=,TMS=
```

SIMULATE - This parameter causes informational messages and report statistics to be issued, but no updates will be made.

PURGE= - This parameter restricts the scope of ARCHVOL records that are to be examined and reconciled. Values that can be specified, and their meanings are:

- VOLUME — (the default). All ARCHVOL records will be candidates for reconciliation.
- NOPROCESS — No ARCHVOL records will be reconciled.

DEVTEXIT= - If you have a need to modify the device type field in one or more ARCHVOLS records, use this optional parameter to specify the name of the module you have written to perform the conversion.

- Parameters Passed To Exit:

TABLE 153. Parameters Passed to REPARCH User Exit

Type	Size	Description
INPUT	XL1	Simulation indicator X'01'
INPUT	XL110	ARCHVOLS record image
IN/OUT	XL4	4 byte device code

POOL= - This parameter restricts the scope of the CA-Disk tape pool records (in the DMSPOOLS subfile) that are to be examined and reconciled. Values that can be specified, and their meanings are:

- SYNCH (the default) — This parameter synchronizes the status of each record in the DMSPOOLS subfile with the ARCHVOLS subfile. It returns to "scratch" status any pool records indicated as "in-use" but not having a matching ARCHVOLS record. It assigns "in-use" status to any DMSPOOLS records that have a matching ARCHVOLS record.
- FORMAT — Format tape pool based on the value specified for sysparm TPOOLSEQ.
- NOPROCESS — No DMSPOOLS records will be reconciled.

TMS= - This parameter sets the scope of CA1 record updates. This requires the TMSCTLEX user exit to be active.

- EXPIRE (the default) — For those ARCHVOLS records that are deleted as a result of reconciliation, this option will result in CA1 expiring the volume. It requires that sysparm TMSCTLEX user exit be specified in sysparm TMSCTLEX.
- NOPROCESS — No CA1 updates will be performed.

Load Library Status Utility

This utility is a diagnostic facility intended for use by the Technical Support Center. It reports on the current status of the CA-Disk load library, indicating:

- Superzap fixes applied to modules
- Modules that have been modified since the base release
- Modules that are down-leveled from the base release

Note: This utility does not accurately report on load libraries built by SMP/E. Use the SMP/E LIST and REPORT commands instead.

The report has two sections.

1. lists the modules that have Superzaps
2. lists the modules that are different from the base release

JCL — Load Library Status Utility

Use the following JCL to execute this utility. (A sample copy is in member LOADLIB in the install library.)

FIGURE 19. Sample JCL for the Load Library Status Utility

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

 File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                      (LLSTATUS) - 01.12              Columns 00001 00072
 Command ===> _                                             Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000100 //SBDLMLLS JOB (0125,#56),'MULLINGS',CLASS=X,MSGCLASS=T,NOTIFY=SBDLM
 000200 //STATUS  EXEC PGM=ADSMI002,PARM=ADSDM636,REGION=3072K
 000300 //STEPLIB  DD  DISP=SHR,DSN=WORKSHOP.R90.LOADLIB
 000400 //ABNLDUMP DD  DUMMY
 000500 //CMDPRINT DD  SYSOUT=*
 000600 //FILES    DD  DISP=SHR,DSN=SBDLM.DMS90.FILES
 000700 //MSGPRINT DD  SYSOUT=*
 000800 //PARMLIB  DD  DISP=SHR,DSN=SBDLM.DMS90.PARMLIB
 000900 //PUNCH    DD  DUMMY
 001000 //SYSPRINT DD  SYSOUT=*
 001100 //SYSUDUMP DD  SYSOUT=*
 001200 //SYSIN    DD  *
 001300  LOADLIB DSN=SBDLM.DMS.LOADLIB,STATUS=LIST
 001400 /*
 ****** ***************************** Bottom of Data ***************************
 4 ■                        Aa   A TCPIP       R 4  C 15       22:36   5/20/98
```

Loadlib Command and Parameters

```
LOADLIB DSN=,VOLUME=,STATUS=,
```

DSN= - Supply the data set name of the CA-Disk load library in this required parameter.

VOLUME= - If the CA-Disk load library is not cataloged, supply its volume serial number with this optional parameter.

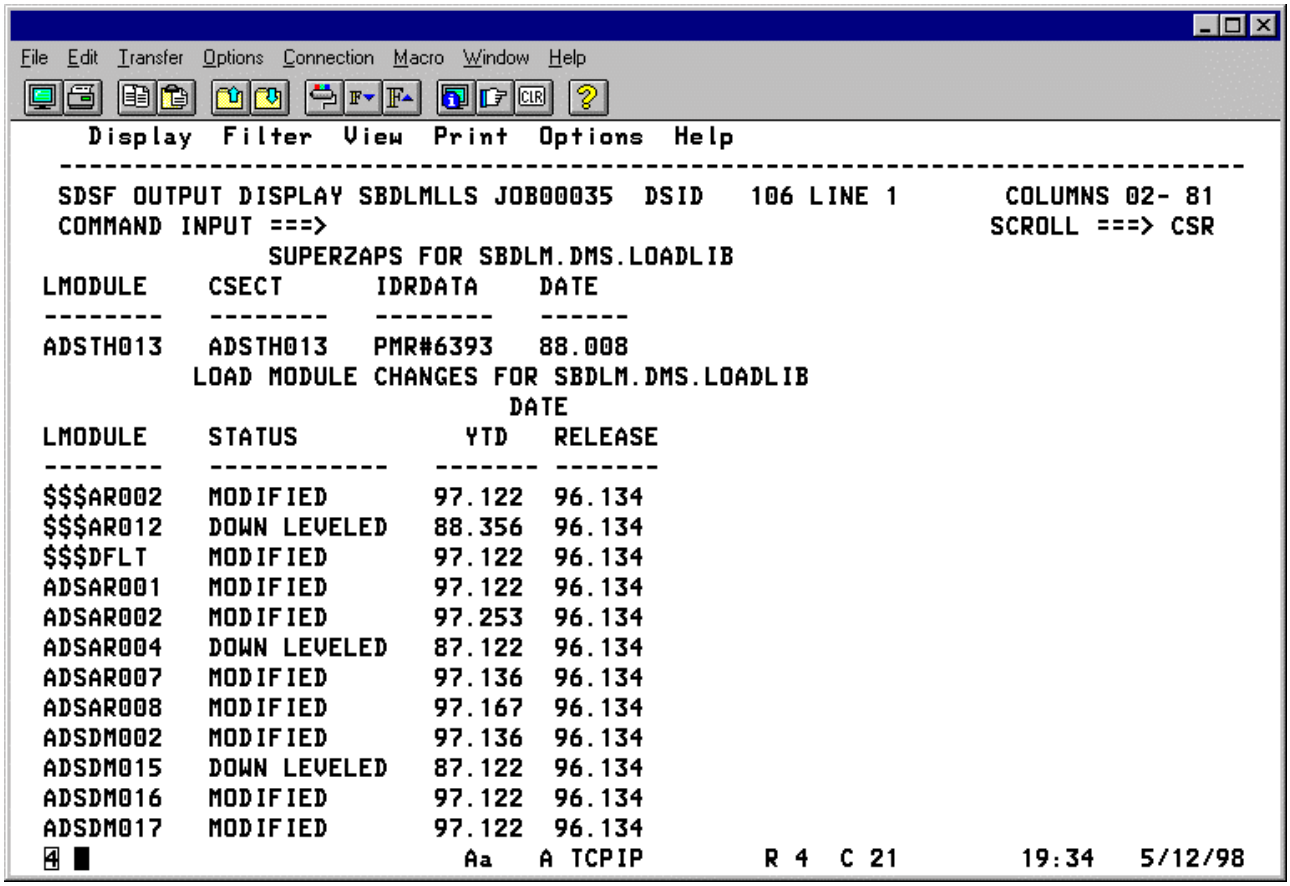
STATUS= - This parameter is used to specify the type of status operation to be done. The following option is available:

- LIST (default) — Report on superzaps applied to modules, and modules not synchronized with the base release of CA-Disk.

Sample Report

As you can see in Figure 20, modules that are not in sync with the base release are identified under the STATUS column as being DOWN LEVELED. These are problem modules that require immediate attention.

FIGURE 20. Sample Load Library Status Report



```
File  Edit  Transfer  Options  Connection  Macro  Window  Help
   Display  Filter  View  Print  Options  Help
 ------------------------------------------------------------------------------
 SDSF OUTPUT DISPLAY SBDLMLLS JOB00035  DSID   106 LINE 1       COLUMNS 02- 81
 COMMAND INPUT ===>                                           SCROLL ===> CSR
                 SUPERZAPS FOR SBDLM.DMS.LOADLIB
LMODULE    CSECT      IDRDATA    DATE
--------   --------   --------   ------
ADSTH013   ADSTH013   PMR#6393   88.008
          LOAD MODULE CHANGES FOR SBDLM.DMS.LOADLIB
                                 DATE
LMODULE    STATUS           YTD   RELEASE
--------   ------------   ------- -------
$$$AR002   MODIFIED       97.122  96.134
$$$AR012   DOWN LEVELED   88.356  96.134
$$$DFLT    MODIFIED       97.122  96.134
ADSAR001   MODIFIED       97.122  96.134
ADSAR002   MODIFIED       97.253  96.134
ADSAR004   DOWN LEVELED   87.122  96.134
ADSAR007   MODIFIED       97.136  96.134
ADSAR008   MODIFIED       97.167  96.134
ADSDM002   MODIFIED       97.136  96.134
ADSDM015   DOWN LEVELED   87.122  96.134
ADSDM016   MODIFIED       97.122  96.134
ADSDM017   MODIFIED       97.122  96.134
4 ■                         Aa   A TCPIP        R 4  C 21        19:34   5/12/98
```

Track Dump Utility

The TRACK DUMP utility allows you to dump the data on a DASD track exactly as it appears on the track. It should be used in problem analysis if you question what data exists on a track.

This utility reads and prints the contents of a track from a DASD volume. The DASD volume can either be online or offline; however, a path must be available to the device (that is, the device must not be “boxed”).

For each track, this utility prints the home address, the capacity (R0) record and all data records on the track. The count information is also printed for the capacity and data records.

The utility accepts either a single absolute track address or a range of absolute track addresses. If the volume is offline, the channel unit address for the volume (CUU) must be supplied; if the volume is online, only the volser need be supplied.

Command parameters NOR and NOD are provided to control the amount of data actually printed.

Use the following JCL to execute this utility. (A sample copy is in member TRACKDMP in the install library.)

```
//TRACKDMP EXEC PGM=ADSMI002,PARM=ADSFH114,REGION=2048K
//STEPLIB   DD  DSN=SAMS.DISK.LOADLIB,DISP=SHR
//ABNLDUMP  DD  DUMMY
//SYSUDUMP  DD  SYSOUT=*
//CMDPRINT  DD  SYSOUT=*
//MSGPRINT  DD  SYSOUT=*
//SYSPRINT  DD  SYSOUT=*
//PARMLIB   DD  DSN=SAMS.DISK.PARMLIB,DISP=SHR
//SYSIN     DD  *
 Specify command as described below:
```

TRACKDMP Command and Parameters

```
DUMP CCHH=,VOL=,NOD,NOR,CUU=
```

CCHH= - Specify an absolute track value, a list of values, a range, a list of ranges, or any combination. The track value is input as cccchhhh (cccc is the cylinder and hhhh is the head). These values are entered in hexadecimal format. Examples of acceptable input are:

```
CCHH=cccchhhh
CCHH=(cccchhhh,cccchhhh,...)
CCHH=cccchhhh-cccchhhh
CCHH=(cccchhhh-cccchhhh,cccchhhh-cccchhhh,cccchhhh)
```

Up to ten values can be provided in the list. A track range counts as one value.

VOL= - The volume serial for the DASD volume you are processing. This parameter is required only for an online volume.

CUU= - The channel unit address for the DASD volume you are processing. This parameter is required only if the volume is offline. If both the CUU= and the VOL= parameters are specified and the volume is online, the supplied volser is compared to the real volser and an error message is issued if they are not the same.

NOD - If this parameter is specified, the home address, R0 record and count fields for all data records on the track are printed. The data portion of the records are not printed.

NOR - If this parameter is specified, the home address and R0 record values are printed for each track. No portion of the data records are printed.

Diagnostic DD Statements

Certain CA-Disk functions will provide additional diagnostic data whenever they detect a specific dd statement being present. The Technical Support staff can request that you supply one or more of these dd statements when researching a reported problem.

```
//ARESDIAG DD SYSOUT=A
```

This dd statement causes the auto-restore function to interpret the control block that is used to initiate the started task; that is, the environment that invoked the auto- restore and the options that are in effect.

```
//MTFLIBDG DD SYSOUT=A
```

This dd statement causes the MI component to display information about each module being loaded; for example, the library in which it was found, the concatenation number, the CA-Disk time and date stamp, etc. It can be used in both batch and online functions.

```
//RECONDMP DD SYSOUT=A
```

This dd statement causes the table of data base names and associated data sets that are extracted from the DBRC recovery control data set to be displayed.

Archive Tape Recovery Utility

The Archive Tape Recovery Utility is intended to perform the function of recovering CA-Disk archive tapes which have been damaged or partially overwritten. It will read a single archive tape as input and bypass I/O errors and optionally skip tapemarks to retrieve any archive data blocks which are still intact.

Each complete data set encountered will then be transferred to a new archive tape volume. This tape volume can then be used by the REBUILD utility to identify it to the files data set. The Archive Tape Recovery Utility is designed to recover only archive tapes. It does not use disk data sets as input or output.

Executing the Utility

Use the following JCL as an example to execute the utility:

FIGURE 21. Sample JCL for the Archive Tape Recovery Utility

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
--------------------------------------------------------------------------------
EDIT                        (TRECOVER) - 01.00              Columns 00001 00072
Command ===> ____________________________________________________ Scroll ===> CSR
****** ***************************** Top of Data ******************************
000100 //JOBCARD  JOB 'ACCT,INFO','NNNNNNN',CLASS=X,MSGCLASS=T,NOTIFY=&USERID
000200 //*******************************************************
000300 //* TAPE RECOVERY UTILITY PROGRAM
000400 //*******************************************************
000500 //S1EXCP   EXEC PGM=ADSMI002,PARM=ADSDM563
000600 //STEPLIB   DD  DISP=SHR,DSN=YOUR.DMS.LOADLIB
000700 //PARMLIB   DD  DISP=SHR,DSN=YOUR.DMS.PARMLIB
000800 //CMDPRINT  DD  SYSOUT=*
000900 //MSGPRINT  DD  SYSOUT=*
001000 //SYSUDUMP  DD  SYSOUT=*
001100 //SYSUT1    DD  DISP=SHR,VOL=SER=XXXXXX,LABEL=(1,BLP),
001200 //         UNIT=TAPE,DSN=YOUR.OLD.ARCHIVE.TAPE.DSN
001300 //SYSUT2    DD  DISP=(NEW,KEEP),VOL=SER=(XXXXXX,XXXXXX),
001400 //         LABEL=(1,SL),UNIT=(TAPE,,DEFER),
001500 //         DCB=(DSORG=PS,RECFM=U,LRECL=0,BLKSIZE=32760),
001600 //         DSN=YOUR.NEW.ARCHIVE.TAPE.DSN
001700 //SYSIN     DD  *
001800  TRECOVER ERRLIMIT=4050,EOFSTOP
****** ***************************** Bottom of Data ***************************
 Aa   A TCPIP        R 4  C 15        18:56  10/15/98
```

Modifying the SYSUT1 DD Statement

The SYSUT1 dd statement refers to the archive tape volume to be scanned. If the archive tape volume is a multivolume data set, include all volumes in the JCL. You must provide the correct name of the “archive tape” on the SYSUT1 dd statement.

If the beginning portion of your CA-Disk tape has been accidentally overwritten, the operating system will no longer recognize the CA-Disk archives as the first standard label file on the tape. If you have this condition, an attempt can still be made to process any remaining CA-Disk archive data, by adding the following parameter to the SYSUT1 dd statement:

```
// LABEL=(5,BLP),
```

Note: StorageTek Redwood devices are exempt from this facility.

This will cause the OPEN SVC to go past enough tape marks to get to the remaining portion (if any) of the CA-Disk archive data set. You can have to experiment with the number on this parameter, depending on the exact nature of the problem. If additional errors occur trying to read the tape (such as wrong length records), you can attempt to bypass these by adding the following DCB parameter to the same SYSUT1 dd statement:

```
// DCB=(BLKSIZE=32760,DSORG=PS,RECFM=U),
```

If CA-Disk is still unable to read this tape, analysis and recovery of any remaining data cannot be done. If you have other tape copy/salvage utilities in your installation, you can be able to use them to create a new tape from the damaged one, from which CA-Disk can then analyze and recover as much data as possible.

If the output tape cannot hold all information from the input tape, you can specify more than one output volume. If you specify only one tape volser and an additional tape is needed, this utility will issue a mount request for a scratch tape. If you create two output volumes, be aware that these tapes are not related to one another. That is, they are not part of a multi-volume group.

Command Input

The command used to direct the execution is:

```
TRECOVER ERRLIMIT= EOFSTOP
```

ERRLIMIT= - This parameter indicates the maximum number of errors to allow before terminating the job. When this limit is reached, processing is stopped and the output tape is closed. The default value is 100.

EOFSTOP - This parameter causes processing to stop when double tape marks are encountered, indicating the end of volume (EOV). Specify this parameter if your archive tape has been damaged.

Do not specify the EOFSTOP parameter if your archive tape has been overwritten or you want to recover past the current end of volume. Processing will continue until the end of the input tape is reached. If any tape marks are encountered, they will be treated as embedded tape marks.

Sample Report

In addition to the printed report, error messages will be generated for all detected error conditions.

FIGURE 22. Sample Archive Tape Recovery Utility Report

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  Display  Filter  View  Print  Options  Help
 ----------------------------------------------------------------------------------------------------------------------
 SDSF OUTPUT DISPLAY SBDLMINT JOB00174  DSID   102 LINE 1       COLUMNS 02- 133
 COMMAND INPUT ===> _                                         SCROLL ===> CSR
1998.289  OCT 16, 1998                          C O M M A N D   S T A T E M E N T S                                  PAGE    1
FRIDAY     1.17 PM                                                                                                 SAMS:DISK 9.0.0
----.----1----.----2----.----3----.----4----.----5----.----6----.----7----.----8----.----9----.----0----.----1----.----2----.----3--
 TRECOVER                                                                                  CMD0001
1998.289  OCT 16, 1998                                       M E S S A G E S                                          PAGE    1
FRIDAY     1.17 PM                                                                                                 SAMS:DISK 9.0.0

ADSDM175 3182 WARNING:  YOU ARE RUNNING SAMS:DISK RELEASE 9.0.0  SOFTWARE
ADSDM175 3182                           WITH A RELEASE 9.0.0B PARMLIB
ADSDM563 3283 TAPE MARK DETECTED AFTER            2 BLOCKS READ
ADSDM563 3286 I/O ERROR, ECB=41000000
ADSDM563 IOB=06000850
ADSDM563 BYTECNT=8000
ADSDM563 UNIT STATUS=06
ADSDM563 CHNL STATUS=00
ADSDM563 3309            4 DATA BLOCKS SKIPPED BEFORE PROCESSING NEXT DATA SET
ADSDM563 3310 CONTROL RECORD FOUND FOR SBDLM.CLIST
ADSDM563 3285 TOTAL OF            2 BLOCKS READ AND TRANSFERRED FOR SBDLM.CLIST
ADSDM563 3310 CONTROL RECORD FOUND FOR SBDLM.DISK.USER.TABLES
ADSDM563 3285 TOTAL OF            1 BLOCKS READ AND TRANSFERRED FOR SBDLM.DISK.USER.TABLES
ADSDM563 3310 CONTROL RECORD FOUND FOR SBDLM.DMS.R900.INSTALL
ADSDM563 3285 TOTAL OF            1 BLOCKS READ AND TRANSFERRED FOR SBDLM.DMS.R900.INSTALL
ADSDM563 3310 CONTROL RECORD FOUND FOR SBDLM.DMS81.PARMLIB
                        Aa   B TCPIP        R 4  C 21        13:54  10/16/98
```

Condition Codes

Expected return codes from this utility are as follows:

TABLE 154. Archive Tape Recovery Condition Codes

CC	Description
0	All data sets have been recovered successfully from the input tape
4	The entire input tape has been processed successfully. However, errors encountered have prevented the recovery of all data sets.
8	The error limit specified on the ERRLIMIT= parameter has been exceeded. Processing of the input tape has been halted; not all data sets have been recovered.
12	Either critical I/O errors have occurred which prevent further processing of the input tape, or OPEN errors for either the input or output tapes have occurred. Not all data sets have been recovered.
16	Command error or an invalid output device type has been specified.

If you receive a return code of less than 16, you still can have recovered some of the data sets to the output tape.

After you have run this utility, run the Archive Tape Analysis Utility to verify that the new archive tape is good. (Just because the Archive Tape Recovery Utility completes successfully does not guarantee the tape is a good archive tape.) If the Archive Tape Analysis Utility indicates the tape is good, use the tape as input to the REBUILD utility. Then run PURGE on the damaged archive tape.

CHAPTER 8

Modifying CA-Disk

We hope that the combination of system parameters and user exits provided eliminates the need to modify any of the CA-Disk source modules. Doing so obviously adds to the difficulty of installing any updated version of the system, since the same or comparable changes will have to be made to the new system as well. Should a source change be needed, however, concerns unique to the assembly and link-edit of CA-Disk modules are presented here.

A common interface module (MI) has been statically linked with each CA-Disk module that calls any other module. A second interface module (MI2) is also available, and should be used in programs you write that call other CA-Disk programs. Turn to the topic "MI Fast Path" on page 275 for more information about usage of MI2 versus MI. Both are included in the CA-Disk load module library for linkage editor processing when assembling modules at your site.

The addressing mode and residency mode of the CA-Disk module that you have modified should remain unchanged. If you change the original program attributes, unexpected results could occur.

You are responsible for changes you make to CA-Disk. This applies to any change to a CA-Disk module, whether made through source code changes or zaps. If you change a CA-Disk module, you are also responsible for properly installing any fixes provided for that module. This means, in part, removing any usermods you have installed to the affected module.

We recommend you make changes as usermods if you install CA-Disk using SMP/E.

Assembly and Link-Edit of CA-Disk *Modules*

If changes are made to the CA-Disk source modules, and for user- written programs using the MI or MI2 interface, the following steps must be taken to obtain proper assembly and link-editing.

1. CA-Disk source code uses structured programming macros. The CA-Disk macro library must therefore be concatenated to the assemble macro library as defined on the //ASM.SYSLIB dd statement. The CA-Disk macro library should be placed first in the concatenation to avoid the possibility of duplicate macro names in conflict with your installation's macros.

2. IBM High-Level Assembler (HLASM) must be used to assemble CA-Disk modules. We will try to allow source members in the CA-Disk.INSTALL library to be assembled with Assembler F.

   However, other CA-Disk programs can have incompatibilities that will require IBM's HLASM.

3. At link-edit time the parameters should specify LET, but not NCAL.

4. At link-edit time the SYSLIB dd statement must point to the CA-Disk load library (DSN=SAMS.DISK.LOADLIB).

The following assembly JCL is provided on the distribution tape in the INSTALL library. You will need to customize several DD statements: STEPLIB, SYSLIB, SYSPUNCH, SYSLIN, SYSIN, SYSLMOD.

```
//*************************************************************
//* SAMS DISK MODULES HIGH LEVEL COMPILE AND LINKEDIT MODULES
//*************************************************************
//* USE THIS JCL FOR SAMS DISK LOAD MODULE UPDATES.
//*
//* USE LKED PARMS (RENT|REUS|REFR).
//*************************************************************
//HLASM    EXEC HLASMCL,REGION.ASM=3072K,
//        PARM.ASM=(OBJ,NODECK,'PC(UHEAD,GEN)',NORLD),
//        PARM.LKED='LIST,MAP',           <--- REUS,REFR,RENT
//        REGION.LKED=3072K
//ASM.SYSLIB   DD  DISP=SHR,DSN=SAMS.DISK.RNNN.MACLIB
//             DD  DISP=SHR,DSN=SYS1.MACLIB
//             DD  DISP=SHR,DSN=SYS1.MODGEN
//ASM.SYSIN    DD  DISP=SHR,DSN=SAMS.DISK.RNNN.SOURCE(MODNAME)
//ASM.SYSTERM  DD  SYSOUT=*
//LKED.SYSLMOD DD  DISP=SHR,DSN=SAMS.DISK.RNNN.LOADLIB(MODNAME)
//LKED.SYSLIB  DD  DISP=SHR,DSN=SAMS.DISK.RNNN.ADMSLOAD
```

The following link JCL is provided on the distribution tape in the INSTALL library. You will need to customize several DD statements: STEPLIB, SYSLIB, SYSPUNCH, SYSLIN, SYSIN, SYSLMOD.

```
//*************************************************************
//* LINK A LOAD MODULE                                        *
//*************************************************************
//*                                                           *
//* INSTRUCTIONS:                                             *
//* REPLACE “CALL” WITH “RENT”, “REUS” OR “REFR”              *
//*************************************************************
//LINK     EXEC PGM=IEWL,REGION=3072K,PARM='XREF,LET,LIST,CALL'
//SYSPRINT  DD  SYSOUT=*
//SYSLMOD   DD  DISP=SHR,DSN=SAMS.DISK.RNNN.LOADLIB
//SYSLIB    DD  DISP=SHR,DSN=SAMS.DISK.RNNN.OBJECT
//SYSUT1    DD  UNIT=SYSALLDA,SPACE=(CYL,3)
//SYSLIN    DD  DISP=SHR,DSN=SAMS.DISK.RNNN.LINKCNTL(MODNAME)
//
```

Issuing Messages

Please turn to "Message Formatting" on page 266 for a description of how to issue messages from a user program or exit module.

Relinking TSO and ISPF Control Modules

Each TSO command processor is associated with a module of the same name in the CA-Disk source library. Control modules for the ISPF functions are also in the source library. They and all of the subordinate (or called) modules are dynamically loaded by the MI interface as they are needed. As a result, each module supporting these online functions can be reassembled and linked into the CA-Disk load library independently.

If you write user exit routines that affect the TSO or ISPF functions, they too can simply be assembled and linked independently into the CA-Disk library. MI will dynamically load and execute your exit routines the same as any other CA-Disk modules, as soon as you supply the sysparms to activate them.

Adding Non-CA-Disk Modules

It is also possible to modify the system without changing or replacing modules, in the sense that CA-Disk modules can be invoked by non-CA-Disk calling modules. Each module given control through the common interface brings into memory those subordinate modules not already loaded. In this manner any function can be initiated in any environment as long as the required dd statements are present.

The Module Interface (MI) Trace aids in testing and verifying changes made to the system. It makes available a printed display of all calls and returns made through the interface module. It also displays the first "n" characters of each parameter value on both module entry and exit.

Coding for an Interactive Environment

If CA-Disk modules are to be used for interactive applications, special programming constraints are often imposed. A good technical understanding of the constraints of the environment as well as the design and constraints of the CA-Disk modules is required. The MI control modules account for these. Programming in other environments can introduce unexpected results.

Module Call Interface

Because of its dynamic nature, CA-Disk installation modules can use the internal interfaces in the same manner as CA-Disk system modules. To dynamically transfer control to a CA-Disk module, the installation module must call via the MI interface with the name of the CA-Disk module to be invoked as the first parameter.

The use of the CA-Disk call interface is illustrated below with an example in assembly language, since it will normally be the language used to invoke the various modules; however, COBOL can also be used to accomplish the invocation. Please note, however, the additional concerns regarding "MI Fast Path" on page 275.

Example — Call Interface

A unique record can be retrieved from a file in the CA-Disk files data set by executing a call in the following manner. This is an example only. Turn to the topic "Common File Handler Interface" on page 33 for specific information on using the DM095 interface.

```
         MVC  DSNKEY,=CL44'PAY.MASTER'
* RETRIEVE THE DATA SET INDEX RECORD
         CALL MI,(DM095,RU,FILENAME,DSNREC,RESULT),VL
         CLC  RESULT,=H'1'
         BH   NOTFOUND
*
DM095    DC    CL8'ADSDM095'  FILE HANDLER CONTROL MODULE
RU       DC    CL2'RU'        SERVICE TYPE
FILENAME DC    CL8'DSNINDEX'  IN  - PARM 1 (FILENAME)
DSNREC   DC   0CL256          IN/OUT AREA
DSNKEY   DC    CL44' '        IN  - PARM 2 (KEY)
         DC    CL212' '       OUT - INDEX DATA
RESULT   DC    H'0'           OUT - RETURN CODE
```

User Exits

Turn to "User Exits" on page 201 for a description of the modifications available via exits. A sample of user exit code is also provided in that section.

Parameter Library Support - Common Parmlib Interface

All functions of CA-Disk require various parameters to guide their execution. These parameters are maintained in a partitioned data set referred to as "parmlib". This is a "read-only" data set for all CA-Disk functions. Any changes to be made to the parameter library must be done with an online editor or some other traditional means of updating a PDS.

User-written modules can read entries from a parameter library member in the same manner as CA-Disk modules, by using the common parmlib interface module. The syntax rules for parmlib entries are described in "PARMLIB

Specifications" on page 476. The following illustrates reading an entry from member=EXMTLIST that can contain a list of data set names that are exempt from processing.

```
         MVC  MBRNAME,=CL10'EXMTLIST'
* Retrieve an entry from the parmlib member
         CALL MI,(ADSTS048,RESULT,MBRNAME,ENTRY#,MAXLEN,                X
              IOAREA),VL
         CLC  RESULT,=H'1'
         BH   NOTFOUND
*
ADSTS048 DC   CL8'ADSTS048'  parmlib interface module
RESULT   DS   H              result
MBRNAME  DC   CL10' '        member name to be read
ENTRY#   DC   H'0'           entry number - 0 retrieves 1st
*                            record, 1 retrieves 2nd, etc.
MAXLEN   DC   H'46'          length of I/O area
IOAREA   DS  0CL46           data from retrieved entry
LENGTH   DS   H              length of record + 2
RECORD   DS   CL44           actual record
```

The parmlib common interface module has the following call structure.

```
CALL MI,(ADSTS048,RESULT,MBRNAME,ENTRY#,MAXLEN,IOAREA),VL
```

TABLE 155. Description of the MI CALL Structure

Name	Description
ADSTS048	Is always the eight-character name of the common interface module to the parmlib: ADSTS048 DC CL8'ADSTS048'
RESULT	The halfword return code set by the control module to reflect the conclusion of the requested service. 1 indicates record found, 2 indicates end of file, and 3 indicates an invalid member name.
MBRNAME	The name of the member in the parameter library in a ten-character field.
ENTRY#	The number of the entry within the member that is to be read. Note that all numbers are relative to zero. The entry number is incremented within the interface module. An entry number of zero will retrieve the first record and the returning entry number will be one. An entry number of seven will retrieve the eighth record and the returning entry number will be eight.
MAXLEN	A halfword containing the length of the I/O area.
I/OAREA	An area in which to place the entry. The data from the entry is actually placed at an offset of two bytes into this area, with the first two bytes serving as a length descriptor. The halfword length field will contain the true length of the specific entry just returned, plus two bytes for the descriptor itself.

Common Sysparms Interface

Besides the TS048 interface to data in any of the parmlib members, a special common interface (DM175) is available for accessing system parameters from member=SYSPARMS. User-written modules can read sysparm values in the same manner as CA-Disk modules by using the common interface. For more information on sysparms, including sysparm names reserved for your use, review Table 24, “System Parameters,” on page 107. The following illustrates reading the VSAMSUPP sysparm.

```
* GET VSAMSUPP SYSTEM PARAMETER
         CALL MI,(DM175,PARMNAME,LEN,PARMVALU,DM175RC),VL

DM175    DC  CL8'ADSDM175'
DM175RC  DC  H'0'
LEN      DC  H'1'
PARMNAME DC  CL8'VSAMSUPP'          name of sysparm desired
PARMVALU DC  CL1' '                 sysparm value returned here
```

The following table helps explain the fields within the Common Interface DM175.

TABLE 156. Field Descriptions within DM175

Field	Description	
DM175	Is always the eight-character name of the common interface module: DM175 DC CL8’ADSDM175’	
PARMNAME	Is the eight-character name of the sysparm whose value is to be returned.	
LEN	Is the halfword field specifying the length (number of bytes) of the output value being returned. It must correspond to the length defined for the PARMVALU field.	
PARMVALU	The area where the value of the sysparm will be returned. Its length must equal the maximum number of characters allowed for the parameter, and agree with the value specified for the LEN parameter.	
DM175RC	The halfword return code from module ADSDM175.	
	0	Indicates that the system parameter was specified and has been returned.
	4	Indicates that it was not specified and the default value has been returned.

System Parameter Validation - Parmedit Member

For editing purposes, each CA-Disk sysparm has a system default value and validation criteria associated with it. Default information indicates the value to be used for each sysparm if you have not specified your installation values in member SYSPARMS of the parmlib data set.

Validation criteria indicate valid values for each sysparm. They are used when CA-Disk loads the sysparm entries into memory. This is done whenever a CA-Disk function is executed in either batch or TSO processing. The value of each sysparm supplied in the SYSPARMS member is checked against the validation criteria specified in the PARMEDIT member. If an improper value is supplied, CA-Disk issues an appropriate message. If the sysparm in error is subsequently accessed, CA-Disk will abend.

One entry is maintained in the PARMEDIT member for each CA-Disk sysparm. The PARMEDIT member is supplied with the CA-Disk system as part of the distributed parmlib data set, and can change with each release.

Although the PARMEDIT member entries can be modified, normally they should not be.

Note: Changes to defaults should be specified in your SYSPARMS member, NOT by changing the PARMEDIT member. The only common need for an update to the PARMEDIT member is to allow or disallow the overriding of a parameter in the input stream. The PARMEDIT member is replaced with each new release of CA-Disk. The SYSPARMS member is user-defined, and not replaced with each release.

Each entry contains the following fields:

TABLE 157. Format of PARMEDIT Entries

Cols	Description		
1-8	System parameter name		
10-12	Minimum number of characters required for parameter value.		
14-16	Maximum number of characters allowed for parameter value.		
18-20	Number of characters to be returned to CA-Disk module requesting the value of the system parameter.		
22-23	Validation Criteria	'Y'	The system parameter value requires a Y or N value indicating either "yes" or "no" for the specified option.
		'N'	
		'NM'	The data for this parameter is of a numeric format. The user must enter numeric data but can enter only significant digits left-justified after the system parameter name.
		''	No data validation will be done on data entered for this parameter value.
25	Enter Y if this system parameter is to be marked obsolete. Blank indicates valid system parameter.		
26	Enter Y if this system parameter can be overridden through the CA-Disk SYSPARMS dd input stream. Enter N if the parameter cannot be overridden.		
28-29	(reserved for future use)		
31-100	The default value to be returned if the system parameter is not supplied by the user.		

Adding User Functions to CA-Disk *ISPF*

There are two ways to add your functions to the CA-Disk ISPF interface. One is to create a static menu selection panel and the other is to add them to the CA-Disk ISPF dynamic menu builder. Before you make any changes, let's examine the structure and purpose of the SPFMENU member of PARMLIB. See the example below.

```
....+....1....+....2....+....3....+....4....+....5....+....6
'SP123    ONLIRPT Y 01  REPORTS - DEFINE YOUR OWN, EXECUTE',
' ONLINE (OPTIONAL)'
'SP026 L  LDSINDX Y 02  LIST - ARCHIVE/BACKUP INDEX ENTRIES'
'SP026 D  DDSINDX Y 02  LIST - (OR DELETE) ARCHIVE/BACKUP ',
'INDEX ENTRIES'
```

Starting in column 2 are the last five characters of the program name (the first three characters of the program must be ADS and are not specified here).

In column 8 is the parameter to be passed to the program. It can be blank (no parameter) or a single character.

The field in columns 11-17 is matched to options specified for each user in SPFOPTNS member. For more information, turn to the topic "Defining ISPF and DSCL User Options" on page 89 in the Installation Guide.

Column 19 is a reserved field and is always Y.

Column 21 identifies the section of the condensed menu to which the function belongs. There are five sections. Keep the numbers in column 21 in ascending order as no validation is performed and unreliable results can occur otherwise.

Beginning in column 25 is the descriptive heading for the function. Lines can be continued and must be enclosed in quotes.

To add entries to the dynamic menu builder, follow these steps.

1. Create a program name that begins with ADS for your function.
2. Enter that name (minus ADS) beginning in column 2.
3. Set the parameter value in column 8.
4. Select a seven-character code for your function and enter it beginning in column 11. Also, you must specify this code in SPFOPTNS member in PARMLIB for the appropriate users.
5. Write your descriptive heading beginning in column 25.

See examples below:

```
....+....1....+....2....+....3....+....4....+....5....+....6
'REST1 Z  ZRTNONE Y 03  RESTORE - FUNCTION 1 '
'BACKP    BACKUPX Y 04  BACKUP DATA SETS '
```

Note: The files and parmlib data sets are allocated by the dynamic menu builder. You can free and reallocate them if needed but be sure the SPFMENU, SPFTEXT and SPFOPTNS members are in the allocated parmlib.

To create a static menu selection panel, follow these steps.

1. Select the functions you want to use in your panel.
2. Invoke them using standard ISPF select statements.

The following are examples from the above SPFMENU example.

```
0,'PGM(ADSSP123)'
1,'PGM(ADSSP026) PARM(L)'
2,'PGM(ADSSP026) PARM(D)'
```

Modifying CA-Disk *ISPF Functions*

When modifying CA-Disk ISPF functions, whether they are JCL/PROCS, panels or parameters, it is best to start with the CA-Disk ISPF dynamic menu builder. The menu builder gives you the name of the CA-Disk module that controls that function (see "Adding User Functions to CA-Disk ISPF" on page 305 for a detailed explanation of the dynamic menu builder).

Within the module are panel names, skeleton JCL names, and required parameters. (To understand these, you must be able to read source code.) A few modules can make calls to other modules to display different panels (that is, ISPF Online Reporting).

For example, if you wished to remove the DELETE action from the archive function, first examine the SPFMENU member in PARMLIB to determine all archive functions. The first module listed for archive is ADSSP001. If you browse the source for ADSSP001, you will find the panel names DMS001A, DMS001B and DMS001C, and skeleton JCL names DMS001A and DMS001B. The DELETE action has no required parameters so the delete action line can be removed.

If, on the other hand, you want to change the PROC names to be executed, it would be much easier to simply scan the ISPSLIB for those PROC names.

To learn more about the IBM product ISPF, consult these IBM manuals: Interactive System Productivity Facility - Dialog Management Services, or Interactive System Productivity Facility (ISPF) and ISPF/Program Development Facility - General Information.

CHAPTER 9

DSCL Processing

During the development of the DSCL support, one of the foremost concerns of the project was to develop a product that is user-friendly. Four major areas needed to be addressed:

- Provide validation of all input commands prior to processing
- Issue meaningful error messages whenever an error is encountered
- Provide the flexibility that the user requires to make DSCL a workable tool
- Provide maximum efficiency in execution time by scanning any catalog only once, regardless of the order of input commands entered by the user.

In an attempt to deal with human factors in the use of the IMPLICIT VSAM support, all commands are validated prior to any action being performed. If any errors are detected, the validation of commands continues, but no action is taken after validation is completed. As each command is read in and validated, an appropriate control block is built (discussed later) containing the pertinent data to the command. As processing continues, the control blocks are linked together to form the processing sequence that will be followed if all validation tests are satisfactory.

The execution order of commands is not necessarily the same order as they are entered by the user — the program performs all of the actions required in a single catalog scan prior to scanning the next catalog/volume (if any). This ensures that a catalog/volume is scanned only once in any given job run, regardless of the number of actions specified by the user. This can be somewhat confusing to the user who might expect results in exactly the same order as which the commands were entered.

Control Block Structure

Six different control blocks are employed by DSCL processing. Five of the control blocks relate directly to the input commands entered by the user. The sixth is the JOB Control Block (JOBCB), which contains informational data regarding the status of the job run. The JOBCB is built during initialization processing in ADSST001, and is linked to all of the other control blocks in the system by the field xxxJOB@ (where xxx is the control block type, such as

CAR or LOC). The JOBCB contains such information as the job return code and the highest DSN expiration date encountered during archival processing. It also contains the highest RETPD specified on all ARCHIVE and BACKUP commands.

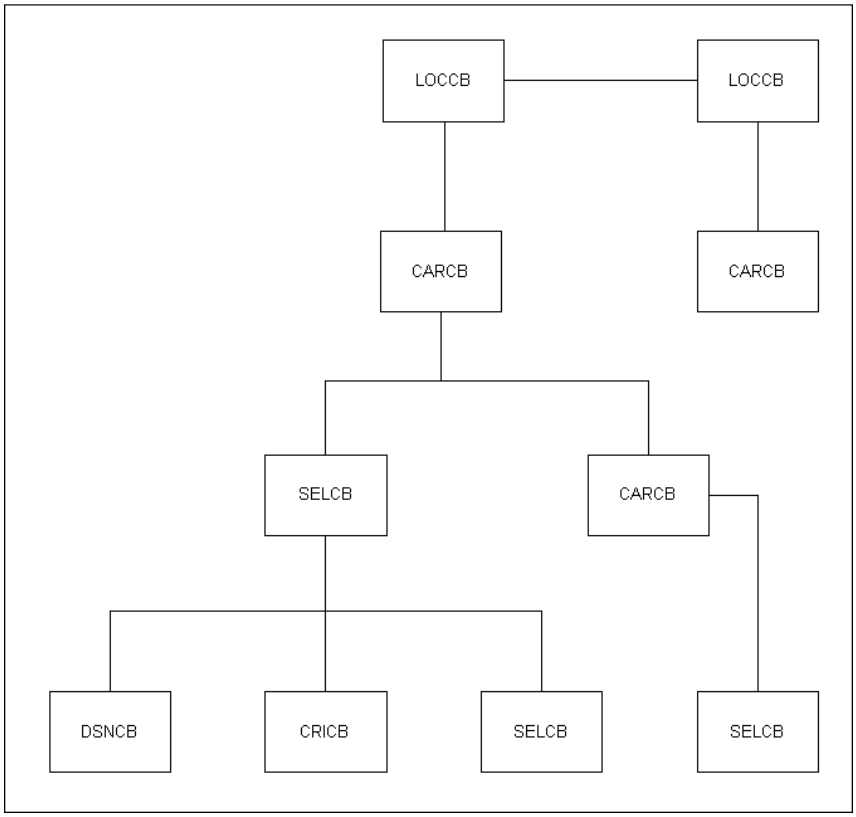
The Locate Control Block (LOCCB) corresponds directly to the SCAN command and is responsible for keeping track of which volumes(s)/catalog(s) to scan. For each SCAN command entered, one LOCCB is built. An exception is made if a SCAN command names the same catalog/volume as a previous SCAN request, such that any subsequent commands are chained off the initial LOCCB for that catalog. The LOCCB also contains any password supplied by the user necessary to open the catalog.

The Command Activation Request Control Block (CARCB) is built for each action command entered (such as BACKUP, ARCHIVE, or DELETE). It is linked directly under the LOCCB and is pointed to by the field LOCDOWN in the LOCCB. This control block informs the program which function to perform on a given catalog in the LOCCB. Multiple actions can be taken on a single catalog/volume scan, which is accomplished by linking multiple CARCBs to each other under a LOCCB. The link is provided by the CARNEXT field in the CARCB. The CARCB contains such information as the action type (DELETE, ARCHIVE or BACKUP), a call list that is executed by ADSST201, and command parameter information.

The Select Control Block (SELCB), illustrated in the Figure below, is built whenever any SELECT or EXCLUDE commands are encountered. It is pointed to by the CARDOWN field in the CARCB. It contains such information as a table name containing DSNs to include or exclude, and a list of volumes to be included or excluded. In addition, it contains a pointer (SELDSN@) to the DSN Control Block (DSNCB) and a pointer (SELDOWN) to the CRIteria Control Block (CRICB). The DSNCB contains a list of data set names entered by the user in the DSN parameter of

the SELECT or EXCLUDE command. The CRICB contains the criteria comparison to be performed as specified in the CRITERIA parameter of the SELECT or EXCLUDE command.

FIGURE 23. Sample Control Block Structure



The following JCL displays typical input commands that would build the structure sample above:

```
SCAN CATALOG=ADS050.AUSERCAT
   SELECT DSN=(ADS.TJP/,ADS.DMS40.PMRFILE),CRITERIA=(TODAY-ARCDT,GE,5)
   EXCLUDE TABLE=SAMPLE
      BACKUP
   SELECT VOLUME=ADS050
      ARCHIVE RETPD=60 SCAN CATALOG=ADS050.SYSTMCAT
      BACKUP
```

CHAPTER 10

Security Processing

Many users are investing in computer hardware, software and security systems that meet the standards documented in the United States Department of Defense Trusted Computer System Evaluation Criteria (commonly called “the Orange Book”), reference number DOD 5200.28-STD.

This document provides a classification scheme for computer systems. The National Computer Security Center (NCSC) currently evaluates together as a Trusted Computer Base (TCB) combinations of computer hardware, and specific releases of operating system and reference monitor (security system). For example, IBM hardware along with MVS/SP 2.2.0, DFP 2.3.0 and RACF 1.8 might be evaluated together for a C2 classification. After evaluation, the NCSC places the system on a Evaluated Product List (EPL).

Many users recognize that any code running APF-authorized can get into storage protection key 0 or the supervisor state, modify any part of memory, issue privileged instructions, bypass any or all parts of MVS system security, and basically do whatever it wants to. Thus the addition of APF-authorized utilities, such as many storage management systems, data base management systems and online transaction processors, can open up such a base system or TCB to exposures. However, at press time, the NCSC does not evaluate such utilities for a statement as to whether they degrade the integrity of the TCB, nor does it place such utilities on any evaluated product list.

Statement of Security Direction

It is our belief that when the security features described on the following pages are active, CA-Disk will not degrade the integrity of computer systems running on hardware, operating systems and reference monitors (security systems) that provide division C1, C2, or B1 protection.

Password protection is in effect only when not overridden by RACF, CA-Top Secret, or other security systems. Also, it is not in effect on DFSMS volumes.

Security Overview

CA-Disk uses a multi-level approach to security. (A graphical representation of this approach follows this text.)

First, no special authorization is required to use CA-Disk — CA-Disk has the same security authorization as any other utility run by any of your users.

Second, CA-Disk uses separate address spaces for separate tasks. Thus, each task can be identified as a unique user (for example Logonid, user ACID or userid) by the security systems installed on your system (for example CA-ACF2, CA-Top Secret, RACF SECURE or ALERT). This provides one complete link between your CA-Disk users and system security.

Third, CA-Disk uses standard system interfaces. Before processing any data in a user dataset, CA-Disk goes through the OPEN SVC of the operating system. Among other things, the OPEN SVC checks authorization with security systems installed on your system, such as CA-ACF2, CA-Top Secret, RACF, SECURE, ALERT and OS- and VSAM-passwords.

To delete, define, catalog, uncatalog or recatalog user datasets, CA-Disk uses the SCRATCH and Catalog Management SVCs. These SVCs also check authorization with your security system(s). This protection is based upon your security systems, such as CA-ACF2, CA-Top Secret, RACF, SECURE, ALERT and OS- and VSAM-passwords.

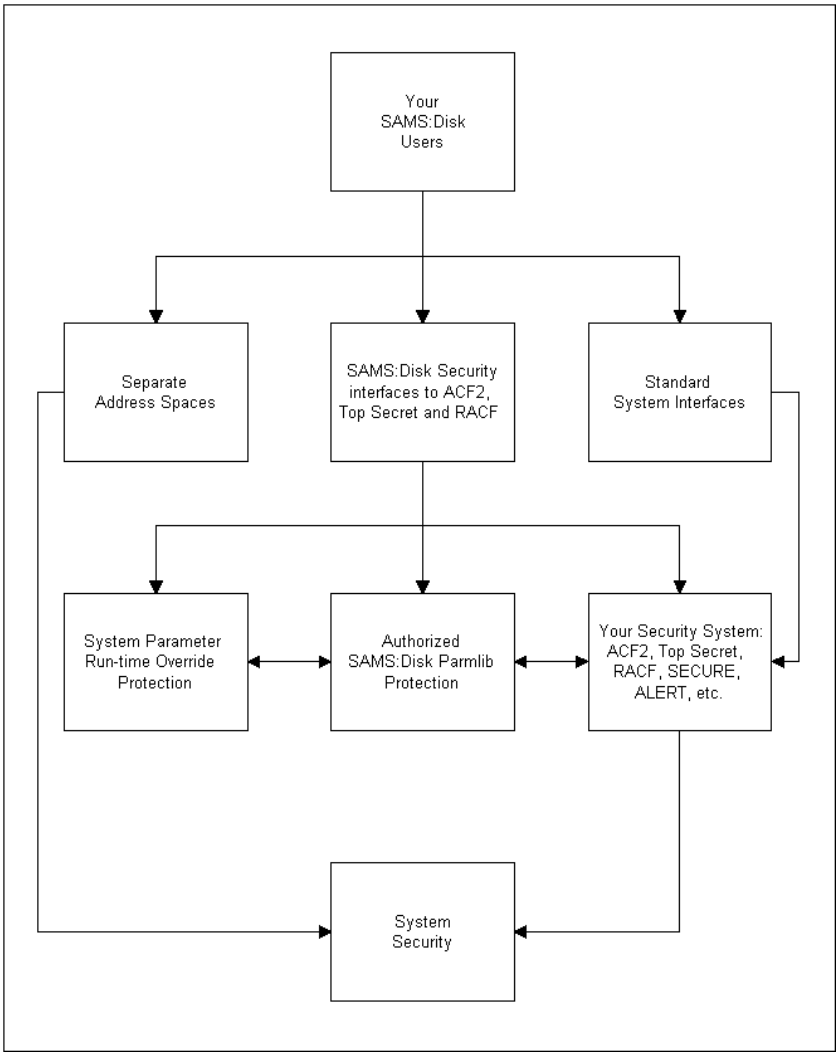
Fourth, CA-Disk provides interfaces to the CA-ACF2, CA-Top Secret and RACF security systems, which are described on the following pages. This protection is based upon setting up system parameter run-time override protection, authorized Parmlib protection and your security system. This provides another complete link between your CA-Disk users and system security.

The integrity of your shop as a whole is, in turn, is based on the functions of your hardware and operating systems, and on the use of a security system to prevent unauthorized update of SYS1.PARMLIB, the authorized CA-Disk Parmlibs, SYS1.NUCLEUS, SYS1.SVCLIB, SYS1.LPALIB and all of your linklist and APF-authorized libraries.

If you have a security system in your shop, you will need to set up your production CA-Disk runs to have sufficient authority to process all datasets managed by CA-Disk. The simplest way to accomplish this is to run these CA-Disk tasks with a Logonid/user ACID/userid that has all authority to all datasets. Depending on your security system, other alternatives can also be possible. Turn to the topic “Activating CA-Disk Security Features and Interfaces” on page 25 in the Installation Guide for a discussion of some of these alternatives. Your individual users can process (archive, restore, migrate, and so on) those datasets for which they are authorized.

The following chart is a graphical representation of the multi-level approach to security used by CA-Disk.

FIGURE 24. Multi-level approach to security



Resources Protected by the CA-Disk Base System

The following resources are specifically protected by the base system of CA-Disk:

1. User datasets on DASD. This protection results from the use of standard system interfaces as described in the Security Overview discussion above.
2. CA-Disk archive datasets on DASD. You can optionally have your security package assign to all users the authority of NONE to all CA-Disk archive datasets on disk. This will prevent utilities other than CA-Disk from reading, modifying or deleting these archives. CA-Disk will take care of its own access to the archives. To determine the dataset names of the CA-Disk archives, see the discussion under "Naming the Archives" on page 61.

Unless you specify CA-Disk sysparm DISKPROT with a value of N, CA-Disk will also password-indicate each archive dataset on disk, but without creating any passwords for these archive datasets on disk. This password-indication will also prevent utilities other than CA-Disk from reading, modifying or deleting these archives. CA-Disk will take care of its own access to the archives. This protection is selectable, and would supplement any protection from your security package.

3. CA-Disk archive datasets on tape. You can optionally have your security package assign to users who need to run backup, archive or merge to tape the authority of WRITE (UPDATE) and to all other users the authority of NONE to all CA-Disk archive datasets on tape. This will prevent utilities other than CA-Disk from reading, modifying or deleting these archives, except by authorized users. If you let CA-Disk sysparm TAPEPRTR default, CA-Disk Restore processing will take care of its own access to the archives. To determine the dataset names of the CA-Disk archives, see the discussion under "Naming the Archives" on page 61.

   If you set sysparm TAPEPROT to a value of Y, CA-Disk will also password-indicate each archive dataset on tape, but will not create any passwords for them. This will also prevent non-authorized users running utilities other than CA-Disk from reading, modifying or deleting the archives. CA-Disk will take care of its own access to the archives. This protection is selectable, and would supplement any protection from your security package.

4. Use of the //PARMLIB dd statement. Combined with the next item and the setting of CA-Disk sysparm SYSPARMO, this feature also protects the setting of CA-Disk sysparms. You can optionally restrict CA-Disk such that CA-Disk will run only when it is using the one authorized parmlib, or one of a list of authorized parmlibs. For details on how to set up this restriction, see "Activating CA-Disk Security Features and Interfaces" on page 25 in the Installation Guide.

5. Use of CA-Disk sysparm overrides through use of the //SYSPARMS dd statement. Use of this dd statement is described in "Overriding Sysparms Instream" on page 106. Users can override sysparms only if SYSPARMO is specified with a value of Y in the SYSPARMS member of your CA-Disk parmlib. For details on how to set up this restriction, see "Activating CA-Disk Security Features and Interfaces" on page 25 in the Installation Guide.

6. Use of the //MTFDEBUG dd statement. For details on how to set up this restriction, see "Activating CA-Disk Security Features and Interfaces" on page 25 in the Installation Guide.

7. CA-Disk ISPF functions. Protection of CA-Disk ISPF functions is documented under "Defining ISPF and DSCL User Options" on page 89 in the Installation Guide.

8. Auto-restore of user datasets. Protection of the CA-Disk auto-restore function is documented under "Installing the Auto-Restore Function" on page 61 in the Installation Guide.

Resources Protected by the Security Interfaces

The following resources are specifically protected by the CA-Disk Security Interfaces:

1. User datasets in the CA-Disk archives. Anyone with access to CA-Disk can access datasets in the CA-Disk archives unless you have installed a CA-Disk Security Interface. The CA-Disk Restore function goes through the OPEN SVC on the target datasets on disk, with OPEN checking authorization with your security package. But CA-Disk cannot go through the OPEN SVC on datasets in the archives, so authorization is not checked with your security package.

   Similarly, in the Deferred Archive and Deferred Restore functions, CA-Disk cannot go through the OPEN SVC on datasets until the Dispose run, so authorization is not checked with your security package. This allows anyone to copy datasets in the archives by using the NEWNAME parameter on the RESTORE command or to archive or restore them in deferred mode, even if they would not otherwise be authorized to copy the datasets.

In addition, anyone with access to CA-Disk can delete (purge) the contents of the CA-Disk archives, unless you have installed a CA-Disk Security Interface. The CA-Disk files dataset contains descriptions of datasets in the archives and of the archives themselves.

You can control the above exposures by installing a CA-Disk Security Interface and reviewing the setting of sysparms SECARVOL and SECFRVOL.

When an user issues a deferred restore, CA-Disk does not know whether or not the dataset will be preallocated at the actual time of the restore. Whether or not the dataset is preallocated will affect the level of authority that CA-Disk checks. The default level of CA-Disk authority that is checked at deferred restore time is OUTPUT. The level of authority checked at the time of the actual restore could be OUTPUT, SCRATCH or ALLOCATE. Sysparm SECURDRS allows the user to specify either OUTPUT, SCRATCH or ALLOCATE as the level of authority CA-Disk will check at the time of the deferred restore. For more detailed information, see the sysparm description for "SECURDRSaaaaaaaa" on page 173.

2. Access to disk volumes. Anyone with access to CA-Disk can have access to disk volumes through the use of the volume-level functions of VBACKUP and VRECOVER, unless a CA-Disk Security Interface is installed and the sysparm SECURVOL defaults to a value of Y. Of special concern is the VRECOVER function, which can write over any volume, unless the CA-Disk security is properly set up.
3. Reporting on user datasets. Anyone with access to CA-Disk can report on user datasets on disk, user datasets in the archives and on the archives themselves unless you have installed a CA-Disk Security Interface. The CA-Disk REPORT functions (except for the PDS directory reports) do not go through the OPEN SVC on user datasets; therefore, authorization is not checked with your security package. You can control this exposure by installing a CA-Disk Security Interface, specifying the CA-Disk sysparm SECURRPT with a value of Y.
4. Auto-restore of user datasets. Protection of the CA-Disk auto-restore function is documented under "Installing the Auto-Restore Function" on page 61 in the Installation Guide.
5. Command names. Anyone with access to CA-Disk can run any CA-Disk command unless you have installed a CA-Disk Security Interface. For example, your installation can choose to allow users to restore individual datasets with the RESTORE command, but allow only the data management group access to recover datasets with the implicit RECOVER command. You can control this exposure by installing a CA-Disk Security Interface, specifying the system parameter SECURCMD with a value of Y, and creating appropriate entries in the SECURDSN member of the parmlib dataset as described in "Security List for CA-Disk Data Set Protection" on page 489.
6. Subfiles of the FILES dataset. Anyone with access to CA-Disk can access any subfile unless you have installed a CA-Disk Security Interface. For instance, your installation can choose to allow users to list information in the DSNINDEX subfile regarding datasets in the archives, but not allow them to list information in the ARCHVOLS subfile regarding the archives themselves. You can control this exposure by installing a CA-Disk Security Interface, specifying the system parameter SECURFIL with a value of Y, and creating appropriate entries in the SECURDSN member of the Parmlib dataset as described in "Security List for CA-Disk Data Set Protection" on page 489.

CA-Disk currently supports interfaces to three security packages: CA-ACF2, CA-Top Secret and RACF. Other security packages can be queried via the user exit SECURPEX.

Resources not Protected by CA-Disk

The following resources are not protected by CA-Disk processing:

1. Auto-restore of user datasets in shops that do not have CA-ACF2, CA-Top Secret or RACF. After an auto-restore completes, OPEN processing prevents user access to datasets to which they are not authorized. Therefore, there is no security hole, but in general anyone can cause the auto-restore of any user dataset archived by CA-Disk except as described above.
2. User datasets moved to tape by the Sequential Migrate function.
3. CA-Disk program names. Some security packages allow you to set special authority rules for datasets when they are accessed through specified programs. In addition, you can have user exits on your operating system that can examine the program name field. This program name is taken from the JCL EXEC PGM parameter for batch functions and from the SPF select PGM parameter for SPF functions. The program name is stored in the CDE control block field CDNAME.

   If you install CA-Disk as documented, all batch and SPF functions will run as the program name ADSMI002. CA-Disk TSO functions will each run with program names equal to their respective command names.
4. The CA-Disk files dataset itself. You will need to grant WRITE (UPDATE) authority to the files dataset to users who run backup or archive functions, who queue archive or restore requests, or who do certain other functions.

   You will need to grant WRITE (UPDATE) authority to the files dataset to users who run the restore or recover functions with non-default values for sysparms RESIXRPD or RETEXCLD.

   You can grant READ authority to the files dataset to users who list the contents of the files dataset, or who run the restore or recover functions with default values for sysparms RESIXRPD and RETEXCLD.
5. The CA-Disk parmlib dataset itself. You will need to grant READ authority to the parmlib dataset to users who run any CA-Disk function. You will need to grant WRITE (UPDATE) authority to the parmlib dataset to users who need to update sysparm settings, exemption tables, etc.
6. The CA-Disk LOADLIB dataset. Since the load library is APF-authorized, you can want to give your users execute or READ authority only to this dataset.
7. Other CA-Disk distribution libraries.
8. CA-Disk work datasets (RACF users see sysparm description for “RACFPDSWn” on page 162).

You should consider these resources as part of your security plan.

Also, note that CA-Disk, even without a security interface, will honor the ADSP attribute for RACF and Top Secret during restore or any other dataset allocation. In this case, if ADSP is turned on, the DS1RACF bit will be set on. CA-Disk will honor the NOADSP attribute as well. If the userid submitting the job has the NOADSP attribute, CA-Disk will maintain the original setting of the DS1RACF bit.

Security and the CA-Disk Auto-Restore Function

As discussed in "Installing the Auto-Restore Function" on page 61 in the Installation Guide, since the CA-Disk auto-restore started task cannot be identified by a JOB statement or a LOGON procedure, most security packages have a Started Procedures Table to associate a started task name with a Logonid/user ACID/userid and possible group name.

If your security package protects resources accessed by started tasks, and you have installed or will install the CA-Disk auto-restore function, and you would like to auto-restore datasets protected by your security package, you must create an entry in your Started Procedures Table to associate the CA-Disk auto-restore started task name DMSAR with a Logonid/user ACID/userid and possible group name.

There is one exception to that last statement — you will probably not have to create an entry if you install the CA-Disk CA-Top Secret or RACF Security Interface, specify sysparm ARSECUREy, and specify neither sysparm SECURFIL nor SECURCMD.

For RACF, the Started Procedures Table is documented in the IBM manual RACF Installation Reference and in the IBM manual Systems Programming Library: RACF.

CA-Disk Security Interface General Information

The CA-Disk Security Interfaces are designed to meet these goals:

1. To provide the same level of security to resources unique to CA-Disk that your security package would normally provide to equivalent resources which, if left unprotected, might be regarded as security exposures.
2. To pre-check authorization. This means that CA-Disk checks authorization before executing a system function, such as OPEN, that will also check authorization. Pre-checking results in two advantages:
   - The messages are often clearer. CA-Disk can issue an authorization-type message rather than a function-type message with return codes and reason codes.
   - Processing is often saved. For instance, if the archive function knows it does not have SCRATCH authority to a user dataset, it can stop the archive before opening the dataset, mounting a tape, backing up the data, etc.
3. To process discrete RACF profiles. Because discrete RACF profiles are deleted when datasets referenced are scratched, special processing is often needed.

   For more details on this goal, see "RACF Discrete Profile Handling" on page 344.
4. To be auditable.

The exposure-closing and pre-checking goals are accomplished by querying your systems security package for approval prior to accessing a resource. In the CA-ACF2 Security Interface, CA-Disk issues the USER CALL-type ACFSVC macro. In the CA-Top Secret and RACF Security Interfaces, CA-Disk issues the RACROUTE REQUEST=AUTH macro.

CA-Disk then examines the return code from your security package. If authorized, CA-Disk continues with the processing of that resource. If not authorized, CA-Disk prints an appropriate message and bypasses that resource. Either way, processing continues to the next resource as required by the CA-Disk commands.

In addition to the CA-Disk authorization failure message, by default each security package issues its own authorization failure message to the system log, the security console, and to the TSO user or job log. It can also create SMF records for a report writer indicating access violations. If you want to inhibit CA-Top Secret or RACF

from issuing their own authorization failure messages or creating SMF records during queries from their respective CA-Disk Security Interfaces running as APF-authorized tasks, specify the sysparm SECURLOG with a value of N.

This last control does not apply to users of the CA-ACF2 Security Interface.

Regardless of the setting of sysparm SECURLOG, each security package will continue to issue its own authorization failure messages for failures occurring outside the CA-Disk Security Interfaces while running CA-Disk functions.

The auditability goal is implemented by CA-Disk internal diagnostic system parameter DIAGAUTH. If this sysparm is specified with a value of Y, CA-Disk will issue additional messages regarding exactly what the CA-Disk Security Interface is passing to your security system, and what your security package returns. For RACF, CA-Disk also prints the profile status of datasets processed, and messages regarding processing of discrete profiles by CA-Disk. This sysparm is very useful in diagnostic situations.

ACF2 trace records are generated when sysparm DIAGAUTH is set to Y.

Running Without a CA-Disk Security Interface

If you run CA-Disk without a Security Interface installed or run with your Security Interface turned off, CA-Disk will not do any authority checking and will not process discrete RACF profiles. You will get the benefits of the authorization checking done by the operating system described in “Security Overview” on page 312, but you will have the exposures listed in the sections entitled “Resources Protected by the Security Interfaces” on page 314 and “Resources not Protected by CA-Disk” on page 316.

Password-Indicated Dataset Support

This topic is presented in “Step 1. Select Password-Indicated data set Support Options” on page 25 in the Installation Guide.

What Authority Does CA-Disk Look For?

For any security protection to be effective, you must have activated the interface as described in “Activating CA-Disk Security Features and Interfaces” on page 25 in the Installation Guide. If you do not have a CA-Disk Security Interface installed, or a resource is not protected in your environment, CA-Disk assumes that it has sufficient authority and will process that resource.

CA-Disk performs authorization checking on volumes and datasets to ensure that you have sufficient authority to those resources. Sufficient authority depends on the function that you are using.

In order to assist you in determining the authority for which CA-Disk will query your security package, we have provided a series of tables. The series contains three sets of tables, one for each of the Security Interfaces currently available: CA-ACF2, CA-Top Secret and RACF. You should use only the set of tables that corresponds to your security package.

Each set of tables contains three types of tables, as follows:

1. Protection of CA-Disk Subfiles

   This table describes the security authority level needed to access subfiles in the files dataset for each CA-Disk function.

2. Protection of CA-Disk Commands

   This table describes the security authority level needed to use the commands for each CA-Disk function.

3. Protection of User Datasets

   This table describes the security authority level needed to access user datasets for each CA-Disk function.

CA-Disk can check authority both to the dataset and its volume. CA-Disk works differently for the ACF2 interface than for the RACF and TOP Secret interfaces. ACF2 will automatically check for volume-level authority while doing dataset authority checking, and make its own determination based on your installation's ACF2 volume and dataset rules. In order to have the RACF and TOP Secret interfaces check for volume level authority, sysparm SECURVOL must default to a value of Y.

Note: The DASDVOL class is checked only when SCRATCH authority is required.

For dataset-level functions, if SECURVOL defaults to a value of Y, CA-Disk checks volume authority first. If volume authority is approved, the function is performed with no further checking. Otherwise CA-Disk continues with dataset authority checking.

For volume-level functions, such as VBACKUP and VRECOVER, CA-Disk only checks for volume-level authority. If volume authority is approved, the function is performed with no further checking. Otherwise, the function is disallowed.

CA-Disk Authority Level Translation Tables for ACF2

In this section are three tables to assist you in determining the level of authority for which CA-Disk will query your ACF2 security package.

The first table, Protection of CA-Disk Subfiles under ACF2, describes the ACF2 authority levels and the CA-Disk internal authority levels needed for access to subfiles in the files dataset.

Please note that security validation will be done on CA-Disk subfiles only if sysparm SECURFIL has been specified with a value of Y and an appropriate entry exists in the parmlib member SECURDSN. If this is not the case, no security validation is done and access is authorized to the subfile.

Down the left-hand side of the table, you will find the CA-Disk functions, and across the top of the table are the names of subfiles within the files dataset. The table indicates the authority level needed to access each subfile for each CA-Disk function. The ACF2 authority level is shown first. The corresponding CA-Disk internal authority level is enclosed in parentheses directly beneath the ACF2 authority level.

Following is an example of how to use this table to determine the security level authorization needed by the ARCHIVE function to access the subfiles used during ARCHIVE processing:

1. Locate the ARCHIVE function in the left-hand column.
2. Read across the column headings to identify the subfiles used by ARCHIVE.

   You will see that CA-Disk will check for the ACF2 authority level of WRITE to access both the DSNINDEX and the ARCHVOLS subfiles. In addition, CA-Disk will check for the ACF2 authority level of READ for the ARCHVOLS subfile. The ACF2 authority level of WRITE translates to the CA-Disk internal authority level of OUTPUT. The ACF2 authority level of READ translates to the CA-Disk internal authority level of INPUT.

Protection of Subfiles Under ACF2

TABLE 158. Protection of Subfiles Under ACF2

Function	DSNINDEX	ARCHVOLS	MIGRECAT	DASDSPCB	ARCHCMDS	RESTCMDS
Archive	Write-output	Write-output Read-input				
Backup	Write-output	Write-output Read-input				
Restore	Write-update Read-input	Read-input				
Auto Restore	Write-update Read-input	Read-input				
Recover	Write-update Read-input	Read-input				
Vbackup	Write-output	Write-output Read-input				
Vrecover	Read-input	Read-input				
Listd	Read-input	Read-input				
Ixmaint	Write-output	Write-output Read-input				
Ixupdate	Write-output	Write-output Read-input				
Merge	Write-output Read-input	Write-output Read-input				
Rebuild	Write-output	Write-output				
Seq. Migr			Write-output			
Billing				Write-output		
Listreq					Read-input	Read-input
Derase					Write-output	Write-output
Format	ALLOC (scratch) on each subfile to be formatted over					
Unload	Read (input) on each subfile					
Reload	Alloc (scratch) on each subfile to be formatted over and Write (output) on each subfile in the unloaded DSNAME					
Reorg	Read (input) on each subfile; ALLOC (scratch) on each subfile; Write (output) on each subfile in the unloaded DSNAME					

Relevant system parameters: - RESIXRPD

Protection of Commands Under ACF2

The following table describes the ACF2 and CA-Disk internal authority levels needed for access to CA-Disk commands.

The table is organized into three columns. The first column lists the CA-Disk functions, the second lists the commands used with that CA-Disk function and the third column specifies the authority level needed for the CA-Disk command. The ACF2 authority level is shown first, followed by the corresponding CA-Disk internal authority level, which is enclosed in parentheses.

Following is an example of how to use this table to determine the security level needed to use the VOLDELETE command:

1. Locate IXMAINT in the Function column.
2. Read across to the Command column to find VOLDELETE.
3. Read across to the Authority Level column to the see that the ACF2 authority level WRITE is needed to use the command.

   This translates to the CA-Disk internal authority level of OUTPUT.

TABLE 159. Protection of Commands Under ACF2

Function	Command	Authority Level
DSCL	[SET], [PASSWORD], [FIND], [SCAN], [SELECT], [EXCLUDE], [REPORT], [ARCHIVE], [BILLING],[DELETE]	Write-output
Archive	[ARCHIVE], [DARCHIVE], [RETAIN]	
Backup	[BACKUP]	
Recover	[RECOVER]	
Vbackup	[VBACKUP]	
Vrecover	[VRECOVER]	
Listd	[LISTD]	
Restore	[RESTORE], [DRESTORE]	
Ixmaint	[DSNDELETE], [VOLDELETE], [PURGE]	
Ixupdate	[DELETE], [CHANGE], [RESET], [BUILDARC]	
Merge	[MERGE]	
Rebuild	[REBUILD]	
Move/Copy	[MOVE], [COPY]	
Idle Space Release	[RELEASE]	
PDS Compress	[SCAN], [COMPRESS], [RELOAD]	
Seq. Migr to Tape	[MODESET], [MIGRATE], [SCAN]	
Delete	[DELETE]	
DASD Space Billing	[BILLING], [ACCUMULATE], [EXTEND]	
Listreq	[LISTREQ]	
Derase	[DERASE]	
Unload files dataset	[UNLOAD]	
Reload files dataset	[RELOAD]	
Reorg files dataset	[RELOAD]	
DSCB Update Utility	[DSCB]	

Note: Security validations will be done on CA-Disk commands only if sysparm SECURCMD is specified with a value of Y and an appropriate entry exists in the parmlib member SECURDSN. If this is not the case, access is authorized for the CA-Disk command.

Protection of User Datasets Under ACF2

The following tables, Protection of CA-Disk Datasets Under ACF2, describe both the ACF2 authority levels and the CA-Disk internal authority levels needed for access to datasets and their volumes. Each CA-Disk function appears in a separate table.

Each function table is organized into two major columns. The first column lists the CA-Disk functions. The second column shows the authority level needed to access the volume or dataset. Depending on the CA-Disk function, the second column is further divided into command parameters that can affect the authorization level needed to access the resource. The ACF2 authority level is shown first, followed by the corresponding CA-Disk internal authority level, which is enclosed in parentheses. The words no check are used if there is to be no security checking done.

While doing the authorization level check on the dataset, ACF2 will also check for volume level authorization. ACF2 uses the volume and dataset rules set up in your installation to determine authority to the dataset.

Following is an example of how to use the table to determine the security level authorization needed for RESTORE processing:

The sample RESTORE command is:

```
RESTORE DSNAME=SBKSV.DATA,NEWNAME=SBKSV.DATA.NEW,SCRATCH
```

The target dataset, SBKSV.DATA.NEW, is preallocated. Sysparm SECURVOL is set to Y.

1. Find the page of the table describing the RESTORE function.

   To find the ACF2 security level authorization needed to access SBKSV.DATA, first locate the dataset to be restored section in the Resource column.

2. Read across to the authority level columns. They show that the ACF2 authority levels of READ and WRITE are needed to access both the dataset and volume, if the SCRATCH and NEWNAME parameters are specified on the command.

   To find the ACF2 security level authorization needed to access SBKSV.DATA.NEW, first locate the target dataset preallocated section in the Resource column.

3. Read across to the authority level columns. They show that the ACF2 authority level of ALLOC is needed if the SCRATCH parameter is specified on the command. If the SCRATCH and NEWNAME parameters are specified on the command, ACF2 authority levels of ALLOC and WRITE are needed to access SBKSV.DATA.NEW.

Protection of User Datasets Under ACF2 — Reports

Resource to be checked	Authority level
Dataset to be reported on	Read-report

Protection of User Datasets Under ACF2 — Archive

Resource to be checked	Authority level		
	With any command	CATALOG action specified	NEWNAME parm specified
Dataset to be archived	Read-input; ALLOC-scratch	Read-input; ALLOC-scratch	Read-rename from
Cluster to be archived	Read-input; ALLOC-scratch & catalog	Read-input; ALLOC-scratch & catalog	Read-rename from
New dataset name	no check	no check	Write-rename to

Protection of User Datasets Under ACF2 — Backup

Resource to be checked	Authority level
Dataset to be backed up	Read-input

Protection of User Datasets Under ACF2 — Restore/Recover

Resource to be checked	Authority level		
	Scratch parm specified	Scratch not specified	NEWNAME parm specified
Dataset to be restored	Write-output	Write-output	Read-rename from
Target dataset preallocated	ALLOC-scratch	Write-output	Write-rename to
Target dataset not preallocated	ALLOC-alloc	ALLOC-alloc	Write-rename to

Relevant system parameters: - SECURDRS

Protection of User Datasets Under ACF2 — LISTD

Resource to be checked	Authority level
Dataset to be reported on	Read-report

Protection of User Datasets Under ACF2 — Ixmaint/Ixupdate

Resource to be checked	Authority level
Dataset to be scratched from CA-Disk archives	ALLOC-scratch

Protection of User Datasets Under ACF2 — MERGE

Resource to be checked	Authority level
Dataset to be scratched from CA-Disk archives	Write-output
Dataset to be overlaid in the CA-Disk archives	Write-output

Protection of User Datasets Under ACF2 — MOVE/COPY

	Authority level	
Resource to be checked	**With any command**	**NEWNAME parm specified**
Dataset to be restored	Write-output	Read-rename from
Dataset to be moved	ALLOC-scratch	Read-rename from
Target dataset preallocated	Write-output	Write-rename to
Target dataset not preallocated	no check	Write-rename to

Protection of User Datasets Under ACF2 — RELEASE

Resource to be checked	Authority level
Dataset to have space released	Write-output

Protection of User Datasets Under ACF2 — PDS Compress

	Authority level		
Resource to be checked	**USE parm specified**	**USE parm not specified**	**Reload command specified**
Dataset to be compressed	Read-input; Write-input	Read-input; Write-output; ALLOC-scratch	no check
Dataset to be reloaded	no check	no check	Write-output

Protection of User Datasets Under ACF2 — SEQ MIGRATE

	Authority level	
Resource to be checked	**Noscratch parm specified**	**Noscratch parm not specified**
Dataset to be migrated	Read-input	Read-input; ALLOC-scratch

Protection of User Datasets Under ACF2 — Delete

Resource to be checked	Authority level
Dataset to be scratched	ALLOC-scratch

Protection of User Datasets Under ACF2 — Listreq

Resource to be checked	Authority level
Dataset to be reported on	READ-report

Protection of User Datasets Under ACF2 — Derase

Resource to be checked	Authority level
Dataset to have request deleted	ALLOC-scratch

Protection of User Datasets Under ACF2 — DSCB Update

Resource to be checked	Authority level
Dataset to be processed	Write-output; ALLOC-scratch

Protection of User Datasets Under ACF2 — Vbackup

Resource to be checked	Authority level
Volume to be backed up	Read-input

Protection of User Datasets Under ACF2 — Vrecover

Resource to be checked	Authority level
Volume to be recovered	Read-input
Target volume	ALLOC-scratch; Write-output
IDVOL for CLIP	ALLOC-scratch; Write-output

Functions that Need no Dataset/Volume Authorization

The following functions do not actually access any user datasets, so no security authorization checking is done at the dataset/volume level. However they do use commands and do access the files dataset, so security authorization checking is done only at the files dataset subfile level and the CA-Disk command level.

- Format files dataset
- Unload files dataset
- Reload files dataset
- Reorg files dataset

Authority Level Translation Tables for Top Secret

In this section are three tables to assist you in determining the level of authority for which CA-Disk will query your Top Secret security package.

The first table, Protection of CA-Disk Subfiles under TOP SECRET, describes the Top Secret authority levels and the CA-Disk internal authority levels needed for access to subfiles in the files dataset.

Please note that security validation will be done on CA-Disk subfiles only if sysparm SECURFIL has been specified with a value of Y and an appropriate entry exists in the parmlib member SECURDSN. If this is not the case, no security validation is done and access is authorized to the subfile.

Down the left-hand side of the table, you will find the CA-Disk functions, and across the top of the table are the names of subfiles within the files dataset. The table indicates the authority level needed to access each subfile for each CA-Disk function. The Top Secret authority level is shown first. The corresponding CA-Disk internal authority level is enclosed in parentheses directly beneath the Top Secret authority level.

Following is an example of how to use this table to determine the security level authorization needed by the ARCHIVE function to access the subfiles used during ARCHIVE processing:

1. Locate the ARCHIVE function in the left-hand column.
2. Read across the column headings to identify the subfiles used by ARCHIVE.

   You will see that CA-Disk will check for the Top Secret authority level of WRITE to access both the DSNINDEX and the ARCHVOLS subfiles. In addition, CA-Disk will check for the TOP SECRET authority level of READ for the ARCHVOLS subfile. The Top Secret authority level of WRITE translates to the CA-Disk internal authority level of OUTPUT. The Top Secret authority level of READ translates to the CA-Disk internal authority level of INPUT.

Protection of Subfiles Under Top Secret

TABLE 160. Protection of Subfiles Under Top Secret

Function	DSNINDEX	ARCHVOLS	MIGRECAT	DASDSPCB	ARCHCMDS	RESTCMDS
Archive	Write-output	Write-output Read-input				
Backup	Write-output	Write-output Read-input				
Restore	Write-update Read-input	Read-input				
Auto Restore	Write-update Read-input	Read-input				
Recover	Write-update Read-input	Read-input				
Vbackup	Write-output	Write-output Read-input				
Vrecover	Read-input	Read-input				
Listd	Read-input	Read-input				
Ixmaint	Write-output	Write-output Read-input				
Ixupdate	Write-output	Write-output Read-input				
Merge	Write-output Read-input	Write-output Read-input				
Rebuild	Write-output	Write-output				
Seq. Migr			Write-output			
Billing				Write-output		
Listreq					Read-input	Read-input
Derase					Write-output	Write-output
Format	SCRATCH (scratch) on each subfile to be formatted over					
Unload	Read (input) on each subfile					
Reload	SCRATCH (scratch) on each subfile to be formatted over and Write (output) on each subfile in the unloaded DSNAME					
Reorg	Read (input) on each subfile; SCRATCH (scratch) on each subfile; Write (output) on each subfile in the unloaded DSNAME					

Relevant system parameters: - RESIXRPD

Protection of Commands Under Top Secret

The following table, Protection of CA-Disk Commands under TOP SECRET, describes the Top Secret and CA-Disk internal authority levels needed for access to CA-Disk commands.

Please note that security validation will be done on CA-Disk commands only if sysparm SECURCMD is specified with a value of Y and an appropriate entry exists in the parmlib member SECURDSN. If this is not the case, access is authorized for the CA-Disk command.

The table is organized into three columns. The first column lists the CA-Disk functions, the second lists the commands used with that CA-Disk function and the third column specifies the authority level needed for the CA-Disk command. The Top Secret authority level is shown first, followed by the corresponding CA-Disk internal authority level, which is enclosed in parentheses.

Following is an example of how to use this table to determine the security level authorization needed to use the VOLDELETE command for the IXMAINT function.

1. Locate IXMAINT in the Function column on the next page.
2. Read across to the Command column and find the VOLDELETE command.
3. Then read across to the Authority Level column to the see that the TOP SECRET authority level WRITE is needed to use the command.

   This translates to the CA-Disk internal authority level of OUTPUT.

TABLE 161. Protection of Commands Under Top Secret

Function	Command	Authority Level
DSCL	[SET], [PASSWORD], [FIND], [SCAN], [SELECT], [EXCLUDE], [REPORT], [ARCHIVE], [BILLING],[DELETE]	Write-output
Archive	[ARCHIVE], [DARCHIVE], [RETAIN]	
Backup	[BACKUP]	
Recover	[RECOVER]	
Vbackup	[VBACKUP]	
Vrecover	[VRECOVER]	
Listd	[LISTD]	
Restore	[RESTORE], [DRESTORE]	
Ixmaint	[DSNDELETE], [VOLDELETE], [PURGE]	
Ixupdate	[DELETE], [CHANGE], [RESET], [BUILDARC]	
Merge	[MERGE]	
Rebuild	[REBUILD]	
Move/Copy	[MOVE], [COPY]	
Idle Space Release	[RELEASE]	
PDS Compress	[SCAN], [COMPRESS], [RELOAD]	
Seq. Migr to Tape	[MODESET], [MIGRATE], [SCAN]	
Delete	[DELETE]	
DASD Space Billing	[BILLING], [ACCUMULATE], [EXTEND]	
Listreq	[LISTREQ]	
Derase	[DERASE]	
Unload files dataset	[UNLOAD]	
Reload files dataset	[RELOAD]	
Reorg files dataset	[RELOAD]	
DSCB Update Utility	[DSCB]	

Protection of User Datasets Under Top Secret

The following tables, Protection of CA-Disk Datasets Under Top Secret, describe both the Top Secret authority levels and the CA-Disk internal authority levels needed for access to datasets and their volumes. Each CA-Disk function appears in a separate table.

Each function table is organized into three major columns. The first column lists the CA-Disk functions. The second column shows the breakdown between volume and dataset level security checks. The third column shows the authority level needed to access the volume or dataset. Depending on the CA-Disk function, the third column is further divided into command parameters that can affect the authorization level needed to access the resource. The Top Secret authority level is shown first, followed by the corresponding CA-Disk internal authority level, which is enclosed in parentheses. The words no check are used if there is to be no security checking done.

Please note that CA-Disk will check access to a volume only when sysparm SECURVOL is specified with a value of Y. When volume-level authorization checking is done, it will always precede the dataset-level check. If access to the volume is authorized, the dataset-level authorization checking will be bypassed. Dataset-level authorization checking is therefore done only if any of these are true:

1. sysparm SECURVOL is specified with a value of N
2. the volume is not protected by your security interface
3. access to the volume is denied

Following is an example of how to use the table to determine the security level authorization needed for RESTORE processing:

The sample RESTORE command is:

```
RESTORE DSNAME=SBKSV.DATA,NEWNAME=SBKSV.DATA.NEW,SCRATCH
```

The target dataset, SBKSV.DATA.NEW, is preallocated. Sysparm SECURVOL is set to Y.

1. Find the page of the table describing the RESTORE function.

   To find the Top Secret security level authorization needed to access SBKSV.DATA, first locate the dataset to be restored section in the Resource column.

2. Read across the volume-level check line to the authority level columns. They show that the Top Secret authority level of WRITE is needed to access the volume, if the SCRATCH and NEWNAME parameters are specified on the command.

   If the volume security check fails or is not done, read across the dataset level line to the authority level columns. They show that if the SCRATCH and NEWNAME parameters are specified on the command, Top Secret authority levels of WRITE and READ are needed to access SBKSV.DATA.

3. To find the Top Secret security level authorization needed to access SBKSV.DATA.NEW, first locate the target dataset preallocated section in the Resource column.

4. Read across the volume level check line to the authority level columns. They show that the Top Secret authority level of ALL is needed if the SCRATCH parameter is specified on the command.

   If the volume security check fails or is not done, and if the SCRATCH and NEWNAME parameters are specified on the command, Top Secret authority levels of SCRATCH and WRITE are needed to access SBKSV.DATA.NEW.

Protection of User Datasets Under Top Secret — Reports

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be reported on	volume level check (of SECURVOL “Y”)	Read-report
	dataset level check (done if vol check fails or not done)	Read-report

Protection of User Datasets Under Top Secret — Archive

		Authority level		
Resource to be checked	Volume or Dataset level check	With any command	Catalog action specified	Newname parm specified
Dataset to be archived	Volume level check (if SECURVOL “Y”)	Read-input; ALLOC-scratch	Read-input; ALLOC-scratch	Read-rename from
	Dataset level check (done if vol check fails or not done)	Read-input; Scratch-scratch	Read-input; Scratch-scratch	Read-rename from
Cluster to be archived	Volume level check (if SECURVOL “Y”)	Read-input; ALLOC-scratch & catalog	Read-input; ALLOC-scratch & catalog	Read-rename from
	Dataset level check (done if vol check fails or not done)	Read-input; Scratch-scratch; Create-catalog	Read-input; Scratch-scratch; Create-catalog	Read-rename from
New dataset name	Volume level check (if SECURVOL “Y”)	no check	no check	Write-rename to
	Dataset level check (done if vol check fails or not done)	no check	no check	Write-rename to

Protection of User Datasets Under Top Secret — Backup

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be backed up	volume level check (of SECURVOL “Y”)	Read-input
	dataset level check (done if vol check fails or not done)	Read-input

Protection of User Datasets Under Top Secret — Restore

		Authority level		
Resource to be checked	Volume or Dataset level check	Scratch parm specified	Scratch parm not specified	Newname parm specified
Dataset to be restored	Volume level check (if SECURVOL “Y”)	Write-output	Write-output	Read-rename from
	Dataset level check (done if vol check fails or not done)	Write-output	Write-output	Read-rename from
Target dataset preallocated	Volume level check (if SECURVOL “Y”)	ALLOC-scratch	Write-output	Write-rename to
	Dataset level check (done if vol check fails or not done)	Scratch-scratch	Write-output	Write-rename to

Resource to be checked	Volume or Dataset level check	Authority level: Scratch parm specified	Authority level: Scratch parm not specified	Authority level: Newname parm specified
Target dataset not preallocated	Volume level check (if SECURVOL “Y”)	ALLOC-alloc	ALLOC-alloc	Write-rename to
	Dataset level check (done if vol check fails or not done)	Create-alloc	Create-alloc	Write-rename to

Relevant system parameters: - SECURDRS

Protection of User Datasets Under Top Secret — LISTD

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be reported on	volume level check (of SECURVOL “Y”)	Read-report
	dataset level check (done if vol check fails or not done)	Read-report

Protection of User Datasets Under Top Secret — IXMAINT

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be scratched from the CA-Disk archives	volume level check (of SECURVOL “Y”)	Write-output
	dataset level check (done if vol check fails or not done)	Write-output

Protection of User Datasets Under Top Secret — MERGE

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be scratched from the CA-Disk archives	volume level check (of SECURVOL “Y”)	Write-output
	dataset level check (done if vol check fails or not done)	Write-output

Protection of User Datasets Under Top Secret — REBUILD

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be overlaid in the CA-Disk archives	volume level check (of SECURVOL “Y”)	Write-output
	dataset level check (done if vol check fails or not done)	Write-output

Protection of User Datasets Under Top Secret — MOVE/COPY

Resource to be checked	Volume or Dataset level check	Authority level: With any command	Authority level: Newname parm specified
Dataset to be copied	Volume level check (if SECURVOL “Y”)	Write-output	Read-rename from
	Dataset level check (done if vol check fails or not done)	Write-output	Read-rename from
Dataset to be moved	Volume level check (if SECURVOL “Y”)	ALLOC-scratch	Read-rename from
	Dataset level check (done if vol check fails or not done)	Scratch-scratch	Read-rename from
Target dataset preallocated	Volume level check (if SECURVOL “Y”)	Write-output	Write-rename to
	Dataset level check (done if vol check fails or not done)	Write-output	Write-rename to
Target dataset not preallocated	Volume level check (if SECURVOL “Y”)	no check	Write-rename to
	Dataset level check (done if vol check fails or not done)	no check	Write-rename to

Protection of User Datasets Under Top Secret — RELEASE

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to have space released	volume level check (of SECURVOL “Y”)	Write-output
	dataset level check (done if vol check fails or not done)	Write-output

Protection of User Datasets Under Top Secret — Compress

Resource to be checked	Volume or Dataset level check	Authority level: Use parm specified	Authority level: Use parm not specified	Authority level: Reload command specified
Dataset to be compressed	Volume level check (if SECURVOL “Y”)	Read-input; Write-output	Read-input; Write-output; ALLOC-scratch	no check
	Dataset level check (done if vol check fails or not done)	Read-input; Write-output	Read-input; Write-output; Scratch-scratch	no check
Dataset to be reloaded	Volume level check (if SECURVOL “Y”)	no check	no check	Write-output
	Dataset level check (done if vol check fails or not done)	no check	no check	Write-output

Protection of User Datasets Under Top Secret — Sequential Migrate

Resource to be checked	Volume or Dataset level check	Authority level: Noscratch parm specified	Authority level: Noscratch parm not specified
Dataset to be migrated	Volume level check (if SECURVOL “Y”)	Read-input	Read-input; ALLOC-scratch
	Dataset level check (done if vol check fails or not done)	Read-input	Read-input; ALLOC-scratch

Protection of User Datasets Under Top Secret — Delete

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be scratched	volume level check (of SECURVOL “Y”)	ALLOC-scratch
	dataset level check (done if vol check fails or not done)	Scratch-scratch

Protection of User Datasets Under Top Secret — LISTREQ

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be reported on	volume level check (of SECURVOL “Y”)	Read-report
	dataset level check (done if vol check fails or not done)	Read-report

Protection of User Datasets Under Top Secret — DERASE

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to have request deleted	volume level check (of SECURVOL “Y”)	ALLOC-scratch
	dataset level check (done if vol check fails or not done)	Scratch-scratch

Protection of User Datasets Under Top Secret — DSCB Updte

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be processed	volume level check (of SECURVOL “Y”)	Write-output; ALLOC-scratch
	dataset level check (done if vol check fails or not done)	Write-output; Scratch-scratch

Protection of User Datasets Under Top Secret — VBACKUP

Resource to be checked	Authority level
Dataset to be backed up	READ-input

Protection of User Datasets Under Top Secret — VRECOVER

Resource to be checked	Authority level
Volume to be recovered	READ-input
Target volume	Scratch-scratch; Write-output
IDVOL for CLIP	Scratch-scratch; Write-output

Functions that Need no Dataset/Volume Authorization

The following functions do not actually access any user datasets, so no security authorization checking is done at the dataset/volume level. However they do use commands and do access the files dataset, so security authorization checking is done only at the files dataset subfile level and the CA-Disk command level.

- Format files dataset
- Unload files dataset
- Reload files dataset
- Reorg files dataset

Authority Level Translation Tables for RACF

In this section are three tables to assist you in determining the level of authority for which CA-Disk will query your RACF security package.

The first table, Protection of CA-Disk Subfiles under RACF, describes the RACF authority levels and the CA-Disk internal authority levels needed for access to subfiles in the files dataset.

Please note that security validation will be done on CA-Disk subfiles only if sysparm SECURFIL has been specified with a value of Y and an appropriate entry exists in the parmlib member SECURDSN. If this is not the case, no security validation is done and access is authorized to the subfile.

Down the left-hand side of the table, you will find the CA-Disk functions, and across the top of the table are the names of subfiles within the files dataset. The table indicates the authority level needed to access each subfile for each CA-Disk function. The RACF authority level is shown first. The corresponding CA-Disk internal authority level is enclosed in parentheses directly beneath the RACF authority level.

Following is an example of how to use this table to determine the security level authorization needed by the ARCHIVE function to access the subfiles used during ARCHIVE processing:

1. Locate the ARCHIVE function in the left-hand column.
2. Read across the column headings to identify the subfiles used by ARCHIVE.

   You will see that CA-Disk will check for the RACF authority level of UPDATE to access both the DSNINDEX and the ARCHVOLS subfiles. In addition, CA-Disk will check for the RACF authority level of READ for the ARCHVOLS subfile. The RACF authority level of UPDATE translates to the CA-Disk internal authority level of OUTPUT. The RACF authority level of READ translates to the CA-Disk internal authority level of INPUT.

Protection of Subfiles Under RACF

TABLE 162. Protection of Subfiles Under RACF

Function	DSNINDEX	ARCHVOLS	MIGRECAT	DASDSPCB	ACHCMDS	RSTCMDS
Archive	Update-output	Update-output; Read-input				
Backup	Update-output	Update-output; Read-input				
Restore	Write-update; Read-input	Read-input				
Auto Restore	Write-update; Read-input	Read-input				
Recover	Write-update; Read-input	Read-input				
Vbackup	Update-output	Update-output; Read-input				
Vrecover	Read-input	Read-input				
Listd	Read-input	Read-input				
Ixmaint	Update-output	Update-output; Read-input				
Ixupdate	Update-output	Update-output; Read-input				
Merge	Update-output; Read-input	Update-output; Read-input				
Rebuild	Update-output	Update-output				
Seq. Migr			Update-output			
Billing				Update-output		
Listreq					Read-input	Read-input
Derase					Update-output	Update-output
Format	ALTER (scratch) on each subfile to be formatted over					
Unload	Read (input) on each subfile					
Reload	ALTER (scratch) on each subfile to be formatted over and Update (output) on each subfile in the unloaded DSNAME					
Reorg	Read (input) on each subfile; ALTER (scratch) on each subfile; Update (output) on each subfile in the unloaded DSNAME					
Racfchk1	Read (input) on each subfile					

Relevant system parameters: - RESIXRPD

Protection of Commands Under RACF

The following table, Protection of CA-Disk Commands under RACF, describes the RACF and CA-Disk internal authority levels needed for access to CA-Disk commands.

Please note that security validation will be done on CA-Disk commands only if sysparm SECURCMD is specified with a value of Y and an appropriate entry exists in the parmlib member SECURDSN. If this is not the case, access is authorized for the CA-Disk command.

The table is organized into three columns. The first column lists the CA-Disk functions, the second lists the commands used with that CA-Disk function and the third column specifies the authority level needed for the CA-Disk command. The RACF authority level is shown first, followed by the corresponding CA-Disk internal authority level, which is enclosed in parentheses.

Following is an example of how to use this table to determine the security level authorization needed to use the VOLDELETE command for the Archive Management function:

1. Locate Archive Management in the Function column.
2. Read across to the Command column and find the VOLDELETE command.
3. Then read across to the Authority Level column to the see that the RACF authority level UPDATE is needed to use the command.

   This translates to the CA-Disk internal authority level of OUTPUT.

TABLE 163. Protection of Commands Under RACF

Function	Command	Authority Level
DSCL	[SET], [PASSWORD], [FIND], [SCAN], [SELECT], [EXCLUDE], [REPORT], [ARCHIVE], [BILLING], [DELETE], [BACKUP], [COPY], [MOVE], [RELEASE], [VBACKUP]	Update-output
SMF Report	[SELECT]	Update-output
Explicit Archive	[ARCHIVE], [DARCHIVE], [MODESET], [PASSWORD]	Update-output
Recover/Restore	[DRESTORE], [PASSWORD], [RESTORE], [RECOVER], [VRECOVER]	Update-output
Deferred Archive/Restore	[DERASE]	Update-output
Deferred Archive/Restore	[LISTREQ]	Read-input
PDS Compress	[SCAN], [COMPRESS], [RELOAD]	Update-output
Seq. Migr to Tape	[MODESET], [MIGRATE], [SCAN], [RECATALOG]	Update-output
DASD Space Billing	[BILLING], [ACCUMULATE], [EXTEND], [SELECT], [RENAME]	Update-output
Archive Management	[ANALYZE], [BUILDARC], [CHANGE], [COPY], [DELETE], [DSNDELETE], [EXTRACT], [LISTD], [LISTV], [MERGE], [PURGE], [REBUILD], [RELOAD], [RESET], [TAPEFIX], [TRECOVER], [UNLOAD], [VALIDATE], [VOLDELETE]	Update-output
Files Management	[FILEDUMP], [FRECOVER], [LOADLIB], [PROCESS], [REPARCH], [RETAPE]	Update-output
Other diagnostics	[DUMP], [RMAP]	Update-output
Install product	[CATALOG], [DSCB], [SET]	Update-output
Tapepool Utilities	[ADD], [DELETE], [ERASE], [LIST]	Update-output
Threshold Manager	[EXCLUDE=], [GLOBAL=], [INCLUDE=], [MULTIVOL=], [POOLDEF=]	Not checked

TABLE 163. Protection of Commands Under RACF

Function	Command	Authority Level
RACF Utilities	[CHECK], [CREATE], [DELETE], [DISPLAY], [RENAME], [SELECT]	Update-output
Non documented Utilities	[BLDL], [COPY], [DISPLAY], [DISPOSE], [DMSGLOBAL], [INIT], [MIGRATE], [PASSWORD], [REPORT], [RETAIN], [SCAN], [SELECT], [SEARCH], [VREPORT]	Update-output

Protection of User Datasets Under RACF

The following tables, Protection of CA-Disk Datasets Under RACF, describe both the RACF authority levels and the CA-Disk internal authority levels needed for access to datasets and their volumes. Each CA-Disk function appears in a separate table.

Each function table is organized into three major columns. The first column lists the CA-Disk functions. The second column shows the breakdown between volume and dataset level security checks. The third column shows the authority level needed to access the volume or dataset. Depending on the CA-Disk function, the third column is further divided into command parameters that can affect the authorization level needed to access the resource. The RACF authority level is shown first, followed by the corresponding CA-Disk internal authority level, which is enclosed in parentheses. The words no check are used if there is to be no security checking done.

Please note that CA-Disk will check access to a volume only when sysparm SECURVOL is specified with a value of Y. When volume-level authorization checking is done (SCRATCH action only), it will always precede the dataset-level check. If access to the volume is authorized, the dataset-level authorization checking will be bypassed. Dataset-level authorization checking is therefore done only if any of these are true:

1. sysparm SECURVOL is specified with a value of N
2. the volume is not protected by your security interface
3. access to the volume is denied

Following is an example of how to use the table to determine the security level authorization needed for RESTORE processing:

The sample RESTORE command is:

```
RESTORE DSNAME=SBKSV.DATA,NEWNAME=SBKSV.DATA.NEW,SCRATCH
```

The target dataset, SBKSV.DATA.NEW, is preallocated. Sysparm SECURVOL is set to Y.

1. Find the page of the table describing the RESTORE function.

   To find the RACF security level authorization needed to access SBKSV.DATA, first locate the dataset to be restored section in the Resource column.

2. Read across the volume-level check line to see that no volume level checking is done. Since no volume check is done, read across the dataset-level line to the authority level columns. They show that if the SCRATCH and NEWNAME parameters are specified on the command, RACF authority levels of UPDATE and READ are needed to access SBKSV.DATA.

   To find the RACF security level authorization needed to access SBKSV.DATA.NEW, first locate the target dataset preallocated section in the Resource column. Read across the volume level check line to the authority level columns. They show that the RACF authority level of ALTER is needed if the SCRATCH parameter is

specified on the command. If the volume security check fails or is not done, and if the SCRATCH and NEWNAME parameters are specified on the command, RACF authority levels of ALTER and UPDATE are needed to access SBKSV.DATA.NEW.

Protection of User Datasets Under RACF — Reports

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be reported on	volume level check (of SECURVOL “Y”)	no check
	dataset level check (done if vol check fails or not done)	Read-report; Control-control

Relevant RACF system parameters: - SECURRPT and VSACCESS

Protection of User Datasets Under RACF — Archive

		Authority level		
Resource to be checked	**Volume or Dataset level check**	**With each command**	**Catalog action specified**	**Newname parm specified**
Dataset to be archived	Volume level check (if SECURVOL “Y”)	Alter-scratch	Alter-scratch	no check
	Dataset level check (done if vol check fails or not done)	Read-input; Alter-scratch	Read-input; Alter-scratch	no check
Cluster to be archived	Volume level check (if SECURVOL “Y”)	Alter-scratch	no check	no check
	Dataset level check (done if vol check fails or not done)	Read-input or Control-control plus Alter-scratch &catalog)	Read-input or Control-control plus Alter-scratch &catalog)	no check
New dataset name	Volume level check (if SECURVOL “Y”)	no check	no check	no check
	Dataset level check (done if vol check fails or not done)	no check	no check	Update-rename to

Relevant RACF system parameters: - RACFBKUP, RACFNEWN, and VSACCESS

Protection of User Datasets Under RACF — Backup

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be backed up	volume level check (of SECURVOL “Y”)	no check
	dataset level check (done if vol check fails or not done)	Read-input or Control-control

Relevant RACF system parameters: - RACFBKUP and VSACCESS

Protection of User Datasets Under RACF — Restore

		Authority level		
Resource to be checked	**Volume or Dataset level check**	**Scratch parm specified**	**Scratch parm not specified**	**Newname parm specified**
Dataset to be restored	Volume level check (if SECURVOL “Y”)	no check	no check	no check
	Dataset level check (done if vol check fails or not done)	Update-output or Control-control	Update-output or Control-control	Read (input) if RACFNEWNy (rename from)
Target dataset preallocated	Volume level check (if SECURVOL “Y”)	Alter-scratch	no check	no check
	Dataset level check (done if vol check fails or not done)	Alter-scratch	Update-output	Update (output) if RACFNEWNy (rename to)
Target dataset not preallocated (checks preexisting, non-CA-Disk saved, profiles only)	Volume level check (if SECURVOL “Y”)	no check	no check	no check
	Dataset level check (done if vol check fails or not done)	based on value of sysparm RACFALLO which defaults to Alter (alloc)	based on value of sysparm RACFALLO which defaults to Alter (alloc)	Based on value of sysparm RACFALLO which defaults to Alter (alloc)(rename to).

Relevant RACF system parameters: - RACFALLO, RACFMODL, RACFNEWN, RACFPRED, SECURDRS, and VSACCESS

Protection of User Datasets Under RACF — LISTD

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be reported on	volume level check (of SECURVOL “Y”)	Alter-scratch
	dataset level check (done if vol check fails or not done)	Alter-scratch

Protection of User Datasets Under RACF — IXMAINT

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be scratched from the files dataset	volume level check (of SECURVOL “Y”)	Alter-scratch
	dataset level check (done if vol check fails or not done)	Alter-scratch

Protection of User Datasets Under RACF — MERGE

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be scratched from the files dataset	volume level check (of SECURVOL “Y”)	no check
	dataset level check (done if vol check fails or not done)	Update-output

Protection of User Datasets Under RACF — REBUILD

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be overlaid in the CA-Disk archives	volume level check (of SECURVOL “Y”)	no check
	dataset level check (done if vol check fails or not done)	Update-output

Protection of User Datasets Under RACF — MOVE/COPY

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be copied	volume level check (of SECURVOL “Y”)	no check
	dataset level check (done if vol check fails or not done)	Update-output or Control-control
Dataset to be moved	volume level check (of SECURVOL “Y”)	Alter-scratch
	dataset level check (done if vol check fails or not done)	Alter-scratch
Target dataset preallocated	volume level check (of SECURVOL “Y”)	no check
	dataset level check (done if vol check fails or not done)	Update-output
Target dataset not preallocated (checks preexisting, non-CA-Disk, saved profiles only)	volume level check (of SECURVOL “Y”)	no check
	dataset level check (done if vol check fails or not done)	based on value of sysparm RACFALLO which defaults to Alter (alloc)

Relevant RACF system parameters: - RACFALLO, RACFNEWN, RACFVCAV, and VSACCESS

Protection of User Datasets Under RACF — RELEASE

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be reported on	volume level check (of SECURVOL “Y”)	no check
	dataset level check (done if vol check fails or not done)	Update-output

Protection of User Datasets Under RACF — Compress

		Authority level		
Resource to be checked	Volume or Dataset level check	Use parm specified	Use parm not specified	Reload command specified
Dataset to be compressed	Volume level check (if SECURVOL “Y”)	no check	Alter-scratch	no check
	Dataset level check (done if vol check fails or not done)	Read-input; Alter-scratch	Read-input; Update-output; Alter-scratch	no check
Cluster to be reloaded	Volume level check (if SECURVOL “Y”)	no check	no check	no check
	Dataset level check (done if vol check fails or not done)	no check	no check	Update-output

Relevant RACF system parameters: - RACFPDSW

Protection of User Datasets Under RACF — Sequential Migrate

		Authority level	
Resource to be checked	Volume or Dataset level check	Scratch parm specified	Scratch parm not specified
Dataset to be migrated	Volume level check (if SECURVOL “Y”)	no check	Alter-scratch
	Dataset level check (done if vol check fails or not done)	Read-input	Read-input; Alter-scratch

Relevant RACF system parameters: - RACFSEQM

Relevant RACF CA-Disk **user exits: -** RACFSMEX

Protection of User Datasets Under RACF — Delete

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be scratched	volume level check (of SECURVOL “Y”)	Alter-scratch
	dataset level check (done if vol check fails or not done)	Alter-scratch

Protection of User Datasets Under RACF — LISTREQ

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be reported on	volume level check (of SECURVOL “Y”)	no check
	dataset level check (done if vol check fails or not done)	Read-report

Protection of User Datasets Under RACF — DERASE

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to have request deleted	volume level check (of SECURVOL “Y”)	Alter-scratch
	dataset level check (done if vol check fails or not done)	Alter-scratch

Protection of User Datasets Under RACF — DSCB Update

Resource to be checked	Volume or Dataset level check	Authority level
Dataset to be processed	volume level check (of SECURVOL “Y”)	Alter-scratch
	dataset level check (done if vol check fails or not done)	Update-output; Alter-scratch

Protection of User Datasets Under RACF — VBACKUP

Resource to be checked	Authority level
Volume to be backed up	Read-input

Protection of User Datasets Under RACF — VRECOVER

Resource to be checked	Authority level
Volume to be recovered	Read-input
Target volume	Alter-scratch; Update-output
IDVOL for CLIP	Alter-scratch; Update-output

Functions that Need no Dataset/Volume Authorization

The following functions do not actually access any user datasets, so no security authorization checking is done at the dataset/volume level. However they do use commands and do access the files dataset, so security authorization checking is done only at the files dataset subfile level and the CA-Disk command level.

- Format files dataset
- Unload files dataset
- Reload files dataset
- Reorg files dataset

RACF Discrete Profile Handling

RACF Terms Defined

DISCRETE PROFILE - A discrete profile is a security mask covering one resource such as a dataset on a particular volume. A discrete profile is deleted by the operating system when the resource covered by that profile is deleted. The profile resides as a record in the RACF dataset.

GENERIC PROFILE - A generic profile is a security mask covering zero or more resources, such as all datasets on all volumes that have a dataset name matching a particular pattern. A generic profile is not deleted by the operating system when a resource covered by that profile is deleted. The profile resides as a record in the RACF dataset.

RACF-INDICATED - A dataset is RACF-indicated if the RACF indicator bit for the dataset is turned on. For non-VSAM datasets, the RACF indicator bit is the DS1IND40 bit (bit x'40' at offset 93 x'5D') located in the format-1 DSCB. For VSAM datasets, the RACF indicator bit is the x'80' bit in the SECF field located in the cluster's catalog entry.

ALWAYS-CALL - A feature of the operating system where a security package is queried for authorization regardless of whether the resource is RACF-indicated or not.

RACF-PROTECTED - Always call is on: A RACF-indicated dataset covered by a discrete or generic profile is protected. A RACF-indicated dataset not covered by a discrete or generic profile is in error. A non-RACF-indicated dataset covered by a generic profile is protected. A non-RACF-indicated dataset not covered by a generic profile is not protected. Always call is off: A RACF-indicated dataset covered by a discrete or generic profile is protected. A RACF-indicated dataset not covered by a discrete or generic profile is in error. A non-RACF-indicated dataset is not protected.

CA-Disk-SAVED PROFILE - When CA-Disk backs up or archives a dataset, CA-Disk will optionally copy the discrete profile, if any, protecting that dataset. This copied or CA-Disk-saved profile contains all of the original profile information, except that the dataset name and volume have been changed (CA-Disk correlates this saved profile back to its DSNINDEX entry via the RACFENCD subfile, described in “Reconciling CA-Disk and RACF Records” on page 346).

The copied profile resides in the RACF dataset, just as any other RACF profile.

The CA-Disk-saved profile is then available to put back the original profile information should the dataset be restored or recovered at a later time.

Generic profiles are not scratched by RACF when resources covered by them are scratched. Therefore, CA-Disk does not copy generic profiles, only discrete profiles.

RACF and VSAM Datasets

Like non-VSAM datasets, VSAM clusters can also be RACF- protected. CA-Disk maintains RACF profiles for VSAM clusters in the same way support is provided for non-VSAM datasets.

Consistent with IBM's policy, CA-Disk maintains discrete profiles for the cluster names of VSAM clusters in ICF catalogs. Consistent with IBM's policy, CA-Disk maintains discrete profiles for each of the cluster, data component, index component, and any AIX or PATH names of VSAM clusters in non-ICF catalogs. CA-Disk maintains this policy even when a cluster is transferred back and forth between the two different types of catalogs.

Backing Up RACF Discrete Profiles (RACFBKUP)

In normal processing, any CA-Disk function that archives or backs up a dataset protected by a discrete profile will also create a CA-Disk-saved profile for the dataset. For some users this can be costly in terms of both job performance and DASD space utilization in the RACF profile dataset.

If you do not want CA-Disk to create CA-Disk-saved profiles for back ups of datasets protected by discrete profiles, specify sysparm RACFBKUP with a value of Y. CA-Disk will create CA-Disk-saved profiles for archives of datasets protected by discrete profiles.

If you do not want CA-Disk to create CA-Disk-saved profiles for back ups nor archives of datasets protected by discrete profiles, specify the RACFBKUP system parameter with a value of F.

If you specify a value of Y or F, it is your responsibility to create a correct profile for datasets being restored prior to using the dataset. Review the RACFMODL discussion below as one method of creating discrete profiles to cover datasets being restored for which CA-Disk has no associated CA-Disk-saved profiles.

Also please review the sysparm RACFPRED regarding its effect on restoring datasets where a discrete profile already exists.

Processing RACF-Indicated Datasets Without Profiles

CA-Disk will process RACF-indicated datasets for which no profile exists when the CA-Disk RACF Security Interface is inactive (sysparm RACFSUPPn is specified). CA-Disk will issue an error message and bypass RACF-indicated datasets for which no profile exist when the CA-Disk RACF Security Interface is active (sysparm RACFSUPPy is specified).

Restoring RACF-Indicated Datasets Having no Profiles

To protect dataset access integrity, CA-Disk normally will not allow RACF-protected datasets to be restored unless either their associated CA-Disk-saved profiles are available or the datasets were protected by a generic profiles when backed up.

If you specify sysparm RACFSUPPn, RACFBKUPy or RACFBKUPf at archive/backup time, suffer loss of some CA-Disk-saved profiles, or go to a disaster recovery site without your RACF dataset, you need some way to restore your datasets that were originally protected by discrete profiles.

This option allows you to specify a default discrete profile to be used for all datasets to be restored that do not have profiles.

The default profile is specified through sysparms that contain a dataset name and volume serial number. The values must correspond to an existing discrete RACF profile.

When a dataset without a CA-Disk-saved profile is to be restored, the attributes and access list of the model profile are assigned to the restored dataset. Normally, the attributes are those of the RACF administrators or some other individual responsible for maintenance of RACF profiles. They can also be used as codes to be passed to RACF user exits that assign attributes based on installation requirements.

To activate the use of model profiles, build a discrete profile to be used as a model, specify sysparm RACFMODL with a value of Y, sysparm RACFMDSN with the 1- to 44-character name of the model profile, and sysparm RACFMVOL with the 1- to 6-character volume association of the model profile.

Reconciling CA-Disk and RACF Records

The RACF encode type 2 technique requires CA-Disk to keep in synchronization three different records relating to saved RACF profiles. First, the DSNINDEX subfile record in the files dataset must contain the bit to indicate that a profile is being maintained for the archived dataset. Then RACF profile records must be kept in the users RACF profile dataset. Third, a RACFENCD subfile record must be in the files dataset relating the DSNINDEX record to the RACF profile.

If CA-Disk is improperly set up and run, the relationship between some records can be destroyed. For example, running the IXMAINT utility to delete expired datasets without the sysparms RACFPROC and RACFSUPP specified with a value of Y will leave disconnected RACF profiles on the RACF profile dataset and will leave unused RACFENCD records in the files dataset. These disconnected records will not cause any harm except to take up space in the datasets.

To verify that this situation does not exist, or to correct it if it does, the following utility reconciles the relationship between RACF profile information. The utility verifies the names of CA-Disk RACF profiles read from the users

RACF dataset with the RACFENCD record and the RACF bit in the DSNINDEX record. The following conditions are processed:

- If a valid profile exists, but the corresponding RACFENCD record does not exist, delete the RACF profile.
- If a RACFENCD record exist for a DSNINDEX record, but a RACF profile does not exist, delete the RACFENCD record and turn off the DSNINDEX record bit.
- If both a valid profile and a RACFENCD record exist for a DSNINDEX record, but the DSNINDEX record bit is off, turn on the DSNINDEX record bit.
- If both a valid profile and a RACFENCD record exist for a DSNINDEX record, but the DSNINDEX record does not exist, delete both the RACF profile and the RACFENCD record.
- If a DSNINDEX record bit is on, but the corresponding RACFENCD record does not exist, turn off the DSNINDEX record bit. Use this utility whenever the integrity of the relationship between profile information is in question. Because of the need to process all CA-Disk-saved profiles, the user ID of the job statement of this utility must have the SPECIAL attribute.

This utility can take a significant amount of time to run, particularly in installations with a large RACF dataset.

The program runs as a batch application. Use caution when executing this utility to ensure no valid records are deleted by mistake. The first step of the utility contains an unload of the files dataset to ensure a good backup copy will always be available. This utility should be run in simulate mode and all return codes should be checked thoroughly to ensure the integrity of the DSNINDEX, RACFENCD subfiles and IBM's RACF profile dataset.

The utility uses the following RACF TSO command to create the list of CA-Disk-saved profiles for reconciling with the RACFENCD and DSNINDEX records:

```
SEARCH MASK(racfusid.,.THM) NOGENERIC
```

The lower case letters represent the values of the CA-Disk sysparm. Information on the command itself can be obtained by entering the TSO command HELP SEARCH. Thus the possibility exists that some profiles that are not CA-Disk-saved profiles can be listed and processed:

- If you have any discrete RACF profiles that are not CA-Disk-saved profiles that have a dataset name that starts with the value of sysparm RACFUSID, followed by a period, followed somewhere later by .THM, then those profiles will be listed by this utility, found to have no corresponding entry in the RACFENCD subfile, and if the XPRO parameter is specified, deleted. It is very unlikely that any non-CA-Disk-saved profiles would meet these criteria and would be deleted by this utility, but you should consider this possibility.

Note for Users With Multiple Files Datasets: If some or all of your files datasets have the same values for the sysparms RACFUSID and RACFDVOL, you must not use the XPRO parameter of this utility. CA-Disk uses the values of these sysparms to determine which profiles are CA-Disk-saved profiles. Running the XPRO parameter on this utility on a files dataset will delete profiles saved for other files datasets if and only if the other files datasets have the same value of the sysparms RACFUSID and RACFDVOL. If you have only one files dataset or all of your files datasets have Parmlibs with different values for RACFUSID or RACFDVOL, you are free to use any or all parameters of this utility.

Following is sample JCL for executing the utility.

```
//S1       EXEC RACFCHK1
//RECONCI.SYSIN  DD *
   CHECK SIM,OKL,XPRO,XENC,XDSN
```

Check Command Parameter Description

```
CHECK SIMULATE,XPROFILES,XENCDRECS,XDSNINDEX,OKLIST
```

SIMULATE - Specification of this parameter causes a report to be produced indicating the data selected and the actions that would have occurred in live mode processing. It is useful in verifying that the input parameters will actually produce the intended results. It is highly recommended that this parameter be used before running in live mode.

XPROFILES - If a valid profile exists, but the corresponding RACFENCD record does not exist, delete the RACF profile.

- If you do not specify this parameter, this condition will not be listed or corrected. Before using this parameter, read carefully the warning above.

XENCDRECS - If a RACFENCD record exist for a DSNINDEX record, but a RACF profile does not exist, delete the RACFENCD record and turn off the DSNINDEX record bit.

- If you do not specify this parameter, this condition will not be listed or corrected.

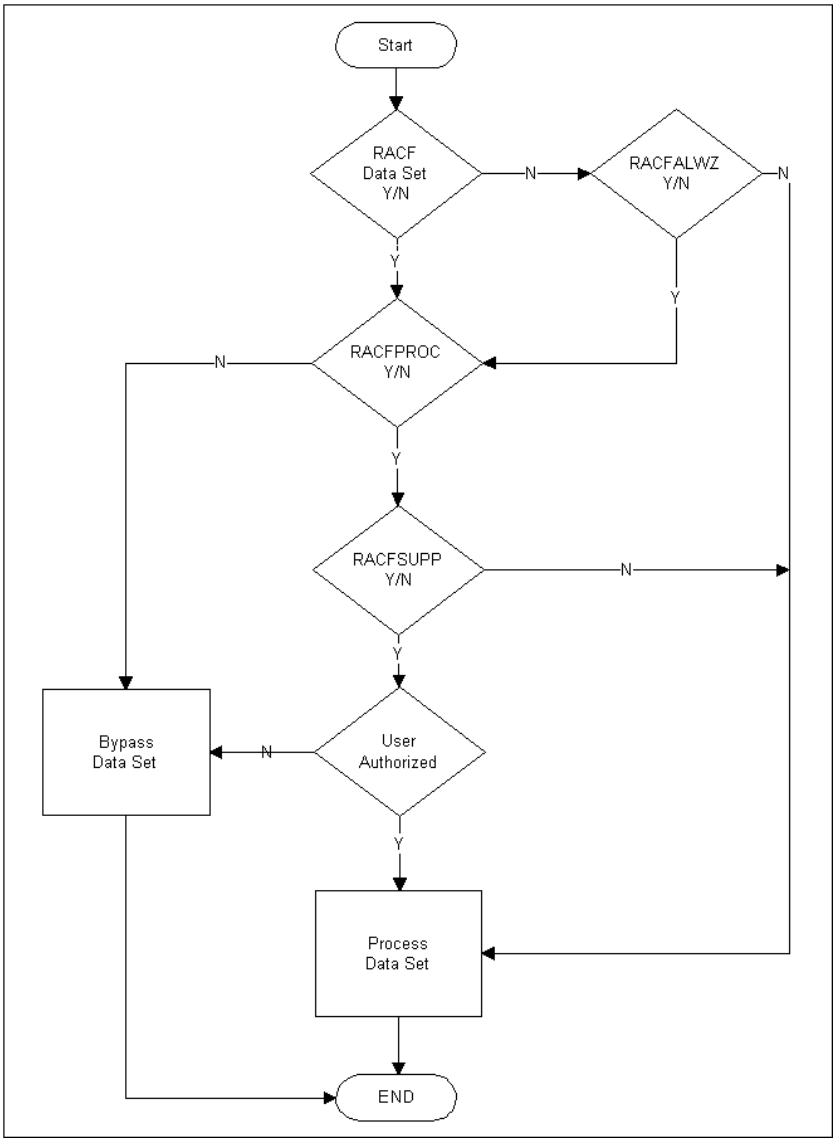
XDSNINDEX - If both a valid profile and a RACFENCD record exist for a DSNINDEX record, but the DSNINDEX record bit is off, turn on the DSNINDEX record bit.

- If both a valid profile and a RACFENCD record exist for a DSNINDEX record, but the DSNINDEX record does not exist, delete both the RACF profile and the RACFENCD record.
- If a DSNINDEX record bit is on, but the corresponding RACFENCD record does not exist, turn off the DSNINDEX record bit.
- If you do not specify this parameter, these conditions will not be listed or corrected.

OKLIST - Specify this parameter to print messages for each valid relationship processed. If you do not specify this parameter, messages will be printed only for error conditions.

RACF Checking Logic Summary in CA-Disk

FIGURE 25. RACF Checking Logic Summary Flowchart



Use of RACF Macros

CA-Disk interfaces to RACF entirely through standard RACF macro interfaces. The macros used and their purposes are:

RACROUTE REQUEST=AUTH (previously RACHECK)

To determine authorization to process datasets and the existence of generic profiles.

ICHEINTY

To determine the existence of discrete profiles.

RACROUTE REQUEST=DEFINE,TYPE=DEFINE

To copy profiles in Backup, Archive, Restore, Recover and Move/Copy. The MENTITY and MVOLSER fields are used to create the new profile with all profile information copied from the existing profile. All fields are copied across. However, if the userid of the task has the Operations attribute, RACF replaces the OWNER field of the new profile with the userid of the task.

RACROUTE REQUEST=DEFINE,TYPE=ADDVOL

To add a volume to discrete profiles in PDS Compress and Move/Copy.

RACROUTE REQUEST=DELETE,TYPE=ADDVOL

To delete discrete profiles.

RACSTAT

To determine if RACF is active on the system.

RACROUTE REQUEST=VERIFY

To give a started task the same authority as the originating task.

RACROUTE

To invoke the System Authorization Facility (SAF) MVS router.

RACF User Exits

If CA-Disk RACF checking is not adequate for users, exit routines have been provided at various points to give user control to validate, reject, modify or monitor CA-Disk processing regarding RACF. Following is a list of these CA-Disk RACF User Exits:

TABLE 164. List of RACF User Exits

Exit Name	Page #	Brief Description
ICHEKAEX	222	Profile Type Pre-Determination Exit
ICHEKPEX	223	Profile Type Post-Determination Exit
SECURAEX	242	Security Validation (post decision) Exit

TABLE 164. List of RACF User Exits

Exit Name	Page #	Brief Description
SECURPEX	243	Security Validation (pre decision) Exit
RAADDAEX	228	RACF RACDEF-Addvol Post Processing Exit
RAADDPEX	229	RACF RACDEF-Addvol (Pre-RACHECK) Exit
RACFDAEX	230	Post-Decode of CA-Disk RACF Profile
RACFDPEX	230	Pre-Decode of CA-Disk RACF Profile
RACFEAEX	231	Post-Encode of CA-Disk RACF Profile
RACFEPEX	231	Pre-Encode of CA-Disk RACF Profile
RACFSMEX	232	Process RACF-Protected DSN with Seq. Migrat
RACHKAEX	232	Post-RACF RACHECK Processing Exit
RACHKPEX	233	Pre-RACF RACHECK Processing Exit
RADEFAEX	233	RACF RACDEF Processing Post-RACDEF Exit
RADEFPEX	234	RACF RACDEF Processing Pre-RACDEF Exit
RADELAEX	235	Post-RACF RACDEF (Delete) Processing Exit
RADELPEX	235	Pre-RACF RACDEF (Delete) Processing Exit

RACF Installation Exits

Since CA-Disk uses standard RACF macros to perform RACF profile maintenance, any RACF user exits implemented will receive control for CA-Disk functions as well as standard dataset processing. This adds to the compatibility of CA-Disk with the user's RACF environment. If RACF user exits are implemented, verify that their being invoked by CA-Disk will not cause problems. The following two situations should be reviewed:

RACF can Need to Bypass CA-Disk Dataset Names

CA-Disk profile dataset (RESOURCE) names have a special encoded format. They have a standard profile format but user exit routines will not be able to recognize the names. Either they should be bypassed by RACF exit routines or special code can be needed to process them. Modules have been provided to decode and re-encode dataset names. They are described below under the heading RACF Internals.

Special RACF Processing for GDG Datasets

Some users have set up RACF to maintain only a single RACF profile for each GDG base dataset name. This improves RACF performance by reducing the number of updates and accesses to the RACF dataset. It also simplifies maintenance of GDG profiles by greatly reducing the number of profiles maintained. This technique is implemented through RACF user exits as described in the IBM manual RACF - Installation Reference Manual.

CA-Disk honors this type of dataset protection. While CA-Disk does detect the existence of model GDG profiles, CA-Disk does not make copies of them in archive, restore or any other function. If your shop uses model GDG profiles, examine closely the sysparm RACGDGVL.

DSNINDEX RACF Flags

The CA-Disk DSNINDEX records for the CA-Disk archives contain flag bits relating to RACF protection. These flags are reported on the LISTD report. For more detail, turn to the topic "LISTD Flags" on page 314 in the User's Guide.

Format of CA-Disk-Saved Profiles

CA-Disk optionally copies discrete profiles of datasets that are backed up or archived. This profile is used to re-create profile information when the dataset is restored. Since the original dataset name cannot be used for the RACF profile name (it can still be in use by the version of the dataset still on DASD), and to allow for multiple versions of a profile for a dataset, CA-Disk creates a special profile name for the dataset.

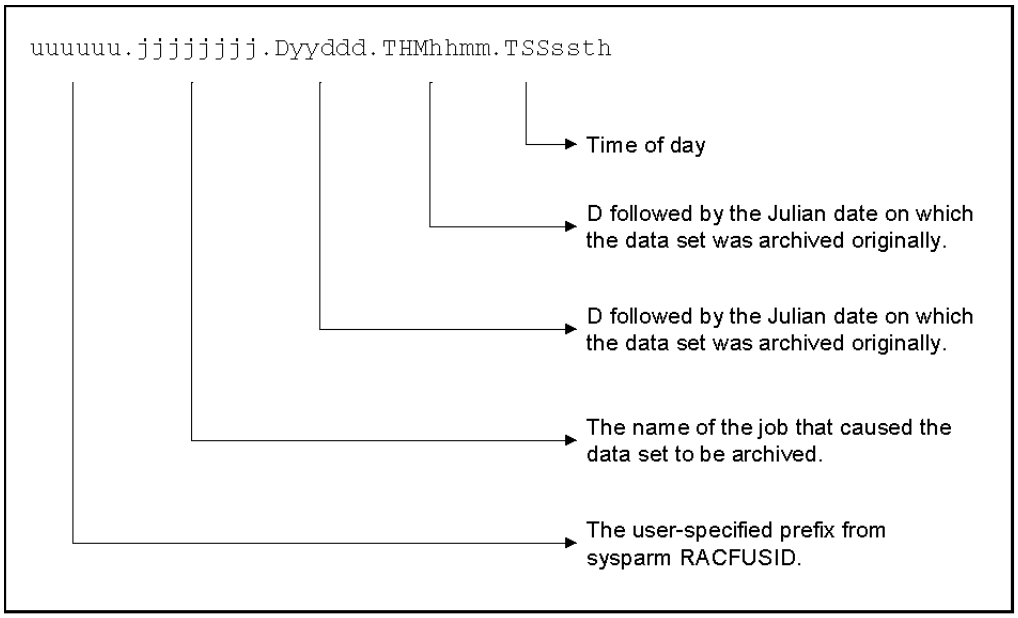
The special profile name is in encoded format referred to as ENCODE FORMAT 2 and is readable and compatible with all RACF utilities.

A separate CA-Disk subfile was created to support the ENCODE FORMAT 2 technique (the RACFENCD subfile of the files dataset). This file cross-references the archive dataset name, date and time to the CA-Disk profile name. This file of information is used to find the CA-Disk profile name when the dataset is to be restored with its discrete profile.

RACF Encode Profile (Format 2)

The CA-Disk profile name is generated by taking the user-specified prefix, JOBNAME, archive date and archive time, and combining them to make one dataset name. The following is the format of the dataset name.

FIGURE 26. Racf Encode Profile Generated Dataset Name



Two CA-Disk modules perform the encoding and decoding of CA-Disk profile dataset names. They are distributed in the CA-Disk source library and are available for use by user programs if desired. Their program and source member names are:

```
ADSDS019 - ENCODE SAMS:Disk-PROFILE dataset name
ADSDS025 - DECODE SAMS:Disk-PROFILE dataset name
```

Comments at the beginning of the module describe how to use the modules.

Conversion of CA-Disk RACF Profile Encoding

Note: This section is provided for those users who have older versions of the encoding technique. If you do not have an older version (referred to as ENCODE FORMAT 1), you can disregard this section.

CA-Disk can only operate with one profile encoding technique in effect at any one time. If a user wishes to change encoding techniques and currently has CA-Disk RACF profiles existing for archived datasets, either all RACF-protected archive datasets must be purged from the archives or a conversion utility must be run to convert existing profiles to the desired format.

Execute the following JCL to convert CA-Disk RACF profiles from Format 1 to Format 2:

```
//UNLOAD   EXEC PGM=ADSMI002,PARM=ADSDM177
//STEPLIB   DD  DSN=SAMS.DISK.LOADLIB,DISP=SHR
//ABNLDUMP  DD  DUMMY
//CMDPRINT  DD  SYSOUT=A
//MSGPRINT  DD  SYSOUT=A
//SYSPRINT  DD  SYSOUT=A
//SYSUDUMP  DD  SYSOUT=A
//FILES     DD  DSN=SAMS.DISK.FILES,DISP=SHR
//PARMLIB   DD  DSN=SAMS.DISK.PARMLIB,DISP=SHR
//SEQFILES  DD  DISP=(NEW,PASS),DSN=&&DSINDX,
//         DCB=(LRECL=259,BLKSIZE=6144,DSORG=PS,RECFM=VB),
//         SPACE=(6144,(360,360),RLSE),UNIT=SYSDA
//SYSIN     DD  *
   UNLOAD FILES=DSNINDEX
//CONVERT  EXEC PGM=ADSMI002,PARM='ADSDS058'
//STEPLIB   DD  DSN=SAMS.DISK.LOADLIB,DISP=SHR
//ABNLDUMP  DD  DUMMY
//MSGPRINT  DD  SYSOUT=A
//SYSUDUMP  DD  SYSOUT=A
//PARMLIB   DD  DSN=SAMS.DISK.PARMLIB,DISP=SHR
//FILES     DD  DSN=SAMS.DISK.FILES,DISP=SHR
//SYSUT1    DD  DSN=&&DSINDX,DISP=(OLD,DELETE)
```

Management of CA-Disk-Saved Profiles

Nearly all management of CA-Disk-saved profiles is done automatically by the relevant functions involved — Archive/Backup, Restore and IXMAINT. In the event of special cases, two independent utility programs are supplied. They provide the capability to list and maintain RACF profiles on both CA-Disk and standard dataset profiles.

The two utilities and their functions are:

1. CA-Disk-Saved Profile List Utility 2

   This utility applies to only those RACF users who have implemented CA-Disk RACF ENCODE FORMAT 2 (sysparm RACFENTY2). It provides a cross-reference listing for CA-Disk archived dataset names to RACF profile dataset names. It provides two reports, the first in sequence by RACF profile dataset name, the second in sequence by archive dataset name, date and time.

2. CA-Disk-Saved Profile Maintenance Utility

   This utility allows the user to LIST, DELETE, CREATE and REPLACE RACF profiles. It is designed to be used as a general profile maintenance utility. It provides the capability to process CA-Disk ENCODED PROFILES.

CA-Disk-Saved Profile List Utility 2

This utility creates two cross-reference reports of CA-Disk RACF profile dataset names to CA-Disk archive dataset names. One report is in sequence by archive dataset name, the other report is in sequence by CA-Disk RACF profile dataset name.

The program runs as a batch program. Following is example JCL for executing the function:

```
//LISTDSN  EXEC DMSLRAC
//SYSIN     DD  *
   (DMS SELECT command goes here)
/*
```

DD Statement Explanations

SYSIN - This input stream is to contain the CA-Disk SELECT command. Its purpose is to further restrict program output to the information specified by the user.

Select Command

The SELECT command restricts list utility output, based on DSNAME or DSNAME PREFIX. If no parameters are specified with the SELECT command, all profiles will be listed. Command format and parameter description follow:

```
SELECT DSNAME=,PROFILE=
```

DSNAME= - Limit archive dataset names listed to the DSNAME or DSNAME PREFIX specified by this parameter. A maximum of 20 DSNAMES or DSNAME PREFIXES can be supplied.

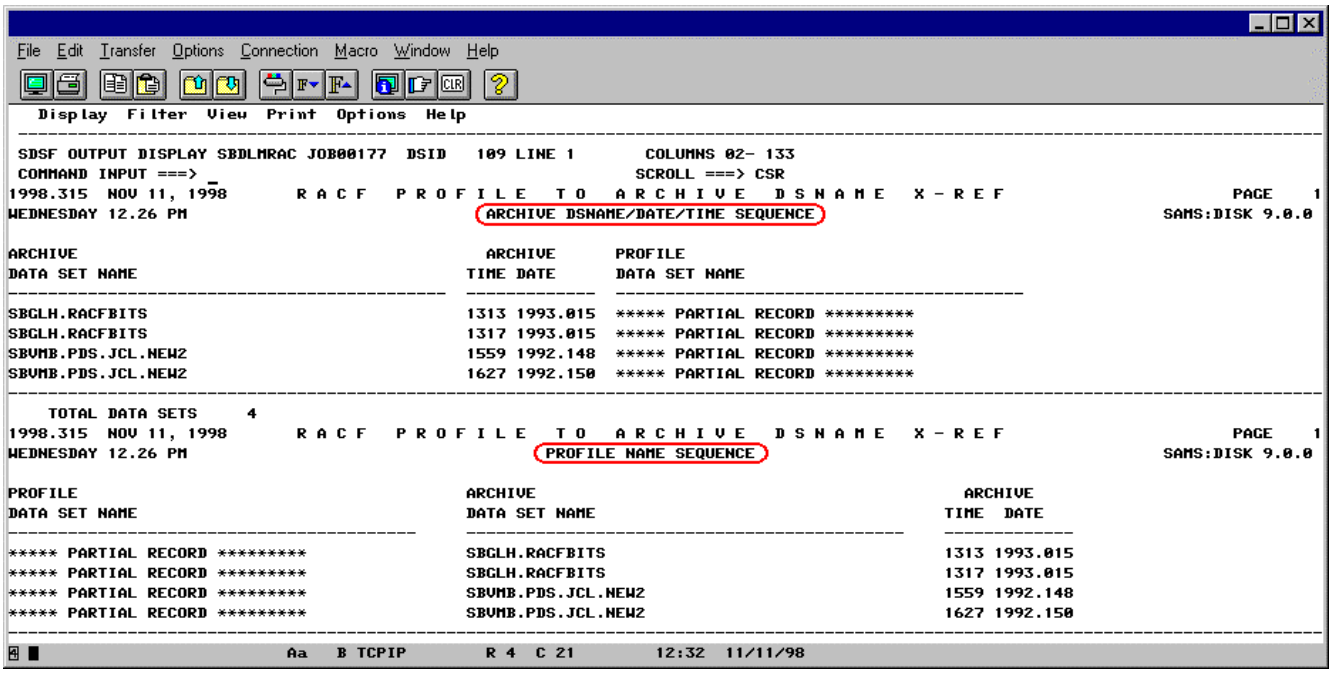
PROFILE= - Limit profile names listed to the NAME or NAME PREFIX specified by this parameter. A maximum of 20 names or prefixes can be supplied.

Example

The following is an example execution of the RACF PROFILE LIST UTILITY 2. Running this utility generates both reports.

```
//LIST     EXEC DMSLRAC
//SYSIN     DD  *
  SELECT DSN=(SYS1./,USER./)
/*
```

FIGURE 27. Sample DMSLRAC Report



```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

 Display  Filter  View  Print  Options  Help
SDSF OUTPUT DISPLAY SBDLMRAC JOB00177  DSID   109 LINE 1       COLUMNS 02- 133
COMMAND INPUT ===> _                                         SCROLL ===> CSR
1998.315  NOV 11, 1998      R A C F   P R O F I L E   T O   A R C H I V E   D S N A M E   X - R E F                PAGE    1
WEDNESDAY 12.26 PM                          ARCHIVE DSNAME/DATE/TIME SEQUENCE                                SAMS:DISK 9.0.0

ARCHIVE                                      ARCHIVE      PROFILE
DATA SET NAME                               TIME DATE     DATA SET NAME
-------------------------------------------  -------------  ----------------------------------------
SBGLH.RACFBITS                              1313 1993.015  ***** PARTIAL RECORD *********
SBGLH.RACFBITS                              1317 1993.015  ***** PARTIAL RECORD *********
SBVMB.PDS.JCL.NEW2                          1559 1992.148  ***** PARTIAL RECORD *********
SBVMB.PDS.JCL.NEW2                          1627 1992.150  ***** PARTIAL RECORD *********
   TOTAL DATA SETS     4
1998.315  NOV 11, 1998      R A C F   P R O F I L E   T O   A R C H I V E   D S N A M E   X - R E F                PAGE    1
WEDNESDAY 12.26 PM                          PROFILE NAME SEQUENCE                                            SAMS:DISK 9.0.0

PROFILE                                     ARCHIVE                                        ARCHIVE
DATA SET NAME                               DATA SET NAME                                TIME  DATE
-------------------------------------------  -------------------------------------------  -------------
***** PARTIAL RECORD *********              SBGLH.RACFBITS                               1313 1993.015
***** PARTIAL RECORD *********              SBGLH.RACFBITS                               1317 1993.015
***** PARTIAL RECORD *********              SBVMB.PDS.JCL.NEW2                           1559 1992.148
***** PARTIAL RECORD *********              SBVMB.PDS.JCL.NEW2                           1627 1992.150
                        Aa   B TCPIP        R 4  C 21        12:32  11/11/98
```

CA-Disk-Saved Profile Maintenance Utility

This utility provides a tool to maintain CA-Disk profile information. It gives the capability to create, delete, list and rename RACF profile data. A user must be authorized to update the profile of a dataset before using this program to maintain it.

The utility should be used to maintain discrete profiles only. Do not attempt to do generic profile maintenance with it.

The program is executed in a batch environment. Transaction listings are created to indicate the status of the requests entered.

Based on input command parameters, this utility can be used to maintain both CA-Disk profiles and standard dataset profiles (discrete profiles only). Multiple input commands can be supplied to process different profiles.

Four different commands are accepted by the utility:

DISPLAY - This command requests that the profile data for a specific RACF profile be displayed. The following fields are shown:

- dataset name
- volume(s)
- dataset type
- universal access
- auditing
- owner ID
- installation data
- access list — group name and access level

DELETE - Delete the specified RACF profile. Take no action on the dataset associated with the profile.

CREATE - This command allows a new profile to be generated from an existing one. The user specifies the name and volume of the existing profile to be used as a model to create the new one.

RENAME - Assign a new name and volume to a RACF profile. If the profile is associated with an existing dataset, the original dataset will no longer have a profile.

Execution JCL Example

```
// EXEC DMSPROF
//SYSIN DD *
(maintenance commands go here)
/*
```

Display Command

```
DISPLAY DSNAME=,VERSION=,NONDMS,VOLUMES=,VSAM
```

DSNAME= - Specifies the 1- to 44-character dataset name of the profile to be listed. This must be a complete name. It can contain a generation dataset version number if desired.

- If this request is to display a CA-Disk profile, a date and time value must be used to determine the encoded dataset name for the profile. Both a date and time command parameter can accompany the dataset name to be used to find the profile. If these are not supplied, the DSNINDEX record will be used to determine the archive date and time for the dataset. The date and time parameters are not normally supplied with the command unless the DSNINDEX record is not available or is unusable.

VERSION= - If the DSNINDEX file is to be used to determine the date and time for a CA-Disk profile, this parameter can indicate the version of a dataset in the index if multiple copies have been archived (that is, VERSION=(-01)).

VOLUMES= - If the request is for a CA-Disk profile, this parameter indicates the volume with which CA-Disk has the profile associated. In this case, the VOLUME parameter is not normally supplied because CA-Disk always associates profiles with the volume specified on sysparm RACFDVOL. The value from the sysparm is used as a default for this parameter when CA-Disk profiles are being processed.

- If this request is for a standard dataset profile, CA-Disk needs the volumes on which the dataset is protected. If this parameter is not specified, the system catalog will be used to determine the volume serials. If the dataset is not cataloged, specify one or more volumes on which this dataset is protected.

NONDMS - This simple parameter indicates the dataset name supplied is for a standard dataset profile.

- If this parameter is not specified, it is assumed the request is for a CA-Disk profile.

> *Note:* DATE and TIME parameters are not valid when this parameter is specified.

VSAM - Specify this parameter to indicate the request is for a profile associated with a VSAM dataset. When not specified, the request is assumed to be for a non-VSAM dataset.

Delete Command

```
DELETE DSNAME=,VERSION=,NONDMS,VOLUMES=,VSAM,LIST
```

DSNAME= - (See "DSNAME=" on page 356)

VERSION= - (See "VERSION=" on page 356)

VOLUMES= - (See "VOLUMES=" on page 357)

NONDMS - (See "NONDMS" on page 357)

VSAM - (See "VSAM" on page 357)

LIST - This parameter causes the profile being deleted to be displayed before deletion. The format in which it is displayed is the same format as the one generated by the DISPLAY command.

Create Command

```
CREATE OLDDSN=,OLDVER=,OLDVOLS=,NEWDSN=,NEWVER=,NEWVOLS=,OLDNONDMS,
       NEWNONDMS,LIST,VSAM
```

OLDDSN= - (See "DSNAME=" on page 356)

OLDVER= - Pertains to the model (original) profile being used. (See "VERSION=" on page 356)

OLDVOLS= - This parameter indicates the volume(s) for which the model profile is defined. (See "VOLUMES=" on page 357)

NEWDSN= - Specifies the 1- to 44-character dataset name of the new profile to be created. (See "DSNAME=" on page 356)

NEWVER= - Pertains to the new profile. (See "VERSION=" on page 356)

NEWVOLS= - This parameter indicates the volume(s) to be assigned to the new profile. (See "VOLUMES=" on page 357)

OLDNONDMS - This simple parameter indicates the dataset name supplied for the model (original) profile is not for a CA-Disk profile.

NEWNONDMS - Same meaning and format as the OLDNONDMS parameter, except that it applies to the profile to be created.

VSAM - Specify this parameter to indicate the request is for a profile associated with a VSAM dataset. When not specified, the request is assumed to be for a non-VSAM dataset.

LIST - This parameter causes the profile created to be displayed. The format in which it is displayed is the same format as the one generated by the DISPLAY command.

Rename Command

```
RENAME OLDDSN=,OLDVER=,OLDVOLS=,NEWDSN=,NEWVER=,OLDNONDMS,NEWNONDMS,LIST,VSAM
```

OLDDSN= - (See "DSNAME=" on page 356)

OLDVER= - Pertains to the model (original) profile being used. (See "VERSION=" on page 356.)

OLDVOLS= - This parameter indicates the volume(s) on which the profile being renamed is defined. (See "VOLUMES=" on page 357.)

NEWDSN= - Specifies the 1- to 44-character dataset name to be assigned to the profile. (See "DSNAME=" on page 356)

NEWVER= - Pertains to the new profile. (See "VERSION=" on page 356)

OLDNONDMS - This simple parameter indicates the profile being renamed is not a CA-Disk profile.

NEWNONDMS - Same meaning and format as the OLDNONDMS parameter except that it applies to the new profile name.

VSAM - Specify this parameter to indicate the request is for a profile associated with a VSAM dataset. When not specified, the request is assumed to be for non-VSAM.

LIST - This parameter causes the profile to be displayed. The format in which it is displayed is the same format as the one generated by the DISPLAY command.

Execution Examples:

The following example demonstrates displaying the RACF profile information for a CA-Disk RACF profile:

```
// EXEC DMSPROF
//SYSIN DD *
DISPLAY DSNAME=USER.LOAD.LIBRARY
/*
```

The following example shows the deletion of the same CA-Disk RACF profile:

```
// EXEC DMSPROF
//SYSIN DD *
DELETE DSNAME=USER.LOAD.LIBRARY
/*
```

The following example shows the deletion of a non-CA-Disk RACF profile:

```
// EXEC DMSPROF
//SYSIN DD *
DELETE DSNAME=USER.PROD.LIBRARY,NONDMS
/*
```

The following example shows the creation of a profile for DSNAME USER.DSN1 from an already existing profile USER.TEST2. Both datasets are on DASD and cataloged.

```
// EXEC DMSPROF
//SYSIN DD *
CREATE NEWDSN=USER.DSN1,NEWNONDMS,OLDDSN=USER.TEST2,OLDNONDMS
/*
```

Changing the Value of RACFDVOL

The following procedure describes how to change the value of RACFDVOL. This sysparm contains the real online DASD volume name that is required by RACF for CA-Disk-saved profiles. Actually, the steps describe what to do before you change the sysparm. Changing the sysparm is quite easy, but without doing the preparatory steps described here, system abends will occur.The jobs are long running, so submit them in batch mode.

1. Run the following TSO command in batch. Use the current values for sysparms RACFUSID and RACFDVOL, and the desired value of the sysparm RACFDVOL:

   ```
   SEARCH MASK(racfusid.,THM) VOLUME(olddvol)
   NOLIST NOGENERIC + CLIST('ALTDSD ',' ALTVOL(olddvol newdvol)')
   ```

   The TSO command creates a CLIST in the dataset 'userid.EXEC.RACF.CLIST'. This dataset contains TSO commands. There is one TSO command for each CA-Disk-saved RACF profile that will change its associated volume from the old DVOL to the new DVOL.
2. Verify that the CLIST contains a command for every CA-Disk-saved profile, and no others.
3. Do step 1 until you are sure that the list contains every CA-Disk-saved profile.
4. Plan a time and procedure for steps 5 through 10.
5. Make sure that no CA-Disk Backup, Archive or Restore jobs are running.
6. Backup the CA-Disk Files dataset and the RACF data base.
7. Do step 1 again.
8. Use batch mode to execute the CLIST you just created in 'youruserid.EXEC.RACF.CLIST'.
9. Verify that each command in the CLIST completed successfully.
10. Change the setting of CA-Disk sysparm RACFDVOL to the new value. The change is now complete, and you can allow CA-Disk processing again.

CHAPTER 11

Report Processing

CA-Disk report processing is accomplished by generating report detail records, sorting them, and passing them to a common report writer routine. Each report detail record is prefixed by its 10-byte report table name and its report sequencing control field. The sequencing control field can optionally be used to provide page control breaks for the report. Only one control field can be used to generate page control breaks.

Overview

CA-Disk places report detail records for all reports requested into one common output data set. After all detail report records are generated, a sort step is executed to order the records by report table name/control field sequence. These records are then passed to the common report writer routine.

The report writer uses the report table name to retrieve the proper table to use to format the report. The format control table contains all information needed to build the printed report line from the supplied detail records. The contents and field definitions for the formatting control tables is documented later in this section. When a detail record is read for a different report, the new format control table is loaded and the processing continues. A flow diagram describing this process is also presented in a later section.

Several reports deal with VSAM KSDS data sets and identify them as “KSDS”. Note however that the convention used for VSAM KSDS Extended data sets is “KSDE”.

REPORTNAME Table — Member REPORTNM

The report name table (member REPORTNM in the parmlib data set) relates modules that construct output records to the tables used by the report writer to format the respective reports. The content of the entries in the report name table is identified below.

TABLE 165. REPORTNM Content Descriptions

Cols	Description		
1-8	Name of module that produces report output records		
9	Character indicating which type of DSCB during VTOC SCAN is to be passed to the module:	1	Format-1 DSCB only.
		5	Format-5 DSCB only. - All DSCB types when blank.
		V	Do not pass any DSCB - Invoke at end of volume for extraction of summary data.
10-19	10-character name of the report table to be used by the report writer in formatting the report		

The last six characters of each report table name serve as the name of the respective report for the REPORT and DISPLAY commands.

REPORTNAME Table Example:

```
TE = 'ADSDM0521RPTNMVDICT'
```

TABLE 166. REPORTNM Example

Value	Definition
ADSDM052	Name of module to be given control to produce the Multiple Volume Dictionary Report records
1	1 Format-1 DSCBs are the only ones passed to the report module
RPTNMVDICT	Table used to format the Multiple Volume Dictionary Report

Format-1 / Format-4 DSCB Appendage Fields

Allocation information regarding individual data sets that is not available in the format-1 DSCB is maintained as an appendage to the DSCB, which is passed to the modules that assemble the data for the report record. Allocation information that has been accumulated from the processing of all DSCB types is available as an appendage to the format-4 DSCB, which is also passed as one of the parameters to the report assemble modules. The fields are described in the appendix of this manual.

Tables for Report Formatting

The majority of the CA-Disk reports are formatted by the use of tables that are stored in the parmlib data set. Any change made to a table is reflected the next time the subject report is printed, since tables are loaded at execution time.

Four types of table entries, or records, are used in report preparation. These are the CONTROL, HEADER, DETAIL, and TOTAL types that are described below.

1. Control Entry

   Only one occurrence of this type can be present for each report format control table. It contains control information about the layout of a specific report.

TABLE 167. Control Entry Description

Cols	Description
1	'C' for entry type
2-4	(not used)
5-7	Limit number of lines per page for overflow. (Overridden by RPTLINES sysparm).
8-10	Rows of detail lines per page[a]
11-13	Columns of detail per page[a]
14-16	Length of print detail to be replicated[a]
17-19	Length of source record for each detail set[a]
20-22	Length of control field that will force page overflow (control breaks)

a. Required for N-up reports

2. Header Entry

   An entry of this type is required for each field and literal to be printed on a report heading line.

TABLE 168. Header Entry Description

Cols	Description
1	'H' for entry type
2	The character placed in this position is the identifier for all entries comprising the given heading line
3	The carriage control character for the heading line in question must be specified in this position for only the first entry of each line

TABLE 168. Header Entry Description

Cols	Description
4	Field type code
7	Length of the source field in bytes
10	Position of field in source record — left blank for literals
11-13	Position on print line for field placement
14-16	Target length of field where different from source length (halfword to zoned)
17-nn	Field type dependent

3. Detail Entry

   An entry of this type is required for each field and literal to be placed on a report detail line.

TABLE 169. Detail Entry Description

Cols	Description
1	’D’ for entry type
2	’1’ is normally placed in the identifier field since only one type of detail line is presently supported
3	The carriage control character to be used for all detail lines is specified here
4	Field type code
7	Length of the source field in bytes
10	Position of field in source record — left blank for literals
11-13	Position on print line for field placement
14-16	Target length of field where different from source length (halfword to zoned)
17-nn	Field type dependent

Fields and literals are specified in the same manner for both of the applicable line types.

TABLE 170. Possible Line Type Values and Descriptions

Cols	Description
4	Field type code
7	Length of the source field in bytes
10	Position of field in source record — left blank for literals
11-13	Position on print line for field placement
14-16	Target length of field where different from source length (halfword to zoned)
17-nn	Field type dependent

4. Total Entry

   An entry of this type is required for each field and literal to be placed on a report total line.

TABLE 171. Total Entry Description

Cols	Description
1	’T’ for entry type
2	Numeric indicator of total line occurrence to which this entry applies
3	The carriage control character to be used for all total lines is specified here

TABLE 171. Total Entry Description

Cols	Description
4	Field type code
5-7	Length of the source field that is to be accumulated. The source field must be either a halfword or fullword in the record. If this value is zero, a length of 2 is assumed. Valid values are 000, 002 or 004.
8-10	Position of field in source record — left blank for literals. If this value is zero, the accumulator is assumed to be a count of records processed.
11-nn	All remaining fields have the same meanings as for detail type records. (See the description of the detail entry for their formats and use.)

Described below are the various field types and special considerations for their use (position 4 in table entry).

TABLE 172. Special Field Types for Position 4

Value	Description	Value	Description
A	Accumulator for Report Totals	S	CA-Disk System Header
B	Binary	T	Time
C	Character	U	Allocation Type
D	Date	V	Data Set Organization
E	Security	W	Record Format
F	Fullword	X	Hexadecimal
H	Halfword	Y	Create/Expire Date
I	Installation Name	Z	Zoned Decimal
J	Create/Expire Date Expanded Form	1	Mount Status
K	Constant / Literal	2	Use Status
M	Mask	3	Standard Header Line 1
N	Page Number	4	Standard Header Line 2
P	Packed Decimal	$	Dollars and Cents
R	Device Type		

Format Table Examples

The following table entry examples are taken from the RPTNOMVDIC table (member RPTNOMVD in the parmlib data set).

Control Entry Example

```
TE = 'C1 058054001132126020'
```

TABLE 173. RPTNOMVD Example

Cols	Sample	Description
1		C entry type
4		(not used)
5-7	058	Lines per page
8-10	054	Detail lines per page

TABLE 173. RPTNOMVD Example

Cols	Sample	Description
11-13	001	Columns of detail
14-16	132	Length of detail line
17-19	126	Length of source record
20-22	020	Length of control field that will force page overflow; in this case, when the 20 characters of the owner name change, the detail record is printed on a new page.

Header Entry Examples

```
TE = 'H5 K013 006 DATA SET NAME'
```

TABLE 174. Header Entry Example

Cols	Sample	Description
1	H	Entry type - header
2	5 LIne	5 of the header
3		(not used)
4	K	Literal - “DATA SET NAME”
5-7	013	Length of literal
8-10		Blanks - not used with a literal
11-13	006	Position in print line where literal begins
14-16		Blanks - not used with a literal
17-28	DS...ME	The literal value that is placed in the print line

Detail Entry Examples

```
TE = 'D1 C006071033 VOL'
```

TABLE 175. Detail Entry Example #1

Cols	Sample	Description
1	D	Entry type - detail
2	1	Line 1 of detail
3		Blank - single space
4	3	Literal in source
5-7	006	Length of literal in source
8-10	071	Position of literal in source relative to table name
11-13	033	Position in print line were the literal is to begin
14-19		Blanks - not used with a literal
20-22	VOL	Comments

```
TE = 'D1 H002082069005Z BLKSIZE'
```

TABLE 176. Detail Entry Example #2

Cols	Sample	Description
1	D	Entry type - detail
2	1 Line	1 of detail
3		Blank - single space
4	H	Data in source is a halfword
5-7	002	Length of data in source
8-10	082	Position of data in source relative to the table name
11-13	069	Position in print line where data is to begin
14-16	005	Maximum length of data in print line is five characters
17	Z	Suppress leading zeros
18-19		Blanks - not used
20-26	BLKSIZE	Comment

Total Entry Examples

```
TE = 'T10K015 005 TOTAL DATA SETS'
```

TABLE 177. Total Entry Example #1

Cols	Sample	Description
1	T	Entry type - total
2	1	Line 1 of total
3	0	Double space
4	K	Constant total header
5-7	015	Length of constant
11-13	005	Position in print line where the literal is to begin
17-28	xxx	The constant value that is placed in the print line

```
TE = 'T1 A000000021005Z BLKSIZE'
```

TABLE 178. Total Entry Example #2

Cols	Sample	Description
1	T	Entry type - total
2	1	Line 1 of total
3		Blank - single space
4	A	This is an accumulator field
5-7	000	Due to the type of accumulator (count), this value is ignored
8-10	000	Accumulator is to count number of records passed to reporting routine
11-13	069	Position in print line where total is to begin
14-16	005	Maximum length of data in print line is five characters
17	Z	Suppress leading zeros
18-19		Blanks - not used
20-26	BLKSIZE	Comment

Note: Each source record begins with two half-words followed by the name of the table to be used to print the record.

```
REC          TABLE          Position in SOURCE
LEN          NAME           1                  21        27
01260000     RPTNOMVDIC     OWNER NAME...      PROJNO    DSN...
```

Report Modification Example

If a change is made to the Multiple Volume Dictionary Report, such as putting an owner in the heading, the following changes also need to be made.

1. Add an entry to the SYSPARMS member in parmlib.

RPTMVDEXexitname - where exitname is the name of the module to be given control to write an owner record.

2. Add a new formatting table as a member in parmlib that is to be used for printing the report.
3. Add the following entry to the report name table (member REPORTNM in parmlib).

   ```
   ADSDM0521RPTNOMVDIC
   ```

4. Write an exit module that calls ADSDM033 to write the new record based on the following parameters passed to the exit module:

TABLE 179. Parameters Passed to module ADSDM033

Parameters	Description		
P1	format-4 DSCB and appendage fields volume serial is in positions 1 to 6		
P2	format-1 DSCB and appendage fields		
P3	standard report record for Multiple Volume Dictionary Report		
P4	Halfword return code:	1	WRITE standard Multiple Volume Dictionary record
		2	DO NOT WRITE standard MVD record

The report can be invoked using the following command:

```
REPORT RPT=NOMVDIC,VOL=VOL001
```

Example of Owner Table

In this table, provision has been made for an OWNER in the header.

FIGURE 28. MEMBER=RPTNOMVD in the PARMLIB data set

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTNOMVD) - 01.00          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132090'
 000002 'H113132   001 standard header - line 1'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129'
 000005 'H1 K032   033   M U L T I P L E   V O L U M E   '
 000006 'H1 K038   065   D A T A   S E T   D I C T I O N A R Y'
 000007 'H2 4132   001 standard header - line 2'
 000008 'H3 K001   001132'
 000009 'H4 K005   001   OWNER'
 000010 'H4 C020001008 OWNER'
 000011 'H5 K011   038 PSWD DEVICE'
 000012 'H5 K031   081   ---ALLOCATION-- ----TRACKS-----'
 000013 'H5 K013   001   DATA SET NAME'
 000014 'H5 K006   029   VOLUME'
 000015 'H5 K011   036   TYPE   ADDR'
 000016 'H5 K007   048   T O DSO'
 000017 'H5 K026   056   RECFM BLKSZ LRECL TYP PRIM'
 000018 'H5 K016   084   SEC   ALLOC IDLE'
 000019 'H5 K030   102   EXT CREDATE  EXPDATE  USEDATE '
 4 ▌                       Aa   A TCPIP        R 17 C 23        16:18  11/17/97
```

Compare the RPTNOMVD member table with the standard MVD table below.

Standard Multiple Volume Dictionary Report Table

FIGURE 29. **MEMBER=RPTNMVDI in the PARMLIB data set**

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTNMVDI) - 01.02          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132090'
 000002 'H113132   001'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129'
 000005 'H1 K032   033   M U L T I P L E   V O L U M E   '
 000006 'H1 K038   065   D A T A   S E T   D I C T I O N A R Y'
 000007 'H2 4132   001'
 000008 'H3 K003   048   C P'
 000009 'H4 K015   036   DEVICE UNIT A R'
 000010 'H4 K031   074   ---ALLOCATION-- ----TRACKS-----'
 000011 'H5 K013   001   DATA SET NAME'
 000012 'H5 K006   029   VOLUME'
 000013 'H5 K011   036   TYPE   ADDR'
 000014 'H5 K007   048   T O DSO'
 000015 'H5 K026   056   RECFM BLKSZ LRECL TYP PRIM'
 000016 'H5 K016   084   SEC   ALLOC IDLE'
 000017 'H5 K030   102   EXT CREDATE  EXPDATE  USEDATE '
 000018 'H6 K040   001   --------------------------- ------ -----'
 000019 'H6 K040   041   - ---- --- --- ----- ----- ----- --- ---'
 4 ▌                       Aa   A TCPIP        R 4  C 15        14:12  12/05/97
```

Example of Record Format for Owner Record

The following table describes the record format of the Owner Record:

TABLE 180. Record Format for the Owner Record

Name	Type	Size	Offset	Description
ownrecrd	DS	0F		
	DC	h’126,0’		Record length
owntable	DC	CL10’RPTNBMVDIC’		Report table name
ownid	DC	CL20’ ’	1	Owner ID
ownprojn	DC	CL6’ ’	21	Project number
owndsna	DC	CL44’ ’	27	Data set name
ownvolno	DC	CL6’ ’	71	Volume
owndevcl	DC	XL2’0000’	77	Device class
owndsorg	DC	XL2’0000’	79	DSORG
ownrecfm	DC	XL1’00’	81	RECFM
ownblksi	DC	XL2’0000’	82	BLKSIZE
ownlrecl	DC	XL2’0000’	84	LRECL
ownscalo	DC	XL1’00’	86	Allocation (trk,cyl,..)
ownscqty	DC	XL3’000000’	87	Second alloc quantity
owntksum	DC	XL3’000000’	90	Total allocation
owncredt	DC	XL3’000000’	93	Creation date
ownexpdt	DC	XL3’000000’	96	Expiration date
owndsind	DC	XL1’00’	99	Security
ownprmal	DC	XL2’0000’	100	Primary allocation
ownidlet	DC	XL2’0000’	102	Idle
ownvolad	DC	CL4’ ’	104	Volume address
owncatlg	DC	CL1’ ’	108	Cataloged
ownlusdt	DC	XL3’000000’	109	Last used date
ownnoext	DC	XL1’00’	112	Number of extents

Compare the OWNER RECORD to the standard MVD record that follows.

Standard Multiple Volume Dictionary Record Format

The following table describes the record format of the Standard Multiple Volume Dictionary:

TABLE 181. Record Format of the Standard MVD

Name	Type	Size	Offset	Description
rptrecrd	DS	0f,		
	DC	h’100,,0’		Record length
rpttable	DC	CL10’REPTNMVDICT’		Report table name
rptdsnam	DC	CL44’ ’	1	Data set name

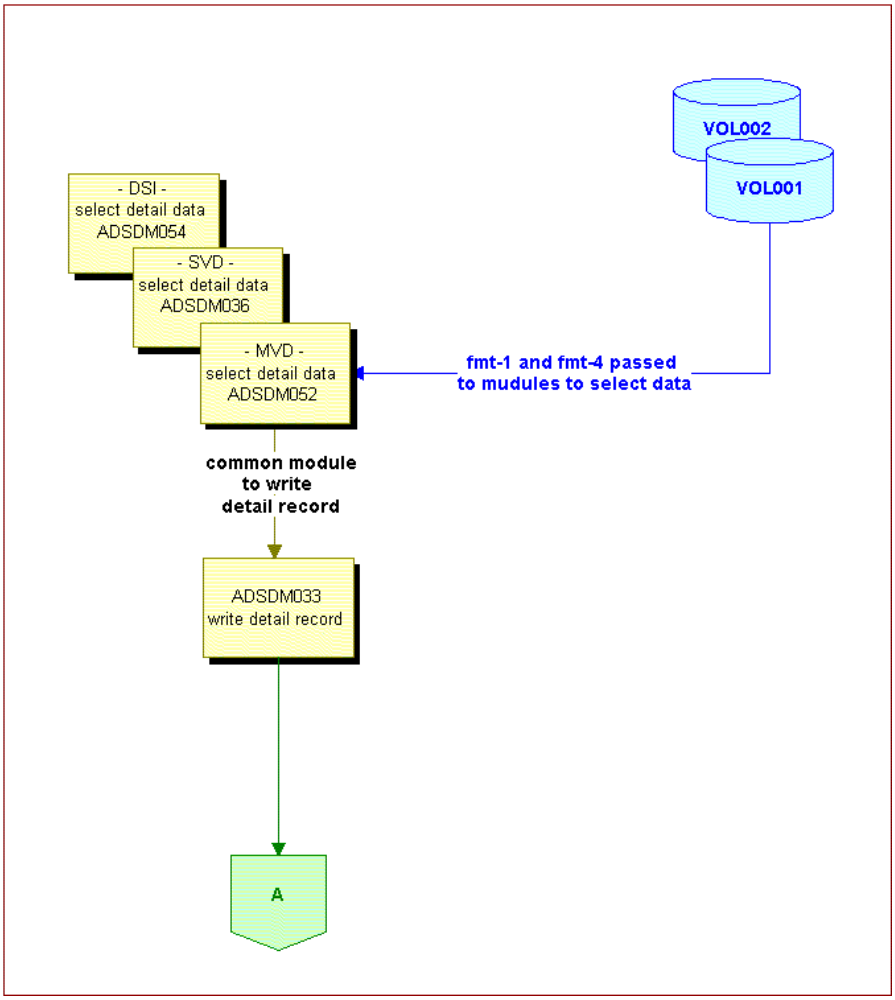
TABLE 181. Record Format of the Standard MVD

Name	Type	Size	Offset	Description
rptvolno	DC	CL6’ ’	45	Volume
rptdevcl	DC	XL2’0000’	51	Device class
rptdsorg	DC	XL2’0000’	53	DSORG
rptrecfm	DC	XL1’00’	55	RECFM
rptblksi	DC	XL2’0000’	56	BLKSIZE
rptlrecl	DC	XL2’0000’	58	LRECL
rptscalo	DC	XL1’00’	60	Allocat (trk,cyl,etc)
rptscqty	DC	XL3’000000’	61	Second alloc quantity
rpttksum	DC	XL3’000000’	64	Total allocation
rptcredt	DC	XL3’000000’	67	Creation date
rptexpdt	DC	XL3’000000’	70	Expiration date
rptdsind	DC	XL1’00’	73	Security
rptprmal	DC	XL2’0000’	74	Primary allocation
rptidlet	DC	XL2’0000’	76	Idle
rptvolad	DC	CL4’ ’	78	Volume address
rptcatlg	DC	CL1’ ’	82	Cataloged
rptlusdt	DC	XL3’000000’	83	Last used date
rptnoext	DC	XL1’00’	86	Number of extents

Report Logic Diagram

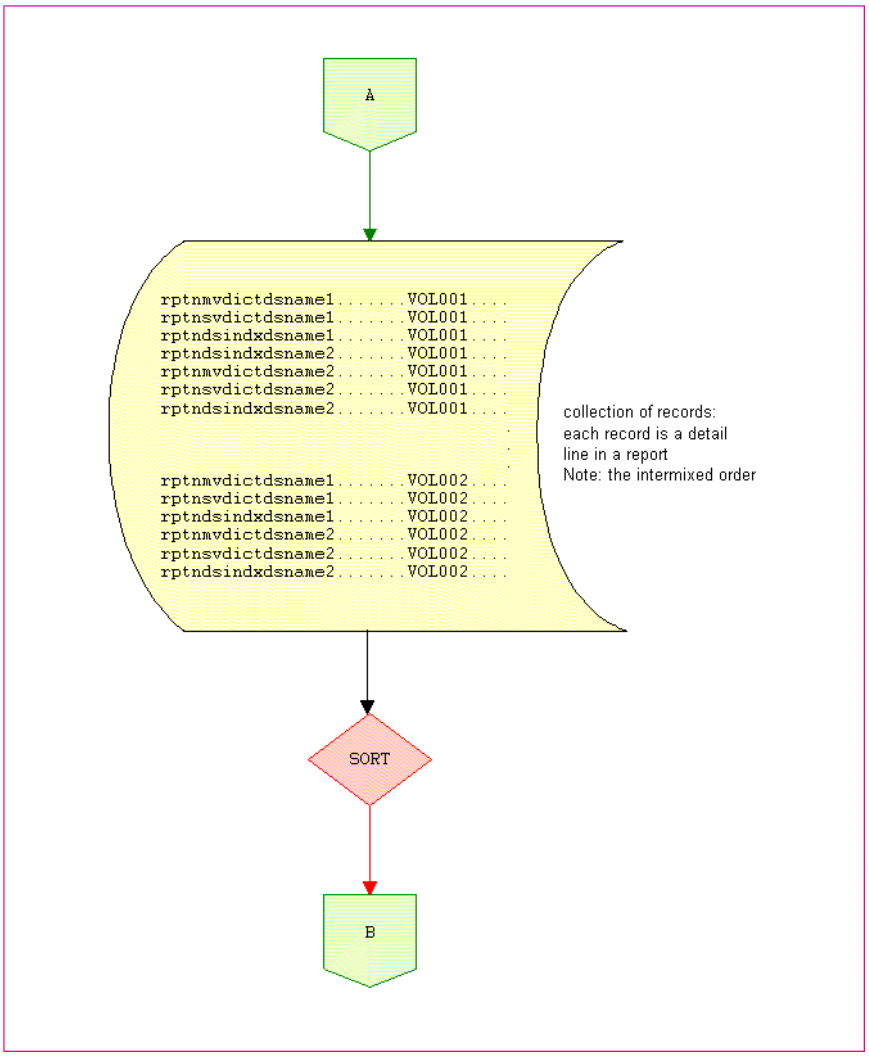
```
REPORT RPT=(MVD,SVD,DSI),VOL=(VOL001,VOL002)
```

FIGURE 30. Report Logic Diagram



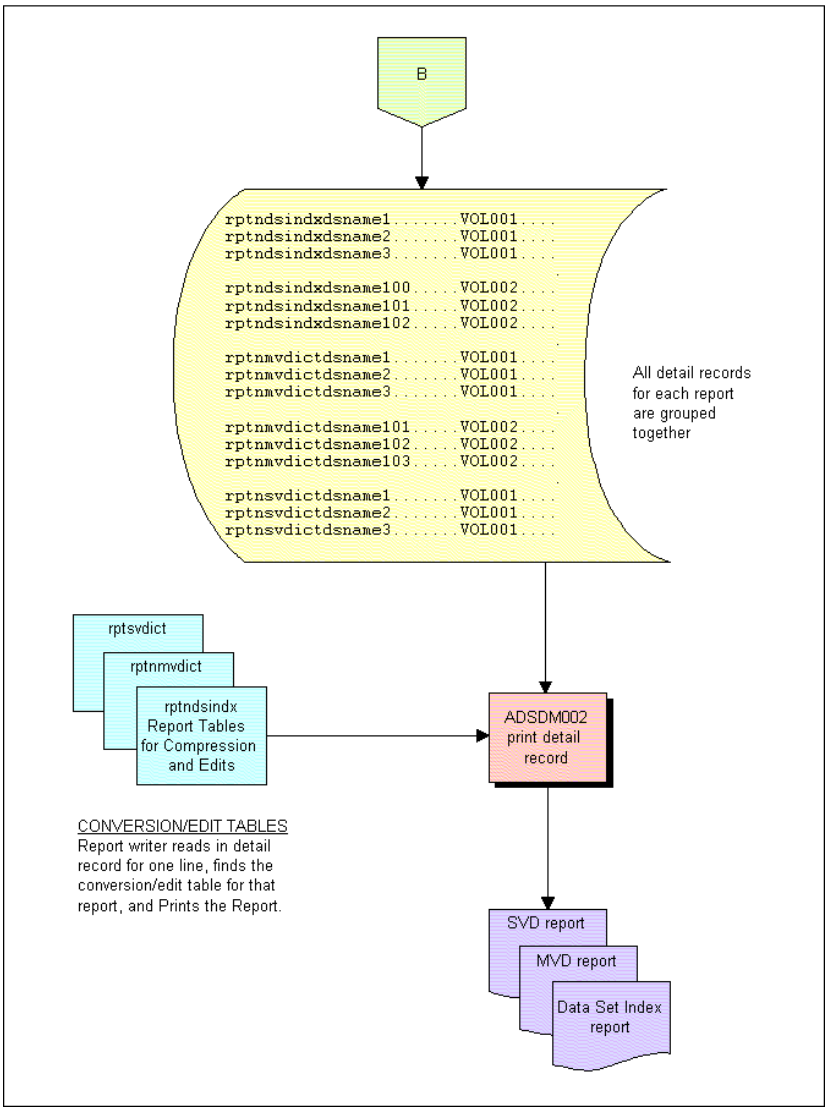
Each record is a detail line in a report as illustrated below:

FIGURE 31. Record Collection portion of Report Logic



All detail records for each report are then grouped together as illustrated below:

FIGURE 32. Detail Records Grouped Together Example



Report Processing - ISAM

DD Statements and Resulting Allocations

Any ISAM data set greater than one cylinder always has three separate areas:

```
                          INDEX AREA ..... I
                          PRIME AREA ..... P
      (Independent)       OVERFLOW AREA .. O
```

A one-cylinder data set never has an I or O area. A fourth area, the Cylinder Overflow (COV), can also exist and appears as the last (xx) number of tracks in each (p) cylinder. Existence of the COV is controlled by the OPTCD and CYLOFL subparameters of the DCB.

When the data set is being created (first allocated), the user can specify explicitly the location and size for each of the three areas. This is done by providing from one to three dd statements, with each dd statement giving the space requirements needed for the area being defined. The PRIME AREA (P) dd statement must always be supplied, with the INDEX (I) and OVERFLOW (O) dd statements needed only if explicit location and/or size is to be controlled. Whatever combinations of the I, P and O dd statements are supplied, they must appear in the order listed. INDEX DD —before — PRIME DD — before — OVERFLOW DD. (Reference: IBM Data Management Services Manual)

As stated above, the I, P and O areas always exist whether or not they were explicitly allocated via a dd statement. When only a PRIME AREA is allocated, ISAM suballocates cylinders from the end of the PRIME AREA to build the cylinder and master indexes. Tracks for the indexes are taken from the end of the cylinder. The remaining portion of the last cylinder used in this fashion is then made available as an OVERFLOW AREA.

Graphically, it appears as follows:

Primary Allocation			
CYL1	CYL2‘	CYL3	CYL4 suballoc I & O
P	P	P	P
			I

If both index and overflow areas are allocated via dd statements, the indexes and overflow records are placed in those areas, of course. However, if only one of the two is specified, then the other is automatically suballocated as described above that is, the indexes are built at the bottom of the last cylinder, and the remainder of the last cylinder is available for overflow.

Primary Allocation			Index	
CYL1	CYL2	CYL3	CYLx suballoc O	
P	P	P	O	OVERFLOW suballocated from the Index extent
COV	COV	COV	I	Index at bottom

EMBEDDED INDEX: There is one case slightly different from the above rules, which is called the embedded ISAM index. Similar to placing a VTOC in the middle of a DASD pack to reduce head-seeking, the cylinder and master indexes can be placed “in the middle” of the PRIME AREA. The user indicates this request via the SPACE parameter on the dd statement for the PRIME AREA. For example:

```
SPACE=(CYL,(10,,2))
```

causes the following allocation:

5 CYL PRIME — CYL INDEX — 5 CYL PRIME

If no separate overflow is defined via a dd statement, the unused portion of the last Index cylinder is still designated as OVERFLOW as described above.

ISAM Space Restrictions

Space for ISAM data sets must be allocated on cylinder boundaries and in integral multiples of whole cylinders. The request can be made in either CYL or ABSTR. When ABSTR is used, the cylinder boundaries and multiples requirements must still be met.

Because of the absolute addressing techniques used within ISAM, secondary extents are not allowed. This does not mean, however, that each ISAM area consists of only one extent. The reason for this is found in the DADSM allocation techniques. When a primary space allocation request is made, DADSM tries to satisfy this in a single extent. If a single extent cannot be found to meet the primary request, DADSM looks for two extents that satisfy the space need. DADSM continues “dividing” the primary into as many as five extents before giving up and issuing the “space not available” message.

To force DADSM to keep the primary space request in a single extent, the CONTIG parameter must be used in the space request. From this it can be seen that any ISAM data set can have up to five extents on any of its volumes. Since a single ISAM data set can be divided across three volumes (I, P and O), the maximum number of extents associated with the data set would be 15.

Note: This does not include the rare case of a prime area that spans multiple DASD volumes.

VTOC ENTRIES- Format 1, 2 and 3 DSCBs

Allocating a data set is synonymous with creating DSCBs in the VTOC. Every data set has a format-1 DSCB, which will describe up to three extents of the data set. If there are more than three extents (ISAM can have up to five), then the format-3 DSCB exists. For ISAM, a format-2 DSCB also exists. It is found on the volume specified by the first dd statement when the data set was created.

Another way of knowing if a DASD pack should have a format-2 DSCB is to consider the volume sequence (VOLSQ) number. When an ISAM data set is created, the volume specified on the first dd statement is called VOLSQ=1. If the second dd points to a different volume, then it is VOLSQ=2. If the third dd points to yet another volume, then it is VOLSQ=3. In the case where the ISAM areas are spread across two or three volumes, the format-2 DSCB will be found only on the volume with VOLSQ=1.

Note: The format-1 DSCB on the volumes with VOLSQ=2 or 3 contains the volume serial of the volume with VOLSQ=1; that is, they point back to the volume containing the format-2 DSCB.

Levels of Indexes:

Level	Description
0	Each PRIME cylinder has a track index
1	If there are two or more cylinders of PRIME data, then there is also a cylinder index
2	The user can specify if master indexes are also to be built by use of the (NTM) parameter. When the number of tracks in the cylinder index reaches the (NTM) number, the first level of master index is built.
3	If tracks in this level reach the (NTM), then a second level of master index is built
4	Similarly a third (highest) level can be built

In summary:

LEVELS=0	implies a 1-cylinder ISAM data set
LEVELS=1	implies cylinder indexes exist
LEVELS=2	implies CYL index plus 1st level master
LEVELS=3	implies CYL plus 1st and 2nd level mstr
LEVELS=4	implies CYL and three levels of master

ISAM Report Logic Overview

Space Calculations - INDEX

USED TRACKS: - The format-2 DSCB contains the start and end addresses for each of the cylinder and three levels of master indexes. The sum of the tracks in each range is the number of used index tracks. Special care must be taken if the starting and ending addresses are in two different extents of the data set.

AVAILABLE FOR OVERFLOW: - The fractional part of a cylinder not used by indexes is potentially available as an independent OVERFLOW AREA.

ALLOCATED TRACKS: - The number of tracks actually allocated for use as an INDEX AREA is calculated three different ways, depending on the manner in which it was allocated. The three methods are referred to as I1, I2, and I3 described below:

1. ALLOCATED TRACKS FOR INDEX = SUM OF INDEX EXTENTS
   This method is used when a separate index area was allocated and the overflow is not suballocated within this index area.
2. ALLOCATED TRACKS =(SUM OF EXTENTS) - (AVAILABLE OVFLOW)
   This method is used when the index area does exist but the overflow is suballocated (does not have its own area).
3. ALLOCATED TRACKS = USED TRACKS
   This method is used when the index does not have its own area, but is suballocated from either the prime or overflow areas.

Space Calculations — Prime - Suballocated Tracks (for INDEX and/or OVFLOW): The format-2 DSCB contains a pointer to the last cylinder available for prime data records. If this address is the same as the last cylinder in the prime extents, no suballocation from the prime area has been done. If they are different, the number of cylinders

suballocated can be computed, taking special care for the case when the actual end of the available prime area is in a different extent from the end of the allocated prime extents. Graphically,

Primary Allocation (extent 1)			Primary Allocation (extent 2)	
CYL1	CYL2	CYL3	CYL9 suballoc I & O	
P	P	P	O	OVERFLOW suballocated from the Prime extent
COV	COV	COV	I	Index at bottom

In the above diagram, CYL 9 is part of the PRIME AREA, extent two. The actual last cylinder available for prime data is CYL 3 in extent one.

Allocated Tracks:

$$Allocated\ Tracks\ (Prime)\ =\ (Sum\ of\ Prime\ Extents - Suballoc)$$

These calculations are done in cylinders and then multiplied by the number of prime tracks per cylinder, where;

$$Prime\ Tracks\ Per\ Cylinder\ =\ Device\ Trks/Cyl - (Cov)$$

Used Tracks: - The format-2 DSCB contains the address of the last data record written in the PRIME AREA. This address minus the beginning address of the PRIME AREA EXTENT 1 gives the used area. Again, the calculations must allow for crossing extents, and by doing the calculations in cylinders, the COV areas can be allowed for as described above under “Allocated Tracks:” on page 379.

Allocated 0 Tracks: - Space calculations for the OVERFLOW AREA can be divided into three cases referred to as O1, O2, and O3 described below:

1. The case when a SEPARATE O AREA exists with no suballocations for index.

$$ALLOCATED\ TRACKS\ =\ SUM\ OF\ THE\ O\ EXTENTS$$

2. The case when a SEPARATE O AREA does not exist, but is suballocated from the INDEX AREA, which can also be suballocated from the PRIME AREA. In this case, the actual available allocated tracks is just the unused portion of the last suballocated cylinder; that is, the portion not used by the indexes. This is the “available overflow” calculated in the Index Space Section.

$$ALLOCATED\ TRACKS\ =\ AVAILABLE\ OVERFLOW\ TRACKS$$

3. The case when a SEPARATE O AREA does exist, but has a number of tracks suballocated for Indexes.

$$ALLOCATED\ TRACKS\ =\ (SUM\ OF\ THE\ O\ EXTENTS) - (USED\ I\ TRKS)$$

Used O Tracks: - The format-2 DSCB contains the address of the LAST OVERFLOW RECORD. For cases O1 and O3, the used tracks are calculated by adding the tracks from the beginning address in EXTENT 1 of the O AREA up to the given address. In case O2, the used tracks is simply the “hh” of the “cchhr” address of the last O record, since that “cc” (of the cchhr) is the only “cc” available as OVERFLOW; that is, the beginning overflow address is cc001 and the last record address is cchhr.

The information to be reported for an ISAM data set can be divided into four major areas:

TABLE 182. Four Major Areas of ISAM data sets

Major Area	Description
Format-1 (and format-3) data	Always available
Format-2 data, direct copy	Available if volume contains a format-2
Detail Space usage data for any of the I, P and O areas	Computable only if the format-2 is on the volume and an analysis of several variables allows it
Other Volume information (Is part of this data set on other volumes?)	An analysis of all the variables will determine what can be reported

ISAM Variables Examined

The VARIABLES ANALYZED to determine how much can be reported are as follows:

1. Does this volume contain extents marked as I, P and/or O?
2. Does this volume contain the format-2 DSCB?
   - Does the LAST PRIME CYLINDER available equal the last cylinder in the PRIME EXTENTS? (suballocation?)
   - Does the CYL ADDRESS of the LAST OVERFLOW RECORD equal the cylinder address of the CYLINDER INDEX?

An analysis of all the different possible combinations of the answers to the above will determine the reportable information and hence the logic flow of the ISAM report modules.

Combinations of the Variables

The following chart indicates all of the combinations of VARIABLES on the left side. The right side contains the different information and case number to be reported.

TABLE 183. ISAM Variable Combinations

VARIABLES						REPORTED INFORMATION					
Extents Flagged								Detail Space			
I	(P)	O	FMT-2 DSCB?	CC P MAX= CC P Allo	CC 0 =CC CI	FMT 1	FMT 2	I	(P)	O	Other Volume
yes			yes	n/a	no	y	y	I1			other 1
yes			yes	n/a	yes	y	y	I2		O2	other 2
yes	yes		yes	n/a	no	y	y	I1	P1		other 3
yes	yes		yes	n/a	n/a	y	y	I2	P1	O2	other 8
yes		yes	yes	n/a	n/a	y	y	I1		O1	other 2
yes	yes	yes	yes	n/a	n/a	y	y	I1	P1	O1	other 8
	yes			n/a	n/a	y					other 4
	yes		yes	yes	n/a	y	y		P1		other 5
	yes		yes	no	n/a	y	y	I3	P2	O2	other 8
	yes	yes		n/a	n/a	y					other 6
	yes	yes	yes	n/a	n/a	y	y	I3	P1	O3	other 8
		yes		n/a	n/a	y					other 7

In the program logic, the DSCBs for each ISAM data set are examined for the conditions above. The six variables in the table are kept as the six high-order bits in a flag byte that is then used to control logic flow for computing the reportable information. Each YES position is a 1 bit, each blank or N/A is a 0 bit. For example, the first combination in the table creates a flag byte of x'90', the last combination is x'20'.

Each vertical line in the previous table corresponds to a bit setting in the flag byte. The following 12 values correspond to each line of the table, and are the actual bit setting for that line. (First is top line.)

1. X'90'
2. X'94'
3. X'D0'
4. X'D4'
5. X'B0'
6. X'F0'
7. X'40'
8. X'58'
9. X'50'
10. X'60'
11. X'70'
12. X'20'

ISAM Format-2 DSCB Offsets

The following tables summarize significant fields in the format-2 DSCB used for SPACE CALCULATIONS.

Which Offset	Cyl Indx	1st M	2nd M	3rd M
Starting address	72	79	1	13
Ending address	106	111	8	20

The START addr offset points to an (mbbcchh) field. If the first (b) byte of this address is not equal to zero, the end address given is in a different extent of the data set; that is, the index itself is split across different extents of the data set.

Offset	Overflow (independent)
of Start Addr	—
of Last Record Written Addr	116 (mbbcchhr)
to Free Tracks (RORG2)	126
to Number of O records	128
to Number of O reads/writes (RORG3)	61
	Cylinder Overflow
to Number of Full Areas	130
to Number of TRKS/Area	52
	Prime
to Last TRK of Last CYL avail.	36 (mbbcchhr)
to Last Record Written	93 (mbbcchhr)
to Number of records	67

CHAPTER 12

Algorithms

Space allocation for a non-VSAM data set is computed in the following manner.

$$\#\ of\ new\ tracks\ =\ \frac{total\ bytes\ of\ user\ data\ in\ input\ data\ set}{bytes\ of\ user\ data\ that\ fits\ on\ 1\ output\ track}$$

A SIZEIN subroutine computes the (TOTAL BYTES IN INPUT data set). A SIZEOUT subroutine computes the (BYTES FITTING ON 1 OUT TRK) and then performs the division.

Note: The largest result is obtained by maximizing the numerator and minimizing the denominator for this calculation.

After the computation is done, a percentage increase will be made based on sysparm values for ADPCTPDS and ADPCTOTH. The number of tracks as indicated by sysparm ADTRACKS is then added. Finally, if CYL or ROUND allocation is required, the total tracks is rounded up to an even multiple of cylinders.

Subroutine SIZEIN performs its calculations in the following manner:

$$User\ Data\ Bytes\ =\ (user\ data\ bytes\ per\ track)(\#\ of\ tracks)$$

1. (# of tracks) is a specific value passed as input.
2. (user data bytes per trk) depends on several items and is estimated as follows, with varying degrees of accuracy.
   - (blksize)(blks per track), — if 0 < BLKSIZE <= MAXBLK or
   - (maxblksz to fit)(1 per trk), — if not the above or
   - track size, — if DSORG=PS/PO,RECFM=V/U, and DEVTYPE changed.

   Note: PS data sets will always fit on the same number of tracks if the same device type is used. If a device-type change occurs, variable and undefined-length records can fit unpredictably, so an attempt is made to error on the safest side possible; that is, assume every single byte on the track was a user data byte.

All PO data sets are much like PS RECFM=V/U simply because of multiple members with each member's last block truncated, and the end-of-file marks embedded throughout the data set. Even if the members are copied to the same device type, more space can be required if the order in which the members are physically arranged on the tracks is changed. A device type change further complicates the ability to accurately predict the correct space requirements. For this reason, sysparms ADPCTPDS and ADTRACKS have been provided to automate a dynamic increase in the predicted space quantity.

Bytes that Fit on an Output Track

The number of bytes to fit on an output track is determined as follows:

1. (blksize)(blks per track), — if 0 < BLKSIZE <= MAXBLK or
2. (maxblksz to fit)(1 per trk), — if not the above

Special Case:

Number of tracks needed = number of tracks on input, if the DSORG=DA and relative track addressing is used. CA-Disk attempts to maintain the TTR position of the data records and hence requires the same number of TT tracks.

VSAM Idle Space Calculation

The idle space that is printed on the VSAM cluster detail report (CLD) and on the multiple and single volume dictionary reports (MVD and SVD), are estimates of free space based on catalog information for the cluster. Its calculation is based on the high used and high allocated relative byte addresses for each of the extents in the data set. It must be kept in mind that this value is only an estimate of idle space, and its accuracy will be dependent on the type of cluster being processed. This value will always express the MINIMUM amount of free space that a cluster has. Even if the idle space displayed is zero, the cluster can indeed have a large amount of embedded free space, especially in a KSDS that was defined with freespace percentages.

To calculate the amount of estimated free space a cluster has, CA-Disk retrieves all of the extent information for the data component of the cluster (and also for the index component if the cluster is a KSDS). The high used and high allocated relative byte addresses (RBA's) for each extent are compared to determine the percentage idle space in the extent. A running total is kept for all extents in the cluster. For the MVD and SVD reports, the percentage is applied at the volume level. The CLD report, however, applies the percentage to the data set as a whole, regardless of the number of volumes on which the data set resides. Because of this fundamental difference in reporting algorithms, the idle space total found at the end of the CLD report can vary slightly from that found in the MVD and SVD reports for the same clusters. This variation is insignificant and should be viewed in the light that these values are indeed only estimates. None of the reports can be considered the “most reliable” since the accuracy will vary between clusters.

The accuracy of the estimated idle space value depends on the type of cluster being viewed and the data that comprises the cluster. For an entry-sequence data set (ESDS), the value will be quite accurate. Because there cannot be any embedded free space in this type of cluster, the high used and high allocated RBAs will give an accurate depiction of the amount of space being used in the cluster. For a relative- record data set (RRDS), the accuracy will depend on the individual cluster. The idle space will reflect the amount of space available BEYOND the last record in the data set — it does not account for any empty record slots embedded in the cluster.

The calculation for key-sequenced data sets (KSDS) gives only the amount of idle space beyond the high used relative byte address in each extent. If a cluster is defined with embedded free space, or if the cluster has gone through numerous control interval and control area splits, the amount of actual idle space in the cluster can be significantly more than that calculated. Again, the idle space displayed will be the minimum amount that can be expected in the cluster.

Obviously this idle space calculation cannot be relied upon to give an absolute estimate of unused space in a cluster. Its true value is in pointing out clusters that can, in fact, be overallocated. If the idle space displayed is high, and the cluster is a KSDS or RRDS, then the chances are good that the cluster has too much space allocated to it. These particular clusters should then be examined to determine if they should indeed be reduced in size.

VSAM New Name Algorithm

When a cluster is restored with a new name, CA-Disk must determine a new name to use for the data component of the cluster. In addition, if the cluster is a key-sequenced data set, a new name must be generated for the index component. If the data (or index) component name was originally generated by VSAM (that is the NAME parameter was not specified for the data and index components when the cluster was defined), CA-Disk will let VSAM generate new name(s) for the cluster's components.

If the data and index components were defined with explicit names when the cluster was created, CA-Disk must generate new names for the components to avoid a “duplicate entry” error when defining the new cluster. The base cluster will be named whatever is specified on the NEWNAME= parameter. The data and index component names will be generated based on a comparison of the original cluster name and the new name that was specified. The base portion of the component name will be identical to the new name specified on the input command. The remainder of the name will be generated by comparing the original base cluster name with the original component name. CA-Disk will compare the two names until a difference is detected. The portion of the component name that is different is then appended to the base cluster's new name to determine the new name for the component. Note that if the comparison between the original base cluster name and the original component name is at other than a “.”, a “.” will be inserted into the component new name before appending the remaining portion of the component name. This is done to ensure that the node being generated will not exceed eight characters in length.

As an example, let's look at how the following new name would be generated:

```
Original cluster name = LABS.TJP.PMRFILE
Original data component name = LABS.TJP.PMRFILE.DATA
```

If the new name specified is LABS.NEW.PMRFILE, the new name for the data component would be created as follows:

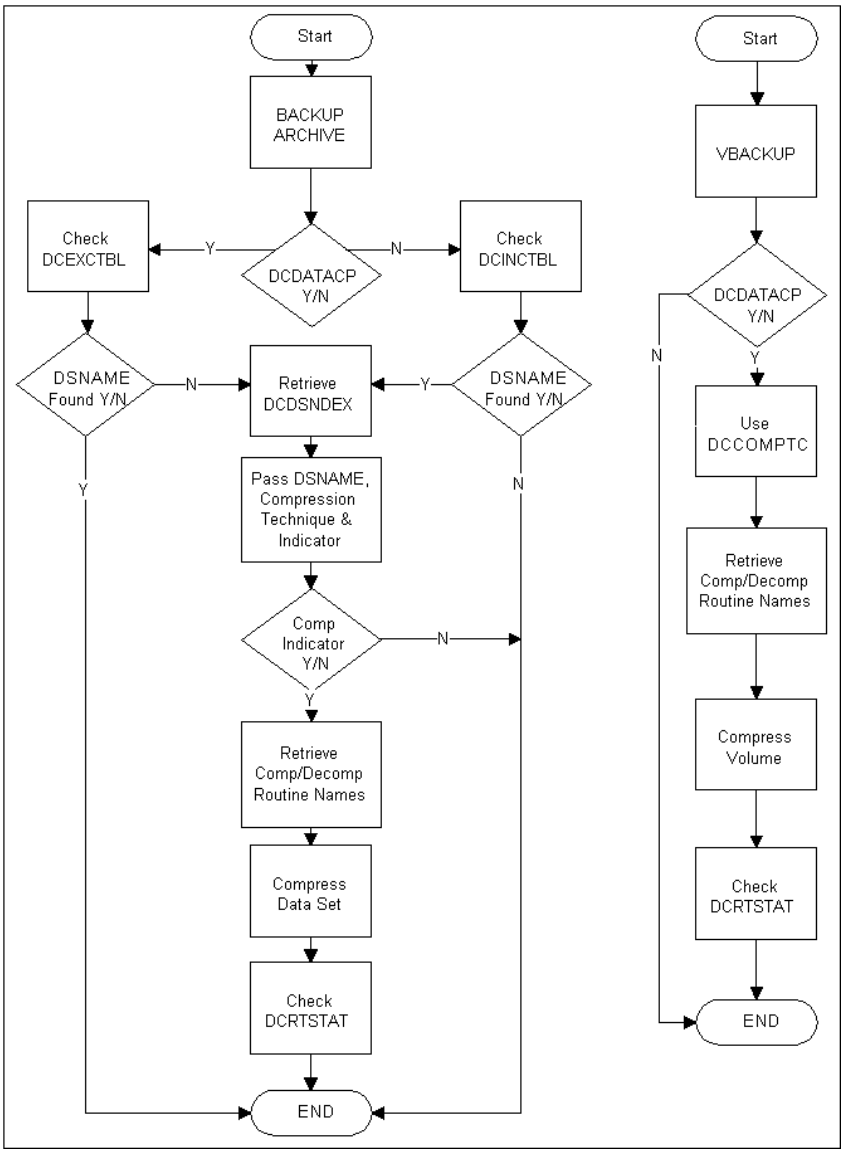
1. The new name is moved into a work area: LABS.NEW.PMRFILE
2. The original cluster name is compared to the original data component name. A difference is detected after the PMRFILE node. The differing characters are:

   `.DATA`
3. The differing characters are appended to the base cluster's new name in the work area to determine the component's new name:

   `LABS.NEW.PMRFILE.DATA`

Data Compression/Decompression

FIGURE 33. General flow of data compression



CA-Disk does data compression and decompression at the archive logical record level. Only the data after the eight-byte prefix is compressed (see the archive record format). A high- level control module is called before moving the archive record to the archive block. It decides whether or not to do data compression when it begins processing for each data set.

Sysparm DCDATACP is used to set the initial value of the compression indicator. If this is not a volume-level backup, the exclusion/inclusion table is referenced. If DCDATACP is specified with a value of Y but the data set is in the exclusion table, the compression indicator will be set to N (no). If sysparm DCDATACP is specified with a value of N but the data set is in the inclusion table, the compression indicator will be set to Y (yes). Inclusion and exclusion tables cannot be used at the same time.

The compression technique number is indicated by sysparm DCCOMPTC. The sysparm default is always selected for volume-level backup.

If user exit DCDSNDEX is active, the technique number, yes/no compression indicator, and the data set name are passed for the exit to make the decision. Each technique number corresponds to two exits—one for compression (DCCMPEXn) and one for decompression (DCDCPEXn). Both must be present at compression time.

If compression statistics are desired, sysparm DCRTSTAT can be specified with a value of Y for archive volume totals or D for data set totals and archive volume totals.

Each compressed record is returned with a two-byte prefix. The first character is X'EE'. It is for double-checking and indicates the record is in compressed format. The second character is X'Fn', n being any character A-F or 0-9, to indicate the compression routine used. This character allows multiple compression routines to be implemented for the same application. This is useful when you have knowledge of the data and know that specific routines handle specific kinds of data more efficiently. Once data has been compressed using a given compression technique, that technique must remain the same.

Data Compression Techniques

There are 16 data compression/decompression exits to chose from. Technique exits 0-7 are reserved for CA-Disk, while technique exits 8-F are available to the user. The following table is a list of the 16 exits:

TABLE 184. Data Compression Techniques available

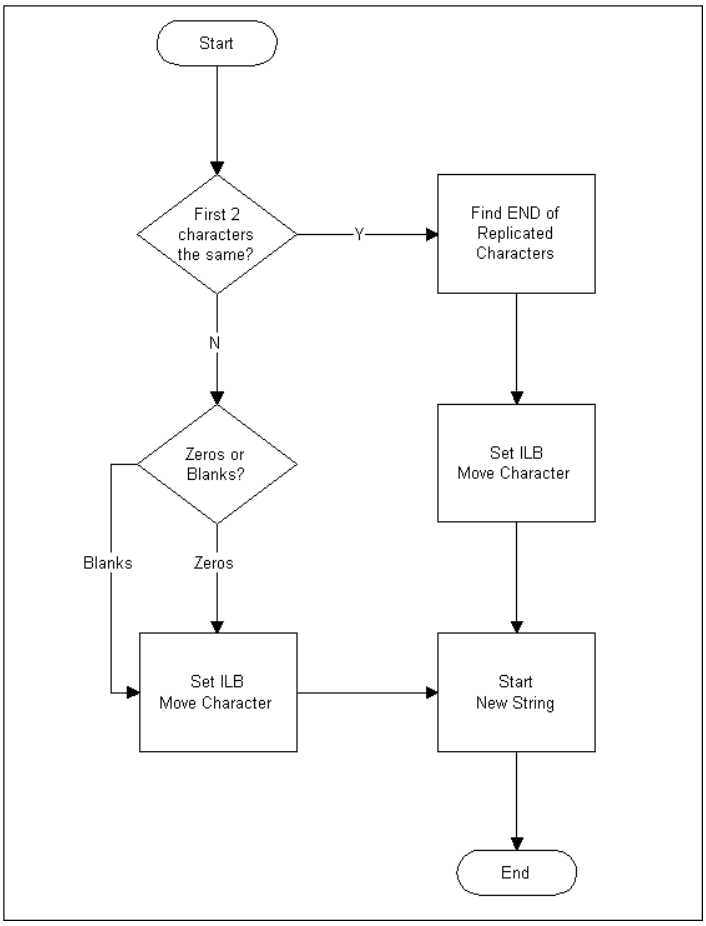
Technique	Compression Exit	Decompression Exit
0	DCCMPEX0-ADSUT371	DCDCPEX0-ADSUT372
1	DCCMPEX1-ADSUT410	DCDCPEX1-ADSUT411
2	DCCMPEX2	DCDCPEX2
3	DCCMPEX3	DCDCPEX3
4	DCCMPEX4	DCDCPEX4
5	DCCMPEX5	DCDCPEX5
6	DCCMPEX6	DCDCPEX6
7	DCCMPEX7	DCDCPEX7
8	DCCMPEX8	DCDCPEX8
9	DCCMPEX9	DCDCPEX9
A	DCCMPEXA	DCDCPEXA
B	DCCMPEXB	DCDCPEXB
C	DCCMPEXC	DCDCPEXC
D	DCCMPEXD	DCDCPEXD
E	DCCMPEXE	DCDCPEXE
F	DCCMPEXF	DCDCPEXF

Data Compression Technique 0 (DCCMPEX0)

This compression routine compresses replicated spaces, binary zeros and other replicated character strings. It starts with a 126-byte fixed-length character string (or the length of the data, whichever is shorter).

Typical flow of this process is illustrated in the figure below:

FIGURE 34. Data Compression Technique 0 Flowchart



INDICATOR/LENGTH BYTE

ILLLLLLL

I - String indicator bit

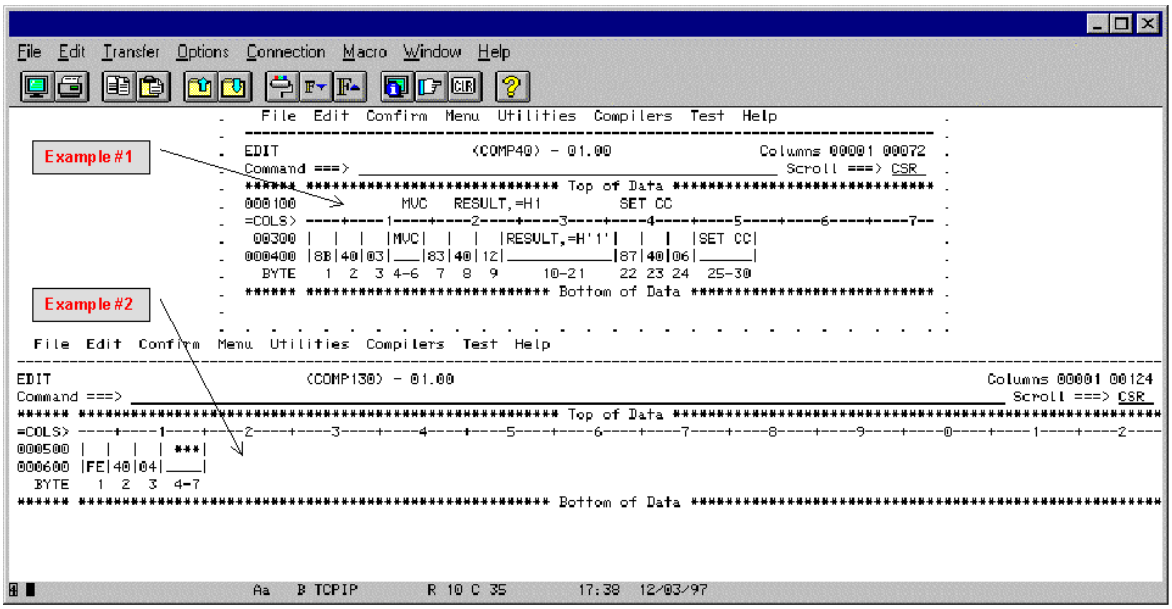
```
1 - Replicated string
0 - Non-replicated string
```

LLLLLLL - LENGTH BITS

The max string length is 126 characters. If there are more than 126 characters in a string, a new character string is started.

Due to the storage technique used, it is possible to compress almost all replicated character strings to 2 characters. For performance reasons, it is not practical to test for all possible replicated character strings. This routine specifically looks only for replicated spaces and binary zeros.

Example 1: - The following string will be compressed as shown. This example shows compressing 40 bytes to 30.



Example 2: - The second example shown above compresses 130 bytes to 7. The 126 spaces are compressed to a replicated string. The space and the 3 asterisks become a 4-byte literal.

Compression Results

Our TSO type packs averaged 55 percent savings in space. The CPU time almost doubled, but the elapsed time remained about the same. Our system packs averaged 30 percent savings in space. The CPU time increased by a fifth and the elapsed time increased by a third.

Overview of Space Allocation for Archival to DISK

CA-Disk determines the amount of space to allocate (the amount of space it needs on a disk pool volume) in the following manner. Quite frequently the primary output will be on disk and the duplicate will be on tape or a 3480 cartridge. The output archive type dictates the block size to be used. This, coupled with the density and length of tape being used, allows an easy and very accurate calculation of the tape capacity. (Since the ends of 3480 tape cannot be clipped off, the most common variable in the calculation has been removed!)

The optimal disk allocation is exactly the same capacity as its backup medium, such that when one is filled and both must be closed, nothing is wasted on the other medium. To do this on an exact equivalence, however, would require that very large amounts of free space be available. Therefore CA-Disk selects 1/16th of that value as the default primary and secondary space allocations. In many cases, an archive run will not need a full tape, and one

or two extents will be more than enough space. However, when a large archive run is made and the disk pool volumes contain plenty of free space, CA-Disk will obtain as many as possible of the 16 extents before closing the archive data set. This maintains high tape usage as well. Any excess (unused) disk space is, of course, released immediately when the data set is closed.

CA-Disk also considers the size of the data sets being archived when determining the amount of space to allocate. If the first data set to be archived is larger than the default primary space, the primary space is reset to the size of the data set. If free space equal to this new primary value is not available but the default value is, as a last resort CA-Disk will attempt to archive the input data set to a multivolume output data set.

Once the target archive data set is allocated and the first data set to be archived is copied to it, CA-Disk will continue to copy additional data sets (the second through nth) as long as there is sufficient space to hold them. To determine whether sufficient space is available, CA-Disk examines the unused space in the current extent, and checks to see if additional extents can be obtained. If there is not sufficient space, the current disk archive data set is closed and a new one is allocated, much like swapping to a new tape.

Space Allocation Algorithm

To begin the creation of a disk archive data set CA-Disk issues an enqueue against the CA-Disk DASD POOL to make sure that other CA-Disk jobs wait until allocation to a volume within the pool is complete. CA-Disk then saves the input data set size, and determines the minimum primary and secondary allocation requirements specified by sysparms SPACEPRI and SPACESEC. When these sysparms are set to their default values, minimum allocation requirements are determined by tape capacity in bytes divided by 16. Tape capacity is calculated based on device type and density to get bytes per foot and multiplied by the number of feet indicated by sysparms TAPEFEET or CARTFEET. The default for this calculation is TAPEFEET unless 3480s are indicated on the ARCHIVEC dd card. If SPACEPRI and SPACESEC are not set at their default values, the minimum allocation requirement is the value * 1024 * 1024. If packing is not used (sysparm ARCHPACKn) the input data set size becomes the minimum allocation requirement. Otherwise the input data set size is used only if it is greater than the calculated minimum value.

If the primary and copy cannot reside on the same volumes as indicated by sysparm ARCSEPC0 then the primary volume list is added to an exclude volume list used during volume selection to ensure they are not selected for the copy.

CA-Disk now begins to build a list of volumes, the volume with the largest amount of free space being first, the next largest being second and continuing until the number of bytes in the data set has been satisfied or until the maximum number of volumes, or the number of volumes in the pool if less than 35, has been reached.

The allocation satisfied the first time through is the primary amount, if successful, the volume is added to the list and to the exclusion list, and the allocation to satisfy becomes the secondary allocation. If the primary amount cannot be satisfied with the first volume returned, and the free space associated with that volume is less than the minimum allocation amount for primary, an abend will occur. If the available free space is greater than the calculated minimum allocation, the data set size is divided by 35, or the number of volumes in the pool if less than 35, and becomes the new minimum primary amount. If the returned free space is still less than the new minimum primary amount, an abend will occur. If not, the primary amount will be set to the returned free space, and will be subtracted from the data set size. The remaining bytes will be divided by 35 (or the number of volumes in the pool if less than 35) minus one to become the new minimum secondary amount. If the new minimum secondary amount is greater than the calculated secondary allocation, the value to satisfy is the minimum secondary allocation. The volser is added to the volume and exclusion lists and processing continues.

The secondary allocations begin when the primary has been satisfied. Only when a volume's free space cannot satisfy the secondary allocation amount or if the input data set size has been satisfied will the volume list be

complete. If there are no volumes in the list at this time, an abend will occur. If there is a list of volumes, they are returned to the calling program so that allocation can take place. When allocation is complete a dequeue is issue against the CA-Disk DASD POOL.

General Comparison of Allocation in Prior Releases

1. The primary allocation value is still set to 1/16 of tape/cart, or the value of SPACEPRI if specified. (No change here.)
2. Release 7.6 and prior CA-Disk takes the first volume from the pool that satisfies SPACEPRI for contiguous free space. If more than one CA-Disk job looks at the same pool of volumes, each can examine volume space at the same time, unaware that the other sees the identical “space status” for the volume. The first job to allocate should be successful, but the others risk even a primary allocation error. If all concurrent primaries succeed, the probability of getting a secondary allocation is certainly decreased for all of the jobs!

   Release 7.7 or higher begins by serializing the examination of the archive pool and the actual allocation of the space. This is done by an ENQ on Major=DMSOS, Minor=ARCHPOOL when the pool is examined, with the DEQ done after the allocation has taken place. Release 7.7 or higher also examines all volumes in the pool and selects the volume with the greatest contiguous free space. This ensures the primary allocation and increases the probability of getting extra extents if needed.

   By serializing the selection and allocation among concurrent CA-Disk jobs, each job will see a different “space status” for the pool volumes, increasing the distribution across different volumes. If different primary volumes are selected (with increased probability), contention for secondary extents is reduced.
3. If the size of the input data set is larger than the size in item 1 above (1/16 tape or SPACEPRI), the data set size is compared to the largest contiguous free space that was found. If the free space is big enough, the primary is set to the data set size and allocated. This is the same as in prior releases, except that the probability is greater that the data set will fit, because the volume with the largest contiguous free space was selected first.

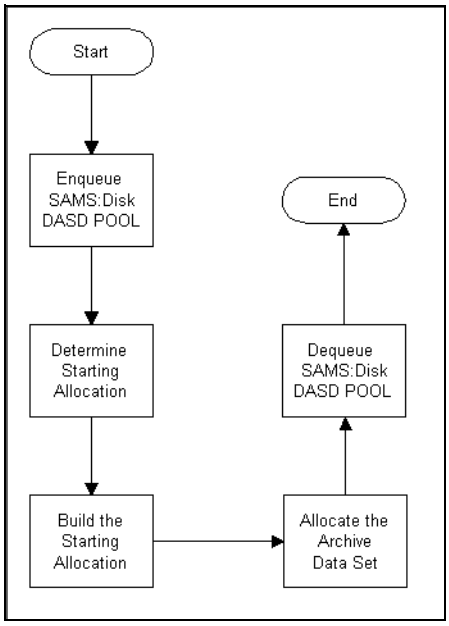
   If the data set size is greater than the largest contiguous free space, additional volumes are selected with the 2nd, 3rd, and so on largest contiguous free space until the size of the data set can be satisfied, and a multivolume allocation is done. If the pool is exhausted without finding enough contiguous space to satisfy the data set, the archive is still attempted on the theory that additional extents from non contiguous free space on each of the selected volumes can be able to satisfy the request. If they do not, a B37 abend for the data set being processed will occur. It will be bypassed as “too large” for the archive pool, and processing continues with the next data set. Again, the significant difference in 7.7 or higher is that each volume selected from the pool is the one with the largest contiguous free space, the next largest, and so on. In 7.6, it was a sequential search of the pool, taking any volume that met the minimum secondary space need. Now we get the volume with the most space that meets that need.

Creating a Disk Archive Data Set

1. CA-Disk issues an enqueue against the CA-Disk DASD POOL, major name = DMSOS, minor name = ARCHPOOL.
2. Determine the primary and secondary values for the archive data set. If sysparms SPACEPRI and SPACESEC are defaulting they are recalculated to tape capacity in bytes divided by 16. If a value is specified, it is multiplied by 1024 and again by 1024 (SPACEPRI*1024*1024).
3. Determine the starting primary allocation. Primary will be the above calculated SPACEPRI value unless data sets are not packed and the data set size is less than the calculated SPACEPRI value, in which case primary will be set to the data set size. If SPACEPRI is still smaller than the data set size, primary will be set to the data set size.

4. Begin an exclusion list of volsers to ensure integrity as selections are being made. If this archive data set is for the copy, and sysparm ARCSEPC0 is set to yes, then the list of primary volsers are added to the exclusion list to ensure they are not re-selected to be the copies.
5. Set the desired number of bytes to the calculated primary value.
6. Build the volume list. Continue building until the limit of 35 (or the number of volumes in the pool if less than 35) has been reached, or the number of bytes in the initial data set has been satisfied in either single or multiple volumes. The building of the volume list abends under the following conditions:
   - If the returned volume does not satisfy the desired number of bytes, and it is the first volume, and returned free space does not satisfy SPACEPRI needs.
   - If the returned volume does not satisfy the desired number of bytes, and it is the first volume, and the data set size divided by 35, or the number of volumes in the pool if less than 35, does not satisfy the minimum primary allocation specification.
   - If a volume was not found that was not in the exclude list or of the correct device type or if it's largest contiguous space could not satisfy the desired number of bytes requested.
7. Return the list of volumes.
8. Allocate the archive data set.
9. Issue a dequeue against the CA-Disk DASD POOL.

This process is illustrated below:



PDS Compression Threshold Calculations

Calculating compression thresholds for a PDS is divided up into two categories:

1. Directory Used Space Percentage

2. Member Data Used Space Percentage

Directory Used Space Percentage

$$PCT\ Used = \frac{((DBUSE - 1) \times 256 + NODBD - 14) \times 100}{(DBUSE + DMBTY) \times 256}$$

or

$$PCT = \frac{(total\ bytes\ used\ in\ all\ full\ DB) + (total\ bytes\ used\ in\ last\ used\ DB) - (14\ bytes\ for\ the\ directory\ EOF)}{total\ bytes\ allocated}$$

Where:

PCT USED - percentage of used directory block space calculated in bytes

DBUSE - number of used directory blocks

NOBDB - number of bytes used in the last used directory block

DBMTY - number of empty directory blocks

Member Data Used Space Percentage

$$PCT\ Used = \frac{((DSNTRK - DIRTRK) \times TRKLN - (DSNBAL + DIRBAL)) \times 100}{(EXTTRK - DIRTRK) \times TRKLN + DIRBAL}$$

or

$$PCT = \frac{(total\ bytes\ of\ all\ full\ data\ trks) + (data\ bytes\ on\ trk\ sharing\ the\ dir.) - (unused\ bytes\ on\ last\ data\ trk)}{total\ bytes\ for\ all\ tracks\ allocated}$$

Where:

PCT USED - percentage of member data used space calculated in bytes

DSNTRK - relative track of the last data record

DIRTRK - relative track of the directory end-of-file

TRKLN - track capacity of the device

DSNBAL - track balance of the last data record or DS1TRBAL

DIRBAL - track balance of the directory end-of-file

EXTTRK - relative track of the last allocated track

PDS Compression Space Calculations

Calculating compression space for a PDS is divided up into three categories:

1. Directory Block Allocation
2. Member Data Space Allocation
3. Secondary Space Allocation

Directory Block Allocation

$$DIRBLKS = \frac{(((DBUSE - 1) \times 256 + NOBDB - 14) \times 100) + (14 \times DD)}{DD \times 256}$$

Where:

DIRBLKS - number of directory blocks to be allocated

DBUSE - number of used directory blocks

NOBDB - number of bytes used in the last used directory block

DD - requested percentage of directory space to be used after reallocation

Member Data Space Allocation

$$EXTRKS = \frac{\frac{((DS1TRK - DIRTRK) \times TRKLN + DIRBAL - DS1BAL) \times 100}{\text{requested percentage of member space to be used after realloc}} - DIRBAL}{\text{track capacity of the device}} + DIRTRK$$

Where:

EXTRKS - total number of tracks to allocate for primary space, including space for the directory

DS1TRK - relative track of the last record written for the compressed PDS.

DIRTRK - calculated relative track for the new directory's end-of-file

DS1BAL - track balance of the last record written for the compressed directory

PDS DIRBAL - calculated track balance of the new directory's end-of-file

TRKLN - track capacity of the device

Secondary Space Allocation

$$SECTRKS = \frac{(MBRTRK \times TRKLN) + (TRKLN - MBRBAL)}{PDS\ SS \times TRKLN}$$

Where:

SECTRKS - number of tracks to request for the secondary allocation amount

TRKLN - track capacity of the device

MBRTRK - relative track of the last record written for the compressed PDS.

MBRBAL - track balance of the last record written for the compressed PDS.

PDS SS - requested percentage of secondary space as a function of the member space used after compression

Notes:

- Percentages are rounded up to the nearest whole point.
- Percentages are based on data size prior to compression.
- One directory block = 256 bytes.
- Allocations are rounded up to the nearest track.

CHAPTER 13

Record Formats

This chapter provides Record Format information on:

- Files Data Set Logging records
- CA-Disk Report records
- Subfiles of the Files Data Set

TABLE 185. List of Record Formats

Name	Page	Name	Page	Name	Page	Name	Page
Files Logging Format		**Report Record Formats**				**Subfiles Record Formats**	

TABLE 185. List of Record Formats

Name	Page	Name	Page	Name	Page	Name	Page
FILES LOGGING	467	ARCHDSNVOL	442	RPTNAMVDIC	400	ARCHCMDS	458
		ARCHVOLDSN	444	RPTNATTRBS	402	ARCHVOLS	459
		MIGRDSNVOL	446	RPTNBMVDIC	404	DASDSPCB	460
		MIGRVOLDSN	448	RPTNDAYWEK	419	DMSPARMS	461
		RPTDSNEX	456	RPTNPODUMP	433	DMSPOOLS	461
		RPTNDEVDSB	406	RPTNPOTTRD	433	DSNINDEX	462
		RPTNDISTRS	408	RPTNPOXREF	433	MIGRECAT	465
		RPTNDSINDX	415	RPTNPO1REF	433	RACFENCD	465
		RPTNDSNDSB	410	RPTNPO2REF	433	RESTCMDS	466
		RPTNDSNX	412	RPTNRACFDN	434	RETEXCLD	467
		RPTNDSUTIL	417	RPTNRACFPF	436		
		RPTNFREESP	421	RPTNSVDICT	438		
		RPTNLAYOUT	423	RPTNVOLDSB	440		
		RPTNMAPPER	425	RPTNVOLX	414		
		RPTNMEMBER	427	RPTPLST1	454		
		RPTNMVDICT	429	RPTPLST2	451		
		RPTNPDSTAT	431	RPTSTATS	450		
		RPTNPOCCHH	433	VSAMCLDATA	452		
		RPTNALLOCS	398				

Report Record Format - (RPTNALLOCS)

Volume Allocation Summary Report — ALLOCS

```
ALLOCS
*
*      Mapping of Report record used to print VOLUME
*      ALLOCATION SUMMARY report
*
DETREC   DS    0F
DETCONLN DS    H            RECORD LENGTH, 74 BYTES
         DS    H            (FILLER, ZEROS)
DETTABLE DS    CL10         REPORT NAME 'RPTNALLOCS'
DETTYPE  DS    XL2          DEVICE TYPE
DETSERAL DS    CL6          VOLUME SERIAL
DETDSCB0 DS    H            DSCB 0 COUNT
DETDSCB1 DS    H            DSCB 1 COUNT
DETDSCB2 DS    H            DSCB 2 COUNT
DETDSCB3 DS    H            DSCB 3 COUNT
DETDSCB4 DS    H            DSCB 4 COUNT
DETDSCB5 DS    H            DSCB 5 COUNT
DETDSCB6 DS    H            DSCB 6 COUNT
DETDEVTK DS    XL4          TRACKS PER DEVICE
DETTKALC DS    XL4          TRACKS ALLOCATED
DETTKAVL DS    XL4          TRACKS AVAILABLE
DETFREXT DS    XL2          EXTENTS
DETTKIDL DS    XL4          TRACKS IDLE
DETFRSPL DS    XL4          1ST LARGEST FREE SPACE
DETFRSPS DS    XL4          2ND LARGEST FREE SPACE
DETFRSPT DS    XL4          3RD LARGEST FREE SPACE
DETVOLAD DS    CL4          VOLUME ADDRESS
DETSMS   DS    CL4          SMS VOLUME STATUS
DETRECLN EQU   *            DETREC LENGTH OF RECORD, 74 BYTES
*
```

Table Used for Printing ALLOCS Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT VOLUMES=/
      REPORT ALLOCS
```

FIGURE 35. Sample Parmlib Member RPTNALLO

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                  .PARMLIB(RPTNALLO) - 01.02          Columns 00001 00072
 Command ===> _                                            Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132050'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129   PAGE NO'
 000005 'H1 K049   041   V O L U M E   A L L O C A T I O N   S U M M A R Y'
 000006 'H2 4132   001   STANDARD HEADER - LINE 2'
 000007 'H3 K001   001132'
 000008 'H4 K015   009   DEVICE UNIT SMS'
 000009 'H4 K046   027   ----DATA SET CONTROL BLOCKS (DSCB) COUNTS-----'
 000010 'H4 K027   075   -----------TRACKS----------'
 000011 'H4 K030   104   ---------FREE SPACE----------'
 000012 'H5 K006   001   VOLUME'
 000013 'H5 K015   010   TYPE  ADDR STAT'
 000014 'H5 K046   027   FREE     F1     F2     F3     F4     F5     F6'
 000015 'H5 K027   075   DEVICE  ALLOC  AVAIL   IDLE'
 000016 'H5 K027   104   EXTENTS       THREE LARGEST'
 000017 'H6 K040   001   ------  ------ ---- ----  --------------'
 000018 'H6 K040   041   --------------------------------  ------'
 000019 'H6 K040   081   ---------------------  -----------------'
 4 ■                       Aa   A TCPIP        R 4  C 15        15:29  12/04/97
```

Report Record Format — RPTNAMVDIC

A-FORM of MVDICT Report

```
AMVDIC
*
*      Mapping of report record used to print A-FORM of MULTIPLE
*      VOLUME DICTIONARY report with owner
*      Note: The RPTDSORG field has
*         1. X'02' for PDS, X'FD' for HFS, and X'FE' for PDSE
*         2. X'40' for PS and X'FC' for PSE
*
AMVDREC  DS    0F
RPTRDW   DS    2H
RPTTABLE DC    CL10            'RPTNAMVDIC'
RPTAINFO DS    0CL86
RPTDSNAM DS    CL44            DATA SET NAME
RPTVOLNO DS    CL6             VOLUME SERIAL
RPTDEVCL DS    XL2             DEVICE CLASS
RPTDSORG DS    XL2             DSORG
RPTRECFM DS    XL1             RECFM
RPTBLKSI DS    XL2             BLOCKSIZE
RPTLRECL DS    XL2             LRECL
RPTSCALO DS    XL1             SECONDARY ALLOCATION
RPTSCQTY DS    XL3             AMOUNT OF SECONDARY ALLOCATION
RPTTKSUM DS    XL3             TOTAL TRACKS ALLOCATED
RPTCREDT DS    XL3             CREATION DATE
RPTEXPDT DS    XL3             EXPIRATION DATE
RPTDSIND DS    XL1             SECURITY
RPTPRMAL DS    XL2             PRIMARY ALLOCATION
RPTIDLET DS    XL2             IDLE TRACKS
RPTVOLAD DS    CL4             VOLUME ADDRESS
RPTCATLG DS    CL1             CATALOG STATUS
RPTLUSDT DS    XL3             LAST USED DATE
RPTNOEXT DS    XL1             NUMBER OF EXTENTS
RPTAOWNR DS    CL11            OWNER
```

Table Used to Print MVD Report by Owner

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT VOLUMES=/
      REPORT MVDICT
```

FIGURE 36. Sample Parmlib Member RPTNAMVD

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARML(B(RPTNAMVD) - 01.01          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132101'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129'
 000005 'H1 K033   032   M U L T I P L E   V O L U M E   D'
 000006 'H1 K037   065    A T A   S E T   D I C T I O N A R Y'
 000007 'H2 4132   001   STANDARD HEADER - LINE 2'
 000008 'H3 K001   001132'
 000009 'H4 K006   046   DEVICE'
 000010 'H4 K029   081    --ALLOCATION---  --TRACKS---'
 000011 'H5 K013   001   DATA SET NAME'
 000012 'H5 K005   030   OWNER'
 000013 'H5 K006   039   VOLUME'
 000014 'H5 K016   046    TYPE  CAT DSORG'
 000015 'H5 K029   063   RECFM BLKSZ LRECL  TYPE  PRIM'
 000016 'H5 K016   094   SEC  ALLOC  IDLE'
 000017 'H5 K021   112   EXT CREDT EXPDT USEDT'
 000018 'H6 K040   001   ------------------------ ----------- --'
 000019 'H6 K040   041   ---- ------ --- ----- ----- ----- ----- '
 4 ▌                       Aa   A TCPIP        R 4  C 15        15:47  12/04/97
```

Report Record Format — (RPTNATTRBS)

Volume Attribute Summary Report

```
ATTRBS
*
*       Mapping of report record used to print VOLUME ATTTRIBUTE
*       report
*
DETREC   DS    0F
DETCONLN DS    H            RECORD LENGTH, 60 BYTES
         DS    H            (FILLER, ZEROS)
DETTABLE DS    CL10         REPORT NAME, 'RPTNATTRBS'
DETDEVCL DS    XL2          DEVICE CLASS
DETVOLNO DS    CL6          VOLUME SERIAL
DETADDRS DS    CL4          UNIT ADDRESS
DETDEVST DS    XL1          DEVICE STATUS
DETVOLST DS    XL1          VOLUME STATUS
DETDSOIS DS    H            ISAM DATA SET COUNT
DETDSOPS DS    H            PS DATA SET COUNT
DETDSODA DS    H            DA DATA SET COUNT
DETDSOPO DS    H            PO DATA SET COUNT
DETDSOAM DS    H            VSAM DATA SET COUNT
DETDSOOT DS    H            UNKNOWN DATA SET COUNT
DETDSTOT DS    H            TOTAL DATA SETS
DETDSCBV DS    H            DSCB COUNT
DETDSCBA DS    H            ALLOCATED DSCB COUNT
DETDEVTK DS    XL4          TOTAL TRACKS FOR DEVICE
DETALOTK DS    XL4          TOTAL TRACKS ALLOCATES
DETALPCT DS    H            PERCENT ALLOCATED
DETSMS   DS    CL4          SMS VOLUME STATUS
DETRECLN EQU   *            DETREC LENGTH OF RECORD, 60 BYTES
*
```

Table Used to Print ATTRIBS Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT VOLUMES=/
      REPORT ATTRBS
```

FIGURE 37. Sample Parmlib Member RPTNATTR

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                 .PARMLIB(RPTNATTR) - 01.02          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132'
 000002 'H113132   001'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129   PAGE NO'
 000005 'H1 K047   043   V O L U M E   A T T R I B U T E   S U M M A R Y'
 000006 'H2 4132   001'
 000007 'H3 K001   001132'
 000008 'H4 K039   009   DEVICE UNIT  SMS    MOUNT    ALLOCATION'
 000009 'H4 K039   050   ---------DATA SET ORGANIZATION---------'
 000010 'H4 K009   092   --TOTAL--'
 000011 'H4 K013   104   ----DSCBS----'
 000012 'H4 K013   120   ---TRACKS----'
 000013 'H5 K006   001   VOLUME'
 000014 'H5 K004   010   TYPE'
 000015 'H5 K031   016   ADDR  STAT   STATUS   ATTRIBUTE'
 000016 'H5 K037   052   PS     DA     PO     IS     AM     UN'
 000017 'H5 K009   092   DATA SETS'
 000018 'H5 K013   104   VTOC    ALLOC'
 000019 'H5 K013   120   DEVICE  ALLOC'
 4 ▌                       Aa   A TCPIP        R 4  C 15        9:01  12/05/97
```

Report Record Format — (RPTNBMVDIC)

B-FORM MVDICT Report

```
BMVDIC
*
*      Mapping of report record used to print B-FORM of MULTIPLE
*      VOLUME DICTIONARY report with owner/project
*      Note: The RPTDSORG field has
*         1. X'02' for PDS, X'FD' for HFS, and X'FE' for PDSE
*         2. X'40' for PS and X'FC' for PSE
*
BMVDREC  DS    0F            B FORM OF MVD REPORT (BMVD)
RPTBRDW  DS    2H
RPTBTABL DC    CL10'RPTNBMVDIC'
RPTBOWNR DS    CL20          OWNER
RPTBPROJ DS    CL06          PROJECT NUMBER
RPTBINFO DS    0CL86
RPTDSNAM DS    CL44          DATA SET NAME
RPTVOLNO DS    CL6           VOLUME SERIAL
RPTDEVCL DS    XL2           DEVICE CLASS
RPTDSORG DS    XL2           DSORG
RPTRECFM DS    XL1           RECFM
RPTBLKSI DS    XL2           BLOCKSIZE
RPTLRECL DS    XL2           LRECL
RPTSCALO DS    XL1           SECONDARY ALLOCATION
RPTSCQTY DS    XL3           AMOUNT OF SECONDARY ALLOCATION
RPTTKSUM DS    XL3           TOTAL TRACKS ALLOCATED
RPTCREDT DS    XL3           CREATION DATE
RPTEXPDT DS    XL3           EXPIRATION DATE
RPTDSIND DS    XL1           SECURITY
RPTPRMAL DS    XL2           PRIMARY ALLOCATION
RPTIDLET DS    XL2           IDLE TRACKS
RPTVOLAD DS    CL4           VOLUME ADDRESS
RPTCATLG DS    CL1           CATALOG STATUS
RPTLUSDT DS    XL3           LAST USED DATE
RPTNOEXT DS    XL1           NUMBER OF EXTENTS
```

Table Used to Print MVDICT Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT VOLUMES=/
      REPORT MVDICT
```

FIGURE 38. Sample Parmlib Member RPTNBMVD

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                  .PARMLIB(RPTNBMVD) - 01.01          Columns 00001 00072
 Command ===> _                                            Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058051001132116020'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129'
 000005 'H1 K032   032   M U L T I P L E   V O L U M E   '
 000006 'H1 K038   064   D A T A   S E T   D I C T I O N A R Y'
 000007 'H2 4132   001   STANDARD HEADER - LINE 2'
 000008 'H3 K001   001132'
 000009 'H4 K005   001   OWNER'
 000010 'H4 C020001008      OWNER'
 000011 'H5 K006   034   DEVICE'
 000012 'H5 K004   052   DATA'
 000013 'H5 K029   081    --ALLOCATION---  --TRACKS---'
 000014 'H6 K013   001   DATA SET NAME'
 000015 'H6 K035   027   VOLUME  TYPE   PROJ  CAT PROT DSORG'
 000016 'H6 K029   063   RECFM BLKSZ LRECL  TYPE  PRIM'
 000017 'H6 K016   094   SEC  ALLOC  IDLE'
 000018 'H6 K021   112   EXT CREDT EXPDT USEDT'
 000019 'H7 K040   001   ------------------------ ------ ------ '
 4 ▌                       Aa   A TCPIP        R 4  C 15        9:15  12/05/97
```

Report Record Format — (RPTNDEVDSB)

BIlling Device Report

```
DEVDSB
*
*       Mapping of report record used to print BILLING DEVICE
*       report
*
DEVREC   DS    0F
         DC    H'84,0',CL10'RPTNDEVDSB'
DEVENTRY DS    0CL56
DEVCLASS DS    XL2            DEVICE CLASS
DEVOLCNT DS    H              VOLUME COUNT
DEVDSCNT DS    F              DATA SET COUNT
DEVACCUM DS    F              ACCUMULATED AMOUNT FOR DEVICE
DEVTOTAL DS    F              TOTAL $ AMOUNT FOR DEVICE
DEVDESCR DC    CL40' '        DESCRIPTION
DEVUNITS DC    CL9'K/B DAYS ' or CL9'TRK DAYS '
DSBRELSE DC    CL6' '         RELEASE 9.9 = C' 9.9'
```

Table Used for Printing Billing Device Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
EXTEND VOLUME=,
   DSNAME=,
   SIMULATE
```

FIGURE 39. Sample Parmlib Member RPTNDEVD

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                  .PARMLIB(RPTNDEVD) - 01.01          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058053001132'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K010   123              '
 000004 'H1 K011   071     PAGE     '
 000005 'H1 N004   078'
 000006 'H1 K042   028          DASD BILLING BY DEVICE TYPE        '
 000007 'H2 4132   001   STANDARD HEADER - LINE 2'
 000008 'H2 K013   120                '
 000009 'H2 K006   070   SAMS:DISK'
 000010 'H2 C006066076   RELEASE NO'
 000011 'H3 K001   001132'
 000012 'H4 K011   001   DEVICE TYPE'
 000013 'H4 K007   035   VOLUMES'
 000014 'H4 K009   045   DATA SETS'
 000015 'H4 C009057058      K/B TRK M/B'
 000016 'H4 K012   070   TOTAL CHARGE'
 000017 'H5 K040   001   ------------------------------    ------'
 000018 'H5 K041   041   -   ---------    --------    ------------'
 000019 'D1 C040017001030   DESCRIPTION'
 4 ▌                       Aa   A TCPIP        R 4  C 15        9:28  12/05/97
```

Report Record Format — (RPTNDISTRS)

Volume Distribution Summary Report

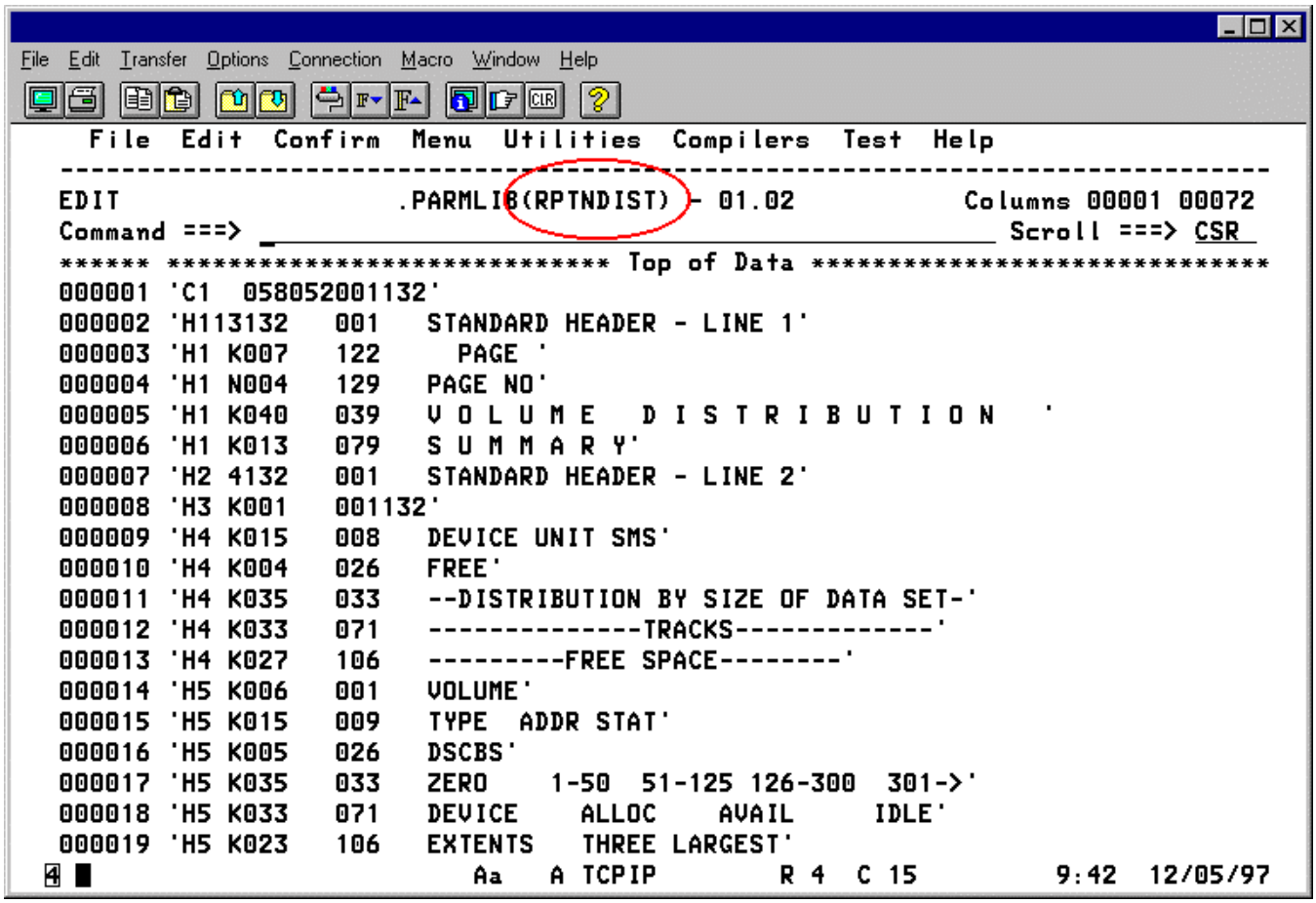
```
DISTRS
*
*       Mapping of report record used to print VOLUME
*       DISTRIBUTION report
*
DETREC   DS    0F
DETCONLN DS    H            RECORD LENGTH, 74 BYTES
         DS    H            (FILLER, ZEROS)
DETTABLE DS    CL10         REPORT NAME, 'RPTNDISTRS'
DETTYPE  DS    XL2          DEVICE TYPE
DETSERAL DS    CL6          VOLUME SERIAL
DETDSCB0 DS    H            DSCB 0 COUNT
DETLT1   DS    H            COUNT OF DATA SETS OF 1 TRACK
DETLT51  DS    H            COUNT OF DATA SETS LT 51 TRACKS
DETLT126 DS    H            COUNT OF DATA SETS LT 126 TRACKS
DETLT301 DS    H            COUNT OF DATA SETS LT 301 TRACKS
DETGT300 DS    H            COUNT OF DATA SETS GT 300 TRACKS
         DS    H            (FILLER)
DETDEVTK DS    XL4          TRACKS PER DEVICE
DETTKALC DS    XL4          TRACKS ALLOCATED
DETTKAVL DS    XL4          TRACKS AVAILABLE
DETFREXT DS    XL2          FREE EXTENTS
DETTKIDL DS    XL4          TRACKS IDLE
DETFRSPL DS    XL4          1ST LARGEST FREE SPACE
DETFRSPS DS    XL4          2ND LARGEST FREE SPACE
DETFRSPT DS    XL4          3RD LARGEST FREE SPACE
DETVOLAD DS    CL4          VOLUME ADDRESS
DETSMS   DS    CL4          SMS VOLUME STATUS
DETRECLN EQU   *            DETREC LENGTH OF RECORD, 74 BYTES
*
```

Table Used for Printing DISTRS Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT VOLUMES=/
      REPORT DISTRS
```

FIGURE 40. Sample Parmlib Member RPTNDIST



```
File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
--------------------------------------------------------------------------------
EDIT                    .PARMLIB(RPTNDIST) - 01.02          Columns 00001 00072
Command ===> _                                              Scroll ===> CSR
****** ***************************** Top of Data ******************************
000001 'C1  058052001132'
000002 'H113132   001   STANDARD HEADER - LINE 1'
000003 'H1 K007   122     PAGE '
000004 'H1 N004   129   PAGE NO'
000005 'H1 K040   039   V O L U M E   D I S T R I B U T I O N   '
000006 'H1 K013   079   S U M M A R Y'
000007 'H2 4132   001   STANDARD HEADER - LINE 2'
000008 'H3 K001   001132'
000009 'H4 K015   008   DEVICE UNIT SMS'
000010 'H4 K004   026   FREE'
000011 'H4 K035   033   --DISTRIBUTION BY SIZE OF DATA SET-'
000012 'H4 K033   071   --------------TRACKS-------------'
000013 'H4 K027   106   ---------FREE SPACE--------'
000014 'H5 K006   001   VOLUME'
000015 'H5 K015   009   TYPE  ADDR STAT'
000016 'H5 K005   026   DSCBS'
000017 'H5 K035   033   ZERO    1-50  51-125 126-300  301->'
000018 'H5 K033   071   DEVICE    ALLOC    AVAIL     IDLE'
000019 'H5 K023   106   EXTENTS   THREE LARGEST'
4 _                       Aa   A TCPIP        R 4  C 15        9:42  12/05/97
```

Report Record Format — (RPTNDSNDSB)

Billing by User Code

```
DSNDSB
*
*      Mapping of report record used to print BILLING BY USER CODE
*
DETREC   DS    0F
         DC    H'159,0',CL10'RPTNDSNDSB'
DETDSBLR DS    0CL86
DETUSRCD DS    CL12            USER CODE
DETVOLNO DS    CL6             VOLUME SERIAL
DETDSNAM DS    CL44            DATA SET NAME
DETACCUM DS    F               ACCUMULATE AMOUNT
DETNOSCN DS    H               NUMBER OF SCANS
DETCREDT DS    XL3             CREATION DATE
DETSCNDT DS    XL3             LAST SCAN DATE
DETUSEDT DS    XL3             LAST USED DATE
DETFLAGS DS    XL1             FLAGS
DETDEVCL DS    XL2             DEVICE CLASS
DETALLOC DS    H               CURRENT TRACKS ALLOCATED
DETMAXUS DS    H               LARGEST NUMBER OF TRKS ALLOCATED
         DS    H
DETRATE  DS    F               BILLING RATE
DETCOST  DS    F               COST
DETNOSCR DS    H               NUMBER OF TIMES SCRATCHED
DETUSRDS DS    CL40            USER DESCRIPTION
DETUNITS DS    CL9             BILLING UNITS
```

Table Used for Printing Billing by User Code

The following figure displays the first page of the source table used for printing when the following command is executed:

```
EXTEND VOLUME=,
   DSNAME=,
   SIMULATE
```

FIGURE 41. Sample Parmlib Member RPTNDSND

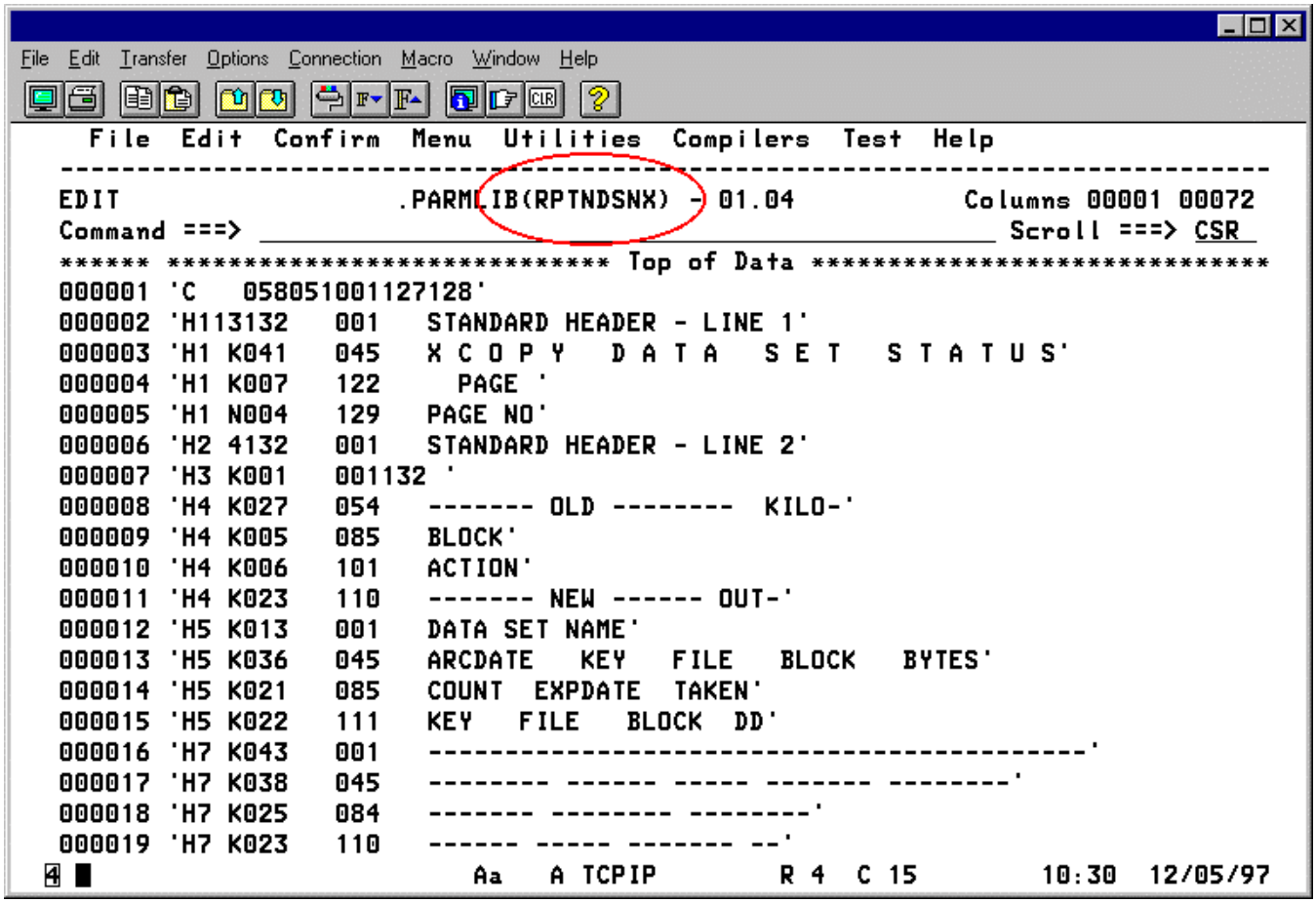
```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                  .PARMLIB(RPTNDSND) - 01.01          Columns 00001 00072
 Command ===> _                                            Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  05805000113214 7012'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K049   043   D A S D   B I L L I N G   B Y   U S E R   C O D E'
 000004 'H1 K007   122     PAGE '
 000005 'H1 N004   129   PAGE NO'
 000006 'H2 4132   001   STANDARD HEADER - LINE 2'
 000007 'H3 K001   001132'
 000008 'H4 K011   001   USER CODE -'
 000009 'H4 C012001013      CODE'
 000010 'H4 K013   030   DESCRIPTION -'
 000011 'H4 C040097044      DESCRIPTION'
 000012 'H5 K001   001132'
 000013 'H6 K006   043   DEVICE'
 000014 'H6 K015   081   -----TRACKS----'
 000015 'H7 K013   001   DATA SET NAME'
 000016 'H7 K026   033   VOLUME     TYPE   CREDATE '
 000017 'H7 K026   060   USEDATE   LAST SCAN  ALLOC'
 000018 'H7 K008   088   MAX USED'
 000019 'H7 K016   108   RATE            '
 4 ▌                       Aa   A TCPIP        R 4  C 15        9:48  12/05/97
```

Report Record Layout — (RPTNDSNX)

The following figure displays the first page of the source table used for printing when the following command is executed:

```
XCOPY FLAG,CHKFLAG,TDSN=
```

FIGURE 42. Sample Parmlib Member RPTNDSNX



```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTNDSNX) - 01.04          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C   058051001127128'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K041   045   X C O P Y   D A T A   S E T   S T A T U S'
 000004 'H1 K007   122     PAGE '
 000005 'H1 N004   129   PAGE NO'
 000006 'H2 4132   001   STANDARD HEADER - LINE 2'
 000007 'H3 K001   001132 '
 000008 'H4 K027   054   ------- OLD --------  KILO-'
 000009 'H4 K005   085   BLOCK'
 000010 'H4 K006   101   ACTION'
 000011 'H4 K023   110   ------- NEW ------ OUT-'
 000012 'H5 K013   001   DATA SET NAME'
 000013 'H5 K036   045   ARCDATE   KEY   FILE   BLOCK   BYTES'
 000014 'H5 K021   085   COUNT  EXPDATE  TAKEN'
 000015 'H5 K022   111   KEY   FILE   BLOCK  DD'
 000016 'H7 K043   001   ----------------------------------------'
 000017 'H7 K038   045   -------- ------ ----- ------- --------'
 000018 'H7 K025   084   ------- -------- --------'
 000019 'H7 K023   110   ------ ----- ------- --'
 4 ■                       Aa   A TCPIP        R 4  C 15        10:30  12/05/97
```

Record Report Format

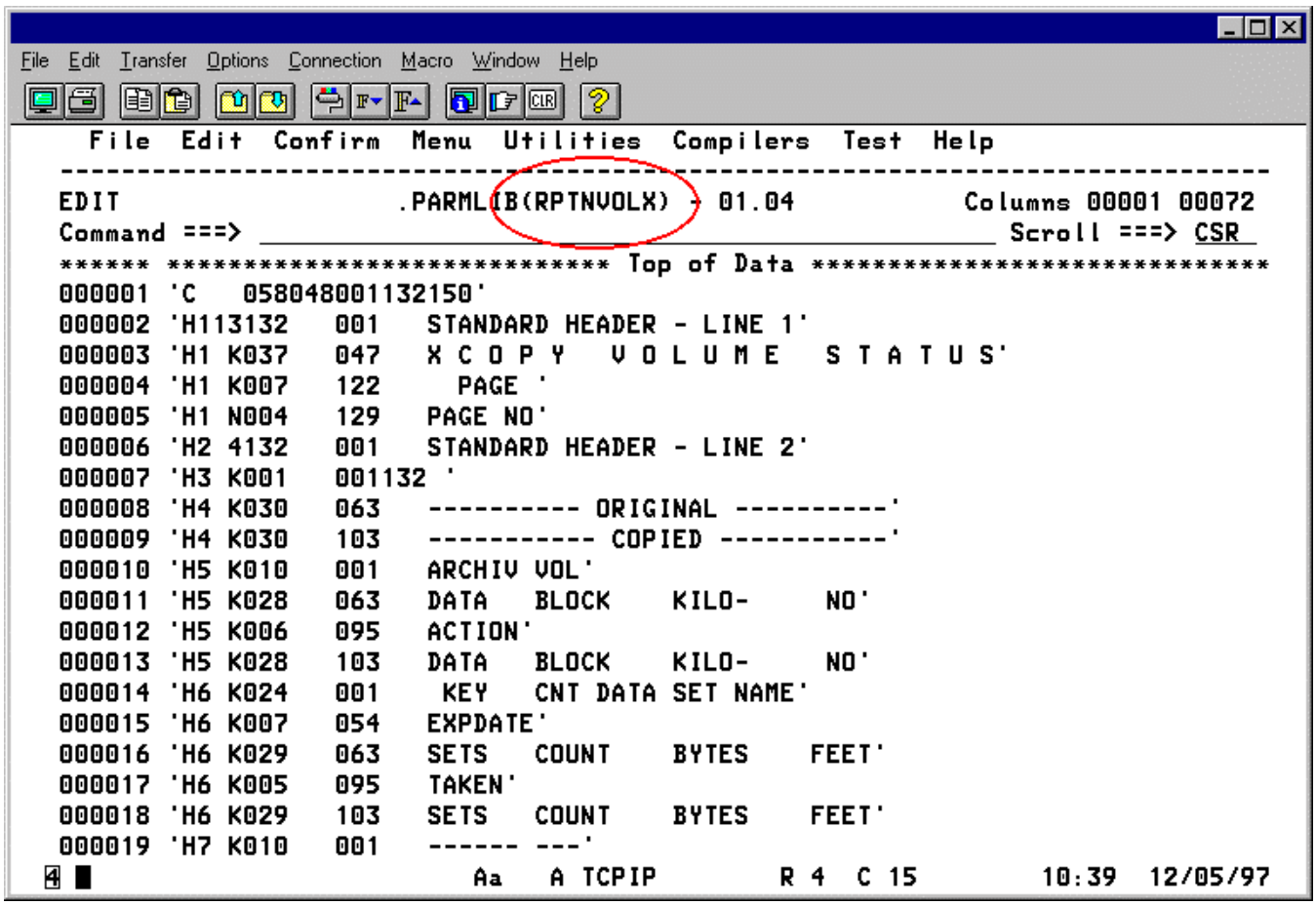
```
*
*      REPORT RECORD FOR COPY DSNINDEX PROCESSING
*
         DS    0F
RPTREC   DS    0CL136
RPTLEN   DS    H,H
RPTNAME  DS    CL12            RPTDSNXCP
RPTLRDSN DS    CL90            USE ONLY THE FIRST 90 BYTES
DISPOSIT DS    CL8
RPTARKEY DS    CL6
RPTBLKNO DS    F
RPTFILNO DS    H
RPTTUNIT DS    AL1
         DS    X
RPTSIM   DS    CL8
```

Report Record Layout — (RPTNVOLX)

The following figure displays the first page of the source table used for printing when the following command is executed:

```
XCOPY FLAG,CHKFLAG,TDSN=
```

FIGURE 43. Sample Parmlib Member RPTNVOLX



```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTNVOLX) - 01.04          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C   058048001132150'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K037   047   X C O P Y   V O L U M E   S T A T U S'
 000004 'H1 K007   122     PAGE '
 000005 'H1 N004   129   PAGE NO'
 000006 'H2 4132   001   STANDARD HEADER - LINE 2'
 000007 'H3 K001   001132 '
 000008 'H4 K030   063   ---------- ORIGINAL ----------'
 000009 'H4 K030   103   ----------- COPIED -----------'
 000010 'H5 K010   001   ARCHIV VOL'
 000011 'H5 K028   063   DATA   BLOCK    KILO-     NO'
 000012 'H5 K006   095   ACTION'
 000013 'H5 K028   103   DATA   BLOCK    KILO-     NO'
 000014 'H6 K024   001    KEY   CNT DATA SET NAME'
 000015 'H6 K007   054   EXPDATE'
 000016 'H6 K029   063   SETS   COUNT    BYTES    FEET'
 000017 'H6 K005   095   TAKEN'
 000018 'H6 K029   103   SETS   COUNT    BYTES    FEET'
 000019 'H7 K010   001   ------ ---'
 4 ■                       Aa   A TCPIP        R 4  C 15        10:39  12/05/97
```

Report Record Format

```
*
** XCOPY VOLUME REPORT RECORD
*
RPTREC   DS    0F
         DS    H,H
RPTNAME  DS    CL12            RPTNVOLXCP
RPTLRVOL DS    CL110
         DS    H
UNEXFEET DS    H
UNEXFILE DS    H
UNEXBLKS DS    F
UNEXBYTS DS    F
DISPOSIT DS    CL8
         DS    H
RPTSIM   DS    CL8
RPTLENTH EQU   *-RPTREC
```

Report Record Format — (RPTNDSINDX)

Data Set Index Report

```
DSINDX
*
*      Mapping of report record used to print DATA SET INDEX
*      Report
RPTBUILD DS    0F
         DC    H'86,0'
RPTTABLE DC    CL10'RPTNDSINDX'
RPTDSNAM DC    CL44            DATA SET NAME
RPTVOLNO DC    CL6             VOLUME SERIAL
RPTDEVCL DC    XL2             DEVICE CLASS
RPTDSORG DC    XL2             DSORG
RPTRECFM DC    XL1             RECFM
RPTBLKSI DC    XL2             BLOCKSIZE
RPTLRECL DC    XL2             LRECL
RPTSCALO DC    XL1             ALLOCATION
RPTSCQTY DC    XL3             SECONDARY QUANTITY
RPTTKSUM DC    XL3             TOTAL TRACKS ALLOCATED
RPTCREDT DC    XL3             CREATION DATE
RPTEXPDT DC    XL3             EXPIRATION DATE
```

Table Used for Printing Data Set Index Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT DSN=/
      REPORT DSINDX
```

FIGURE 44. Sample Parmlib Member RPTNDSIN

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                  .PARMLIB(RPTNDSIN) - 01.01          Columns 00001 00072
 Command ===> _                                            Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058053003045086'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129   PAGE NO'
 000005 'H1 K027   053   D A T A   S E T   I N D E X'
 000006 'H2 4132   001   STANDARD HEADER - LINE 2'
 000007 'H3 K001   001132'
 000008 'HC K013   001   DATA SET NAME'
 000009 'HC K006   036   VOLUME'
 000010 'H4 K045   001   ----------------------------------------    '
 000011 'H4 K045   046   ----------------------------------------    '
 000012 'H4 K045   091   ----------------------------------------    '
 000013 'D1 C044001001033   DSN'
 000014 'D1 C006045036      VOL'
 000015 'T1 K040   001   ----------------------------------------'
 000016 'T1 K040   041   ----------------------------------------'
 000017 'T1 K040   081   ----------------------------------------'
 000018 'T1 K012   121   ------------'
 000019 'T2 K015   015   TOTAL DATA SETS'
 4 ▌                       Aa   A TCPIP        R 4  C 15        10:57  12/05/97
```

Report Record Format — (RPTNDSUTIL)

Data Set Utilization Report

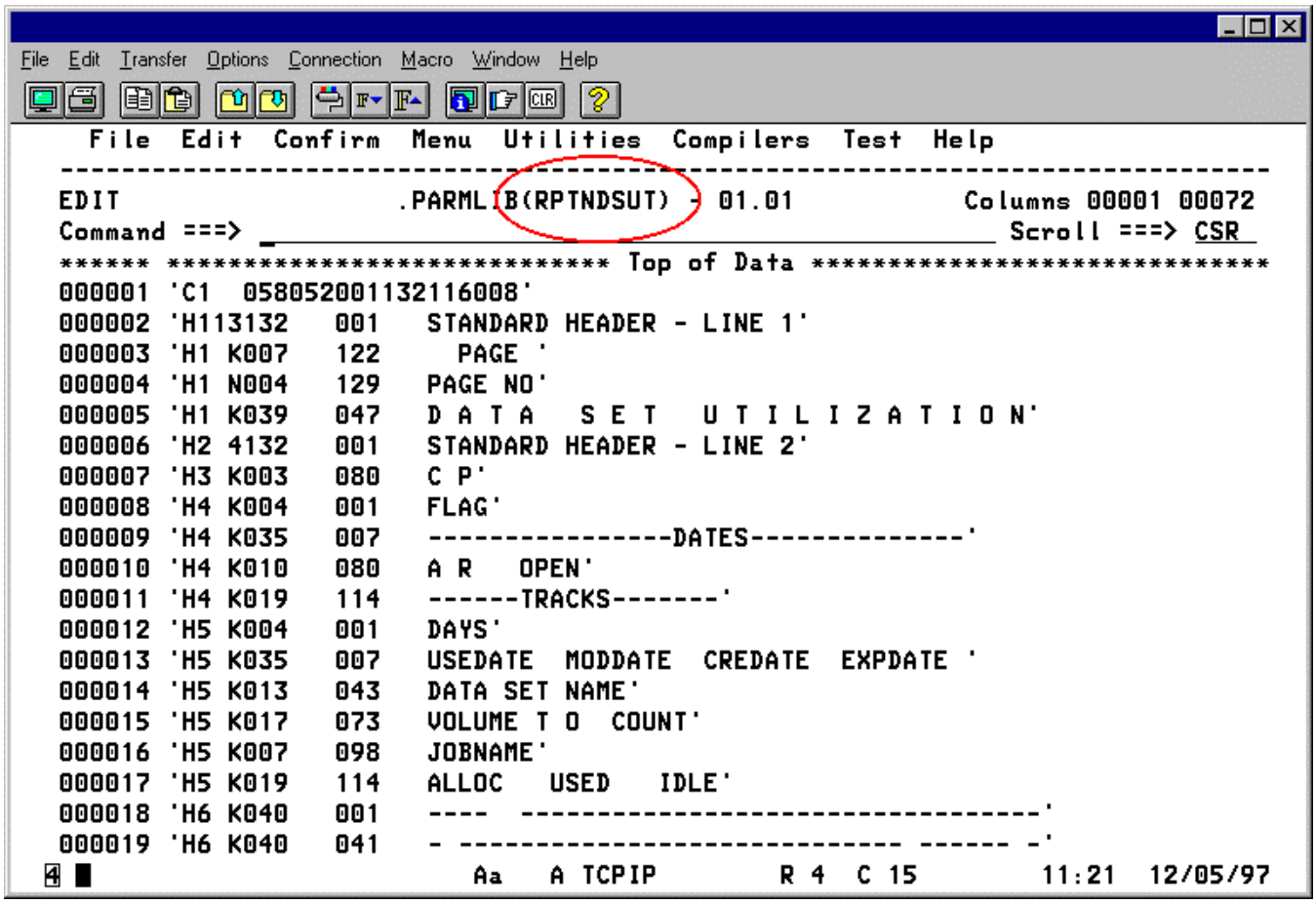
```
DSUTIL
*
*      Mapping of report record used to print DATA SET UTILIZATION
*      report
RPTBUILD DS    0F
         DC    H'127,0'
RPTTABLE DC    CL10'RPTNDSUTIL'
RPTDSNCF DC    CL8' '            1
RPTSQFLD DC    XL3'000000'       9
RPTDSN   DC    CL44' '           12
RPTVOL   DC    CL6' '            56
RPTCDATE DC    XL3'000000'       62
RPTEDATE DC    XL3'000000'       65
RPTOPENS DC    XL4'00000000'     68
RPTJOBNM DC    CL8' '            72
RPTTRKS  DC    XL2'0000'         80
RPTUSED  DC    XL2'0000'         82
RPTIDLE  DC    XL2'0000'         84
RPTLUPDT DC    XL3'000000'       86
RPTLUSED DC    XL3'000000'       89
RPTLUFLG DC    C' '              92
RPTCATLG DC    C' '              93
RPTSECTY DC    C' '              94
RPTOWNER DC    CL11' '           95
```

Table Used for Printing DSUTIL Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT DSN=/
      REPORT DSUTIL
```

FIGURE 45. Sample Parmlib Member RPTNDSUT



```
File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
--------------------------------------------------------------------------------
EDIT                    .PARMLIB(RPTNDSUT) - 01.01          Columns 00001 00072
Command ===> _                                               Scroll ===> CSR
****** ***************************** Top of Data ******************************
000001 'C1  058052001132116008'
000002 'H113132   001   STANDARD HEADER - LINE 1'
000003 'H1 K007   122     PAGE '
000004 'H1 N004   129   PAGE NO'
000005 'H1 K039   047   D A T A   S E T   U T I L I Z A T I O N'
000006 'H2 4132   001   STANDARD HEADER - LINE 2'
000007 'H3 K003   080   C P'
000008 'H4 K004   001   FLAG'
000009 'H4 K035   007   ----------------DATES--------------'
000010 'H4 K010   080   A R   OPEN'
000011 'H4 K019   114   ------TRACKS-------'
000012 'H5 K004   001   DAYS'
000013 'H5 K035   007   USEDATE  MODDATE  CREDATE  EXPDATE '
000014 'H5 K013   043   DATA SET NAME'
000015 'H5 K017   073   VOLUME T O  COUNT'
000016 'H5 K007   098   JOBNAME'
000017 'H5 K019   114   ALLOC   USED   IDLE'
000018 'H6 K040   001   ----  ----------------------------------'
000019 'H6 K040   041   - ---------------------------- ------ -'
4 ■                       Aa   A TCPIP        R 4  C 15        11:21  12/05/97
```

Report Record Format — RPTNDAYWEK

Utilization by Date and Time Report

```
SMFRPT
*
*      Mapping of Report record used to print UTILIZATION BY
*      DATE AND TIME REPORT
*
RECORD  DS    0CL76
DSNAME  DC    CL44' '            DATA SET NAME
VOLSER  DC    CL6' '             VOLUME SERIAL NUMBER
        DC    CL2                BLANKS
JOBTIME DC    XL4'00'            DURATION OF JOB
JOBDATE DS    0PL4               DATE (JULIAN) OF JOB RUN
        DC    PL1'0'
        DC    PL3'0'
SMFTIME DC    XL4'00'            SMF JOB TIME
SMFDATE DC    PL4                SMF (JULIAN) DATE OF JOB RUN
        DC    X'0'
START   DC    XL3'00'            START DATE
        DC    X'0'
STOP    DC    XL3'00'            STOP DATE
```

Table Used for Printing DAYWEK Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
//SMFDATA  EXEC SMFRPT
   SELECT COUNT=1000,TAPESCAN REALVOLS
```

FIGURE 46. Sample Parmlib Member RPTNDAYW

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTNDAYW) - 01.01          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058051'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K055   037   U T I L I Z A T I O N   B Y   D A T E   A N D   T',
 000004 ' I M E'
 000005 'H1 K007   122     PAGE '
 000006 'H1 N004   129   PAGE NO'
 000007 'H2 4132   001   STANDARD HEADER - LINE 2'
 000008 'H3 K001   001132'
 000009 'H4 K011   064   DAY OF WEEK'
 000010 'H4 K021   098   H O U R   O F   D A Y'
 000011 'H5 K027   105   1 1 1 1 1 1 1 1 1 1 2 2 2 2'
 000012 'H6 K013   001   DATA SET NAME'
 000013 'H6 K006   046   VOLUME'
 000014 'H6 K019   060   S  M  T  W  T  F  S'
 000015 'H6 K047   085   0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3'
 000016 'H7 K040   001   ---------------------------------------'
 000017 'H7 K040   041   ---- ------        -------------------- '
 000018 'H7 K040   081       ------------------------------------'
 000019 'H7 K012   121   ----------- '
 4 ■                       Aa   A TCPIP        R 4  C 15        11:29  12/05/97
```

Report Record Format — (RPTNFREESP)

Volume Freespace Report

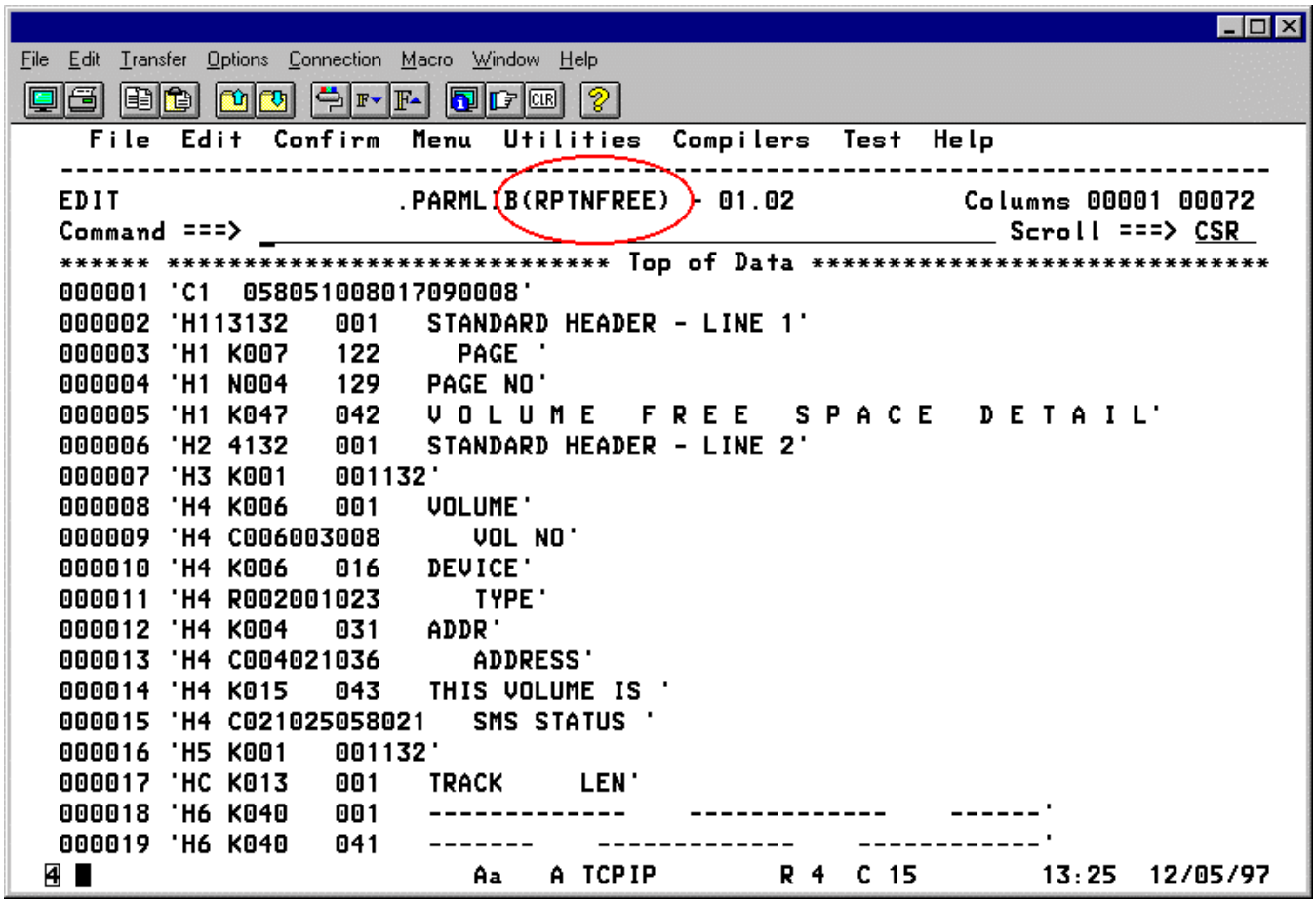
```
FREESP
*
*      Mapping of report record used to print FREESPACE report
*
DETREC   DS    0F
DETCONLN DS    H            RECORD LENGTH, 61 bytes
         DS    H            (FILLER ZEROS)
DETTABLE DS    CL10         REPORT NAME 'RPTNFREESP'
DETDEVCL DS    XL2          DEVICE CLASS
DETVOLNO DS    CL6          VOLSER
DETBEGTK DS    XL4          BEGINNING TRACK
DETENDTK DS    XL4          ENDING TRACK
DETLENTK DS    XL4          TRACK COUNT
DETVOLAD DS    CL4          VOLUME UNIT ADDRESS
DETSMS   DS    CL21         SMS VOLUME STATUS
DETEXTNO DS    XL2          EXTENT NUMBER
DETRECLN EQU   *            DETREC LENGTH OF RECORD, 61 BYTES
```

Table Used to Print Volume Freespace Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT VOLUMES=/
      REPORT FREESP
```

FIGURE 47. Sample Parmlib Member RPTNFREE



Report Record Format — (RPTNLAYOUT)

Volume Layout Report

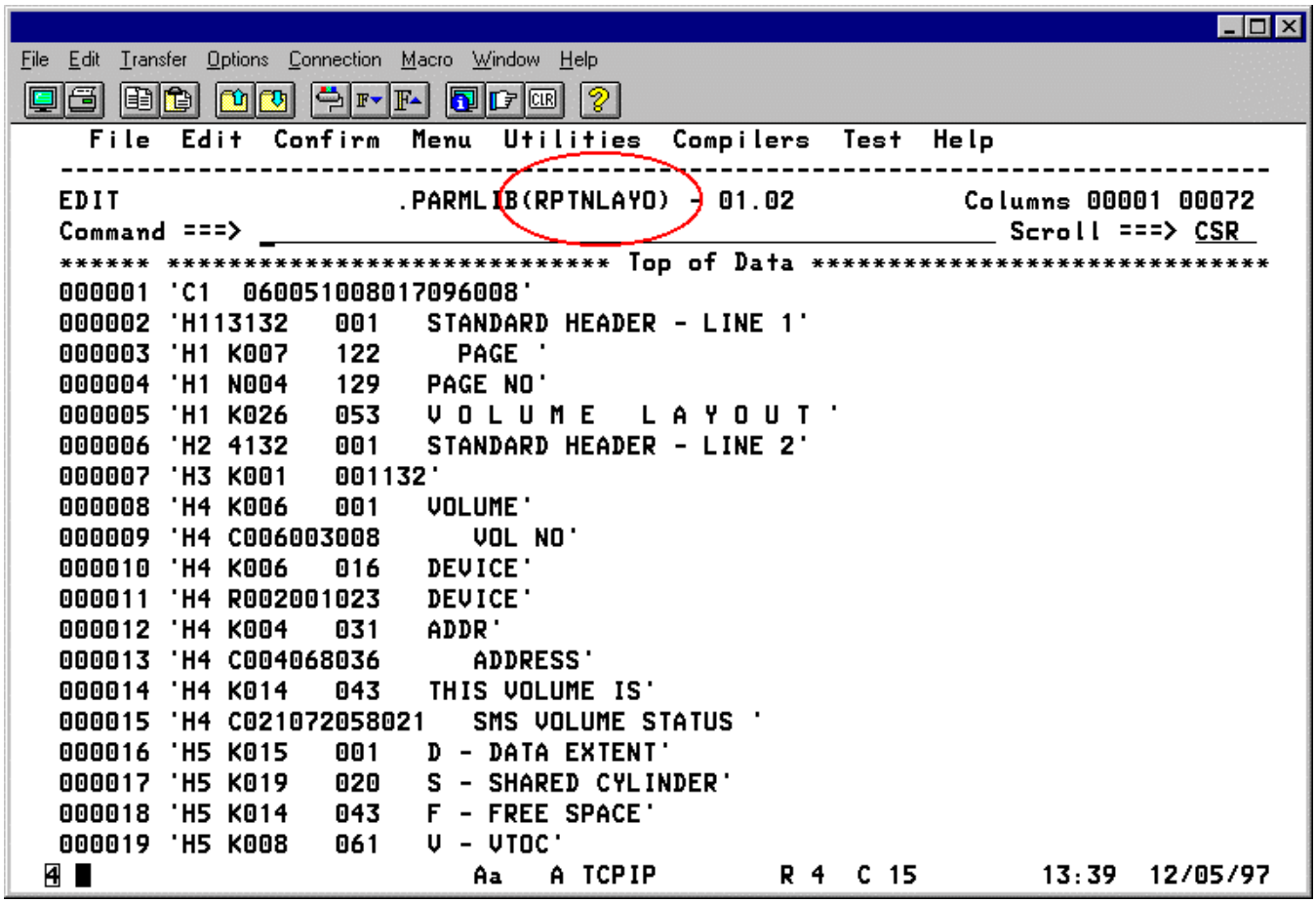
```
LAYOUT
*
*      Mapping of report record used to print VOLUME LAYOUT
*      report
*
DETREC   DS    0F
DETCONLN DS    H            RECORD LENGTH, 106 BYTES
         DS    H            (FILLER, ZEROS)
DETTABLE DS    CL10         REPORT NAME 'RPTNLAYOUT'
DETDEVCL DS    XL2          DEVICE CLASS
DETVOLNO DS    CL6          VOLUME SERIAL
DETBEGTK DS    XL4          BEGINNING TRACK
DETENDTK DS    XL4          ENDING TRACK
DETLENTK DS    XL4          TRACK LENGTH
DETEXTNO DS    HL2          EXTENT NUMBER
DETDSNAM DS    CL44         DATA SET NAME
DETEXTYP DS    CL1          EXTENT TYPE
DETVOLAD DS    CL4          VOLUME ADDRESS
DETSMS   DS    CL21         SMS VOLUME STATUS
DETRECLN EQU   *            DETREC LENGTH OF RECORD, 106 BYTES
*
```

Table Used for Volume Layout Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT VOLUMES=/
      REPORT LAYOUT
```

FIGURE 48. Sample Parmlib Member RPTNLAYO



Report Record Format — (RPTNMAPPER)

Volume Mapping Report

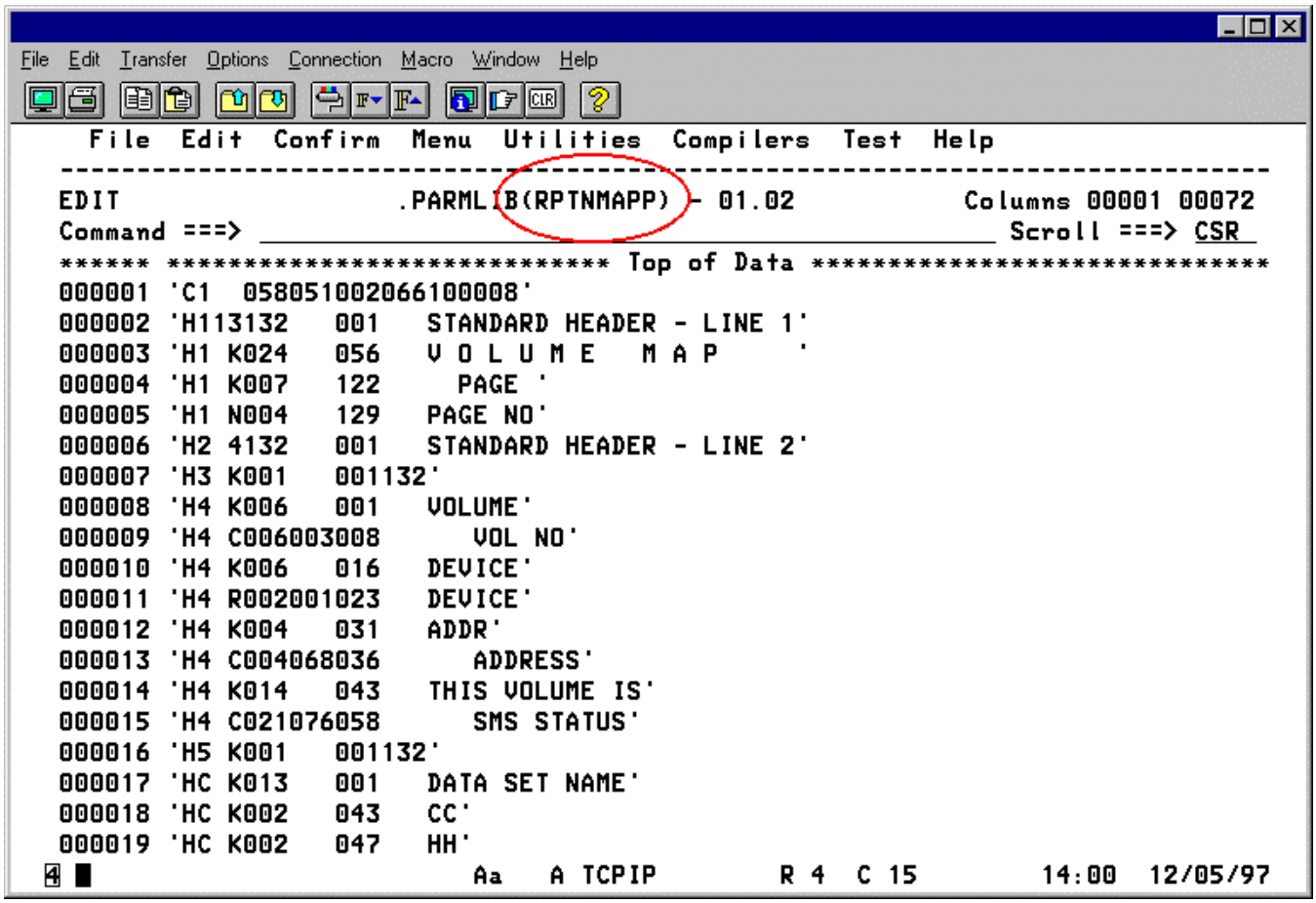
```
MAPPER
*
*      Mapping of report record used to print VOLUME MAPPING
*      report
*
DETREC   DS    0F
DETCONLN DS    H            RECORD LENGTH, 110 BYTES
         DS    H            (FILLER, ZEROS)
DETTABLE DS    CL10 REPORT NAME, 'RPTNMAPPER'
DETDEVCL DS    XL2          DEVICE CLASS
DETVOLNO DS    CL6          VOLUME SERIAL
DETBEGTK DS    XL4          BEGINNING TRACK
DETENDTK DS    XL4          ENDING TRACK
DETLENTK DS    XL4          TRACK COUNT
DETEXTNO DS    H            EXTENT NUMBER
DETDSNAM DS    CL44         DATA SET NAME
DETEXTYP DS    CL1          EXTENT TYPE
DETVOLAD DS    CL4          VOLUME ADDRESS
DETCCVAL DS    XL2          CC VALUE
DETHHVAL DS    XL2          HH VALUE
DETSMS   DS    CL21         SMS VOLUME STATUS
DETRECLN EQU   *            DETREC LENGTH OF RECORD, 110 BYTES
*
```

Table Used to Print Volume Mapping Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT VOLUMES=/
      REPORT MAPPER
```

FIGURE 49. Sample Parmlib Member RPTNMAPP



Report Record Format — (RPTNMEMBER)

PDS Directory / Member Report

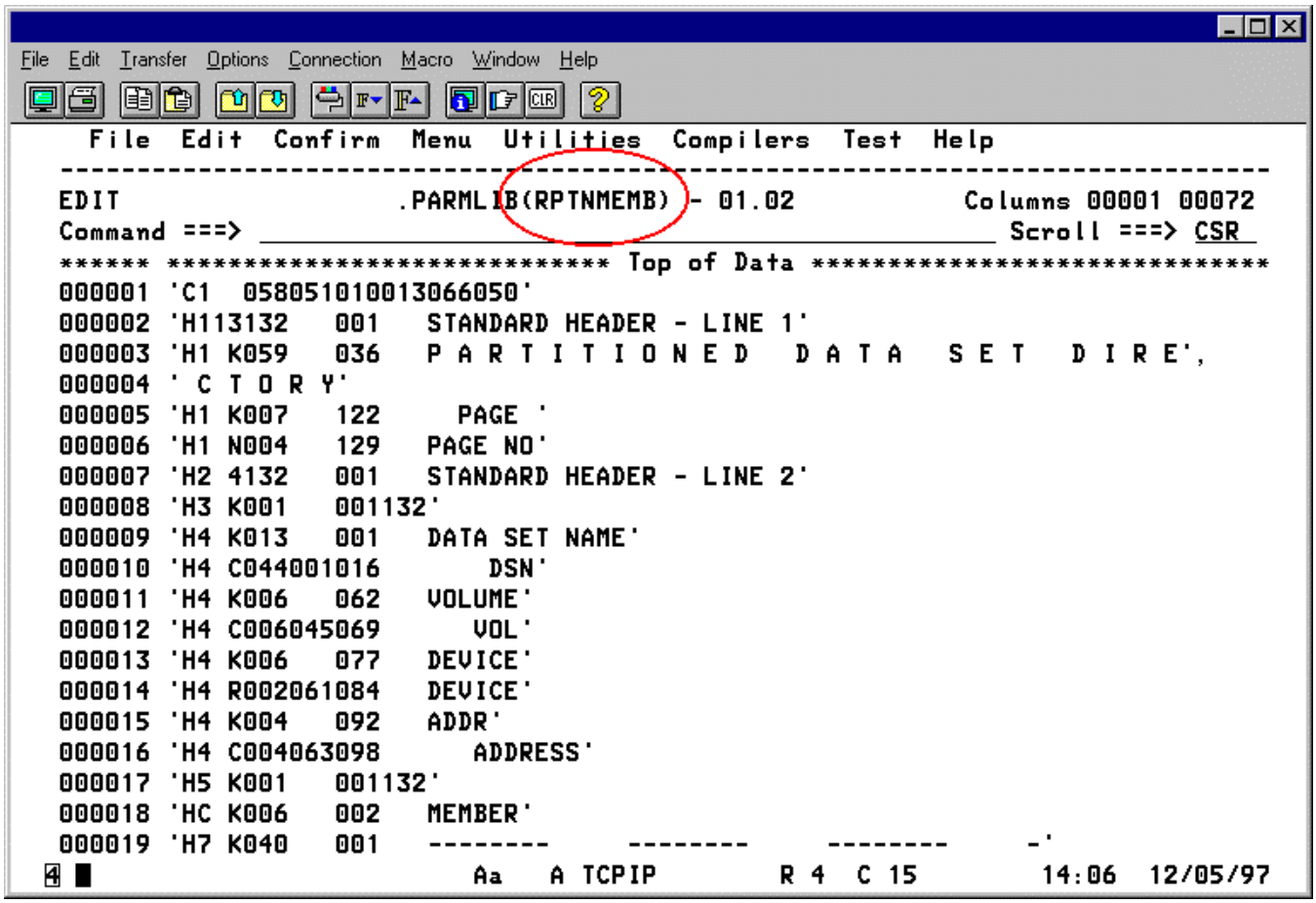
```
MEMBER
*
*      Mapping of report record used to print MEMBER report
*
MEMREC   DS    0F
         DC    H'74'
         DS    H
MEMTBLNM DC    CL10'RPTNMEMBER'
MEMDSN   DS    CL44            DATA SET NAME
MEMVOL   DS    CL6             VOLUME
MEMNAME  DS    CL8             MEMBER NAME
MEMALIAS DS    C               ALIAS
MEMRENT  DS    C               RE-ENTRANT
```

Table Used to Print MEMBER Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
FIND DSNAMES=
   REPORT MEMBER
```

FIGURE 50. Sample Parmlib Member RPTNMEMB



Report Record Format — (RPTNMVDICT)

Multiple Volume Data Set Dictionary Report

```
MCDICT
*
*       Mapping of report record used to print MULTIPLE VOLUME
*       DATA SET DICTIONARY report
*
*
RPTBUILD DS    0F
         DC    H'108,0'
RPTTABLE DC    CL10'RPTNMVDICT'
MVDDSNAM DS    CL44
MVDVOLNO DS    CL6
MVDDEVCL DS    XL2
RPTDSORG DS    XL2
RPTRECFM DS    XL1
RPTBLKSI DS    XL2
RPTLRECL DS    XL2
RPTSCALO DS    XL1
RPTSCQTY DS    XL3
RPTTKSUM DS    XL4
RPTCREDT DS    XL3
RPTEXPDT DS    XL3
RPTDSIND DS    XL1
RPTPRMAL DS    XL4
RPTIDLET DS    XL4
RPTVOLAD DS    CL4
RPTCATLG DS    CL1
RPTLUSDT DS    XL3
RPTNOEXT DS    XL1
```

Table Used to Print MVDICT Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
FIND DSNAMES=
   REPORT MVDICT
```

FIGURE 51. Sample Parmlib Member RPTNMVDI

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTNMVDI) - 01.02          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132090'
 000002 'H113132   001'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129'
 000005 'H1 K032   033   M U L T I P L E   V O L U M E   '
 000006 'H1 K038   065   D A T A   S E T   D I C T I O N A R Y'
 000007 'H2 4132   001'
 000008 'H3 K003   048   C P'
 000009 'H4 K015   036   DEVICE UNIT A R'
 000010 'H4 K031   074   ---ALLOCATION-- ----TRACKS-----'
 000011 'H5 K013   001   DATA SET NAME'
 000012 'H5 K006   029   VOLUME'
 000013 'H5 K011   036   TYPE   ADDR'
 000014 'H5 K007   048   T O DSO'
 000015 'H5 K026   056   RECFM BLKSZ LRECL TYP PRIM'
 000016 'H5 K016   084   SEC   ALLOC IDLE'
 000017 'H5 K030   102   EXT CREDATE  EXPDATE  USEDATE '
 000018 'H6 K040   001   --------------------------- ------ -----'
 000019 'H6 K040   041   - ---- --- --- ----- ----- ----- --- ---'
 4 ▌                       Aa   A TCPIP        R 4  C 15        14:12  12/05/97
```

Report Record Format — (RPTNPDSTAT)

Partitioned Data Set Status Report

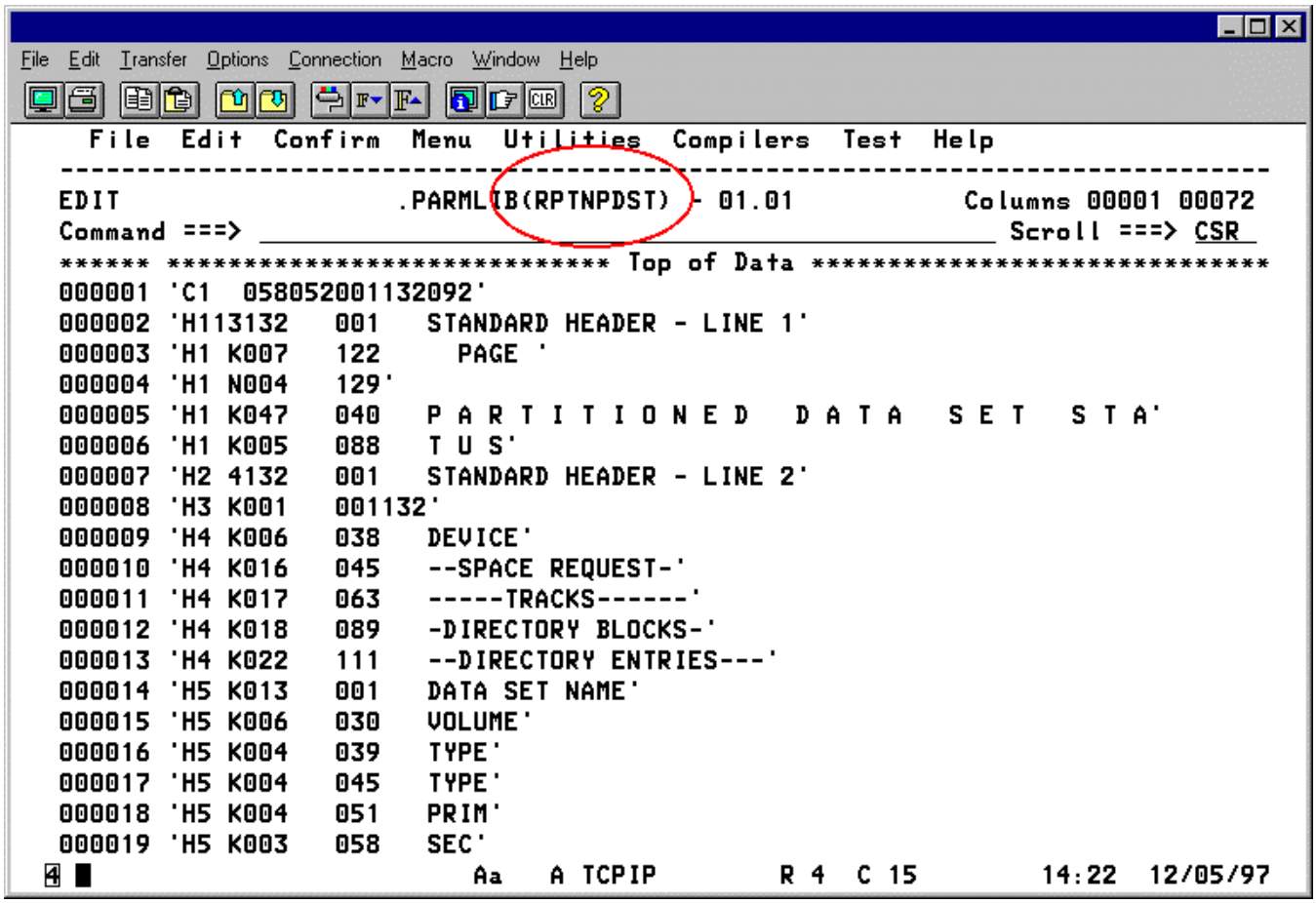
```
PDSTAT
*
*      Mapping of report record used to print PARTITIONED DATA
*      SET STATUS report
*
DETPDSTB DS    0F
         DC    H'92,0'            record length
DETTABLE DC    CL10'RPTNPDSTAT'
DETDSNAM DS    CL44               DATA SET NAME
DETDEVIC DS    XL2                DEVICE CLASS
DETVOLSR DS    CL6                VOLSER
DETEXTNT DS    XL1                NUMBER OF EXTENTS
DETTRKAL DS    XL2                TRACKS REQUESTED WHEN CREATED
DETTRKUS DS    XL2                TRACKS USED
DETTRKID DS    XL2                TRACKS IDLE
DETDIRAL DS    XL2                DIRECTORY BLOCKS ALLOCATED
DETDIRUS DS    XL2                DIRECTORY BLOCKS USED
DETDIREM DS    XL2                DIRECTORY BLOCKS EMPTY
DETMBRCU DS    XL2                MEMBER COUNT
DETMBRPO DS    XL2                MEMBER LIMIT
DETMBRGR DS    XL2                AVAILABLE ENTRIES
DETMBLEN DS    XL2                ENTRY LENGTH
DETALPRI DS    XL2                TRACKS ALLOCATED
DETALSEC DS    XL2                SECONDARY ALLOCATION
DETALTYP DS    XL1                ALLOCATION TYPE
```

Table Used to Print PDS Status Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
FIND DSNAMES=
   REPORT PDSTAT
```

FIGURE 52. Sample Parmlib Member RPTNPDST



Report Record Format — (RPTNPOCCHH, RPTNPODUMP, and RPTNPOTTRD)

Partitioned Data Set Directory (CCHHR, DUMP, TTRD)

```
*
*       Mapping of report record used to print the POCCHH,
*       PODUMP, AND POTTRD reports.
*
RECORD   DS    0F
RECLEN   DS    H            Length of record (86 to 148)
         DS    H
RPTPGM   DS    CL10         Name of module to format the rpt
DSNAME   DS    CL44         Dataset name
VOLUME   DS    CL6          Volume
KEY      DS    CL8          Key
CCHHR    DS    XL5          CCHHR of member
ALIAS    DS    C            Member name is an alias
DEVTYPE  DS    XL2          Devtype
VOLAD    DS    CL4          Volume address
#ENTLEN  DS    H            Length of user data
#DIRENT  DS    XL62         User data
```

Report Record Format — (RPTNPOXREF, RPTNPO1REF, and RPTNPO2REF)

Data Set Name Cross-Reference for PDS Member Names

```
*
*       Mapping of report record used to print the POXREF,
*       PO1REF, and PO2REF reports.
*
POXREC   DS    0F
POXLEN   DS    H            Length of record
         DS    H
POXTBLN  DS    CL10         Name of module to format the rpt
POXMBR   DS    CL8          Member name
POXDSN   DS    CL44         Dataset name
POXVOL   DS    CL6          Volume serial
POXALIAS DS    C            Member name is an alias
POXDEV   DS    XL2          Device type
POXVADDR DS    CL4          Volume address
```

Report Record Format — (RPTNRACFDN)

RACF Profile X-REF in Dsname Sequence

```
*       Mapping of report record used to print RACF PROFILE TO
*       ARCHIVE DSNAME X-REF, in archive dsname/date/time sequence
*
RDNRECD  DC    H'106,0'
         DC    CL10'RPTNRACFDN'
RDNDSN   DC    CL44' '            Dataset name
RDNDATE  DC    XL3'0'             Archive date
RDNTIME  DC    CL4' '             Archive time
RDNPROF  DC    CL41' '            Profile name
```

Table Used to Print X-REF Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
LISTDSD PREFIX() ALL
   SELECT DMSONLY,DSNAME=,PREFIX=
```

FIGURE 53. Sample Parmlib Member RPTNRFDN

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

   File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTNRFDN) - 01.01          Columns 00001 00072
 Command ===> _                                              Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132104'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129   PAGE NO'
 000005 'H1 K045   030   R A C F   P R O F I L E   T O   A R C H I V E'
 000006 'H1 K026   075     D S N A M E   X - R E F'
 000007 'H2 4132   001   STANDARD HEADER - LINE 2'
 000008 'H2 K033   049   ARCHIVE DSNAME/DATE/TIME SEQUENCE'
 000009 'H3 K001   001132'
 000010 'H4 K007   001   ARCHIVE'
 000011 'H4 K007   049   ARCHIVE'
 000012 'H4 K007   062   PROFILE'
 000013 'H5 K013   001   DATA SET NAME'
 000014 'H5 K010   047   TIME DATE '
 000015 'H5 K013   062   DATA SET NAME'
 000016 'H6 K040   001   ----------------------------------------'
 000017 'H6 K040   041   ----  -------------  -------------------'
 000018 'H6 K022   081   ----------------------'
 000019 'D1 C044001001     DSN'
 4 ■                       Aa   A TCPIP        R 4  C 15        14:36  12/05/97
```

Report Record Format — (RPTNRACFPF)

RACF Profile X-REF in Profile Name Sequence

```
*       Mapping of report record used to print RACF PROFILE TO
*       ARCHIVE DSNAME X-REF, in profile name sequence
*
RPFRECD  DC    H'106,0'
         DC    CL10'RPTNRACFPF'
RPFPROF  DC    CL41' '            Profile name
RPFDSN   DC    CL44' '            Dataset name
RPFDATE  DC    XL3'0'             Archive date
RPFTIME  DC    CL4' '             Archive time
```

Table Used to Print X-REF Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
LISTDSD ALL
   SELECT DSNAME=,PROFILE=
```

FIGURE 54. Sample Parmlib Member RPTNRFPF

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTNRFPF) - 01.01          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132104'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129   PAGE NO'
 000005 'H1 K045   030   R A C F   P R O F I L E   T O   A R C H I V E'
 000006 'H1 K026   075     D S N A M E   X - R E F'
 000007 'H2 4132   001   STANDARD HEADER - LINE 2'
 000008 'H2 K021   055   PROFILE NAME SEQUENCE'
 000009 'H3 K001   001132'
 000010 'H4 K007   001   PROFILE'
 000011 'H4 K007   047   ARCHIVE'
 000012 'H4 K007   097   ARCHIVE'
 000013 'H5 K013   001   DATA SET NAME'
 000014 'H5 K013   047   DATA SET NAME'
 000015 'H5 K010   095   TIME  DATE'
 000016 'H6 K040   001   ---------------------------------------'
 000017 'H6 K040   041   -     ----------------------------------'
 000018 'H6 K027   081   ----------    -------------'
 000019 'D1 C041001001     PROFILE NAME'
 4 ■                      Aa   A TCPIP        R 4  C 15        15:05  12/05/97
```

Report Record Format — (RPTNSVDICT)

Single Volume Dictionary Report

```
SVDICT
*
*      Mapping of report record used to print SINGLE VOLUME
*      DICTIONARY report
*      Note: The RPTDSORG field has:
*         1. X'02' for PDS, X'FD' for HFS, and X'FE' for PDSE
*         2. X'40' for PS and X'FC' for PSE
*
RPTBUILD DS    0F
         DC    H'108,0'
RPTTABLE DC    CL10'RPTNSVDICT'
SVDDEVCL DS    XL2
SVDVOLNO DS    CL6
SVDDSNAM DS    CL44
RPTDSORG DS    XL2
RPTRECFM DS    XL1
RPTBLKSI DS    XL2
RPTLRECL DS    XL2
RPTSCALO DS    XL1
RPTSCQTY DS    XL3
RPTTKSUM DS    XL4
RPTCREDT DS    XL3
RPTEXPDT DS    XL3
RPTDSIND DS    XL1
RPTPRMAL DS    XL4
RPTIDLET DS    XL4
RPTVOLAD DS    CL4
RPTCATLG DS    CL1
RPTLUSDT DS    XL3
RPTNOEXT DS    XL1
```

Table Used to Print SVDICT Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
SCAN REALVOLS
   SELECT VOLUMES=
      REPORT SVDICT
```

FIGURE 55. Sample Parmlib Member RPTNSVDI

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTNSVDI) - 01.02          Columns 00001 00072
 Command ===> _                                              Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132096008'
 000002 'H113132   001'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129'
 000005 'H1 K028   034   S I N G L E   V O L U M E   '
 000006 'H1 K038   062   D A T A   S E T   D I C T I O N A R Y'
 000007 'H2 4132   001'
 000008 'H3 K003   048   C P'
 000009 'H4 K015   036   DEVICE UNIT A R'
 000010 'H4 K031   074   ---ALLOCATION-- ----TRACKS-----'
 000011 'H5 K013   001   DATA SET NAME'
 000012 'H5 K006   029   VOLUME'
 000013 'H5 K011   036   TYPE   ADDR'
 000014 'H5 K007   048   T O DSO'
 000015 'H5 K026   056   RECFM BLKSZ LRECL TYP PRIM'
 000016 'H5 K016   084   SEC   ALLOC IDLE'
 000017 'H5 K030   102   EXT CREDATE  EXPDATE  USEDATE'
 000018 'H6 K040   001   --------------------------- ------ -----'
 000019 'H6 K040   041   - ---- --- --- ----- ----- ----- -------'
 4 ▌                       Aa   A TCPIP        R 4  C 15        15:11  12/05/97
```

Report Record Format — (RPTNVOLDSB)

Billing Volume Report

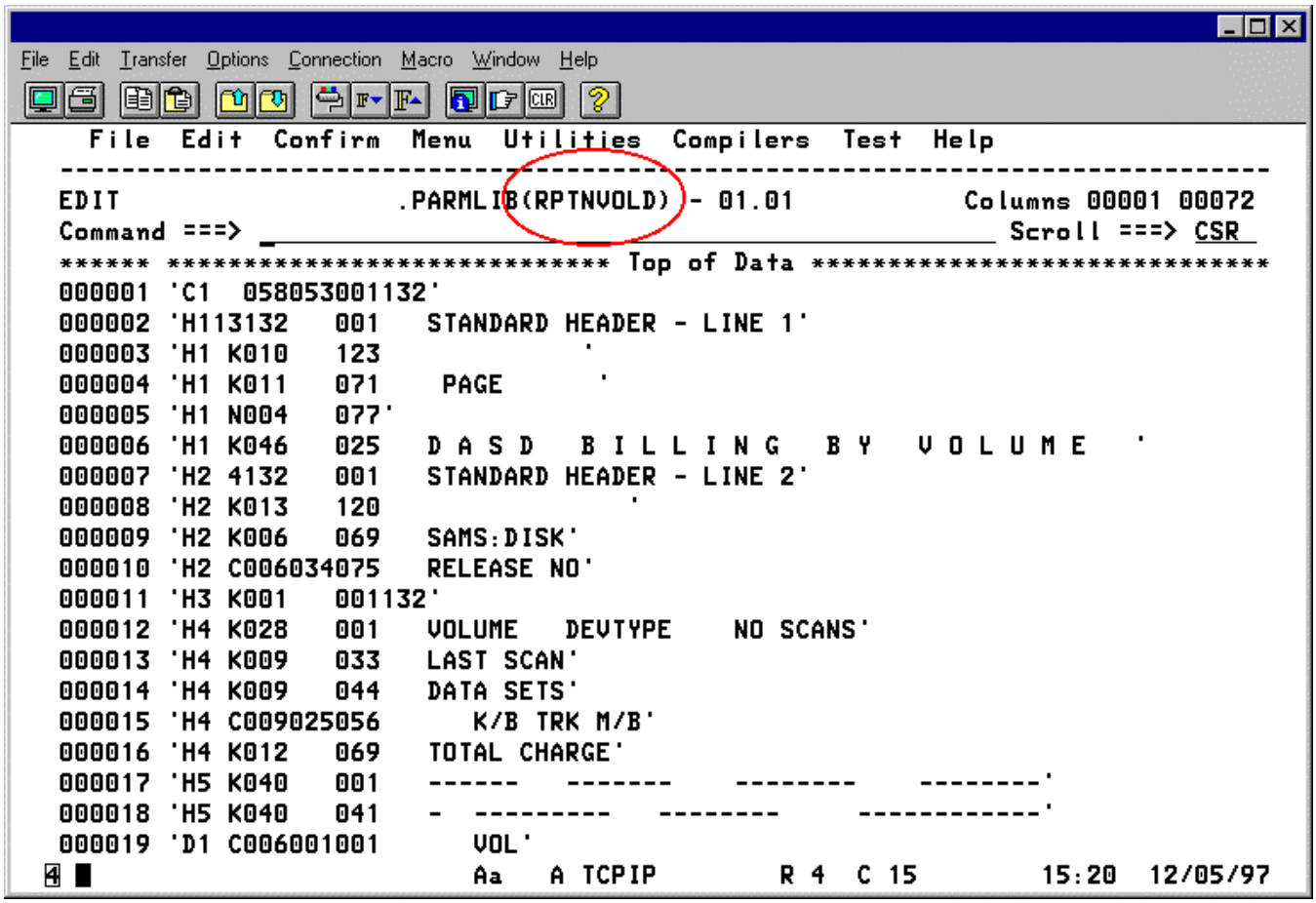
```
VOLDSB
*
*       Mapping of report record used to print BILLING VOLUME
*       report
*
VOLREC   DS    0F
         DC    H'76,0',CL10'RPTNVOLDSB'
VOLENTRY DS    0CL24
VOLSERNO DS    CL6            VOLSER
VOLDSCNT DS    H              DATA SET COUNT
VOLACCUM DS    F              ACCUMULATED AMOUNT FOR VOLUME
VOLTOTAL DS    F              VOLUME TOTAL - $
VOLDEVCL DS    XL2            DEVICE CLASS
VOLNOSCN DS    H              NUMBER OF SCANS
VOLSCNDT DS    XL3            SCAN DATE - YY DDDD
         DS    XL1            RESERVED
VOLUNITS DC    CL9'K/B DAYS ' or CL9'TRK DAYS '
VOLRELSE DC    CL6' '         RELEASE 9.9 = C' 9.9'
         DC    CL23' '        FILLER FOR SORT MINIMUM LENGTH
```

Table Used for Printing Billing Volume Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
//TOTALS  EXEC EXTEND
//SYSIN    DD  *
   EXTEND VOLUME=
```

FIGURE 56. Sample Parmlib Member RPTNVOLD



Report Record Format — (ARCHDSNVOL)

Archive/Backup/Delete/Expire Report

```
ARCHDSNVOL
*
*      Mapping of report record used to print ARCHIVE/RETAIN
*      report in DSN/VOL order
*
ARCHRECN DC    H'215,0'
         DC    CL10'ARCHDSNVOL'
NETAILIN DC    CL176' '
NOLDSNAM DC    CL44' '            OLD DATA SET NAME
NNEWNAME DC    CL44' ',CL3' '     NEW DATA SET NAME
NVOLSER  DC    CL6' '             VOLUME SERIAL
NASTRICH DC    C' ' *             ASTERISK
NDEVTYPE DC    CL6' '             DEVICE TYPE
NDSORG   DC    CL3' '             DSORG
NRECFM   DC    CL2' '             RECFM
NBLKSIZE DC    CL6' '             BLKSIZE
NLRECL   DC    CL5' '             LRECL
NOPTCD   DC    CL2' '             OPTCD
NKEYLEN  DC    CL3' '             KEY LEN
NACT     DC    C' '               ACTION TAKEN
NSCRATCH DC    C' '               SCRATCHED
NCATALOG DC    C' '               CATALOGED
NCRIT1   DC    C' '               REASON
NCRIT2   DC    C' '               REASON
NCRIT3   DC    C' '               REASON
NCRIT4   DC    C' '               REASON
NKILOBYT DC    CL6' '             KILOBYTES
NTRACKS  DC    CL6' '             TRACKS
NSIMULAT DC    CL8' '             SIMULATE
NTOTARCH DC    XL2'0'             DATA SET ARCHIVED
NTOTSCR  DC    XL2'0'             DATA SET SCRATCHED
NTRKARCH DC    XL2'0'             TRACKS ARCHIVED
NTRKSCR  DC    XL2'0'             TRACKS SCRATCHED
NBYTARCH DC    XL4'0'             KILOBYTES ARCHIVED
NBYTSCR  DC    XL4'0'             KILOBYTES SCRATCHED
NTITLE   DC    C' '               REPORT TITLE
```

Table Used to Print Archive Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
//DSCL      DMS
//SYSIN      DD   *
SCAN REALVOLS
   SELECT CRITERIA=(USEDT,LT,TODAY-180)
      ARCHIVE
//SYSPARMS   DD   *
ARCHSORTNDN
```

FIGURE 57. Sample Parmlib Member ARCHDSNV

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                  .PARMLIB(ARCHDSNV) - 01.01          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058051001132249'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 5001201032   TITLE'
 000004 'H1 K007   122     PAGE '
 000005 'H1 N004   129   PAGE NO'
 000006 'H2 4132   001   STANDARD HEADER - LINE 2'
 000007 'H2 K029   052   DATA SET NAME/VOLUME SEQUENCE'
 000008 'H3 K001   001132 '
 000009 'H6 K023   003   DATA SET NAME / NEWNAME'
 000010 'H6 K006   048   VOLUME'
 000011 'H6 K033   055   DEVTYPE DSORG RECFM BLKSZ  LRECL '
 000012 'H6 K045   088   OPTCD KEYL ARC SCR CAT REASON   KBYTES TRACKS'
 000013 'H7 K040   001     ------------------------------------'
 000014 'H7 K040   041   ------ ------ ------- ----- ----- ----- '
 000015 'H7 K040   081   ------ ----- ---- --- --- --- ------ ---'
 000016 'H7 K012   121   ----- ------'
 000017 'H807008177002     SIMULATE'
 000018 'D1 C001098001      ASTERISK'
 000019 'D1 C088001003044   DSN'
 4 ▌                        Aa   A TCPIP        R 4  C 15        15:30  12/05/97
```

Report Record Format — (ARCHVOLDSN)

Archive/Backup/Delete/Expire Report

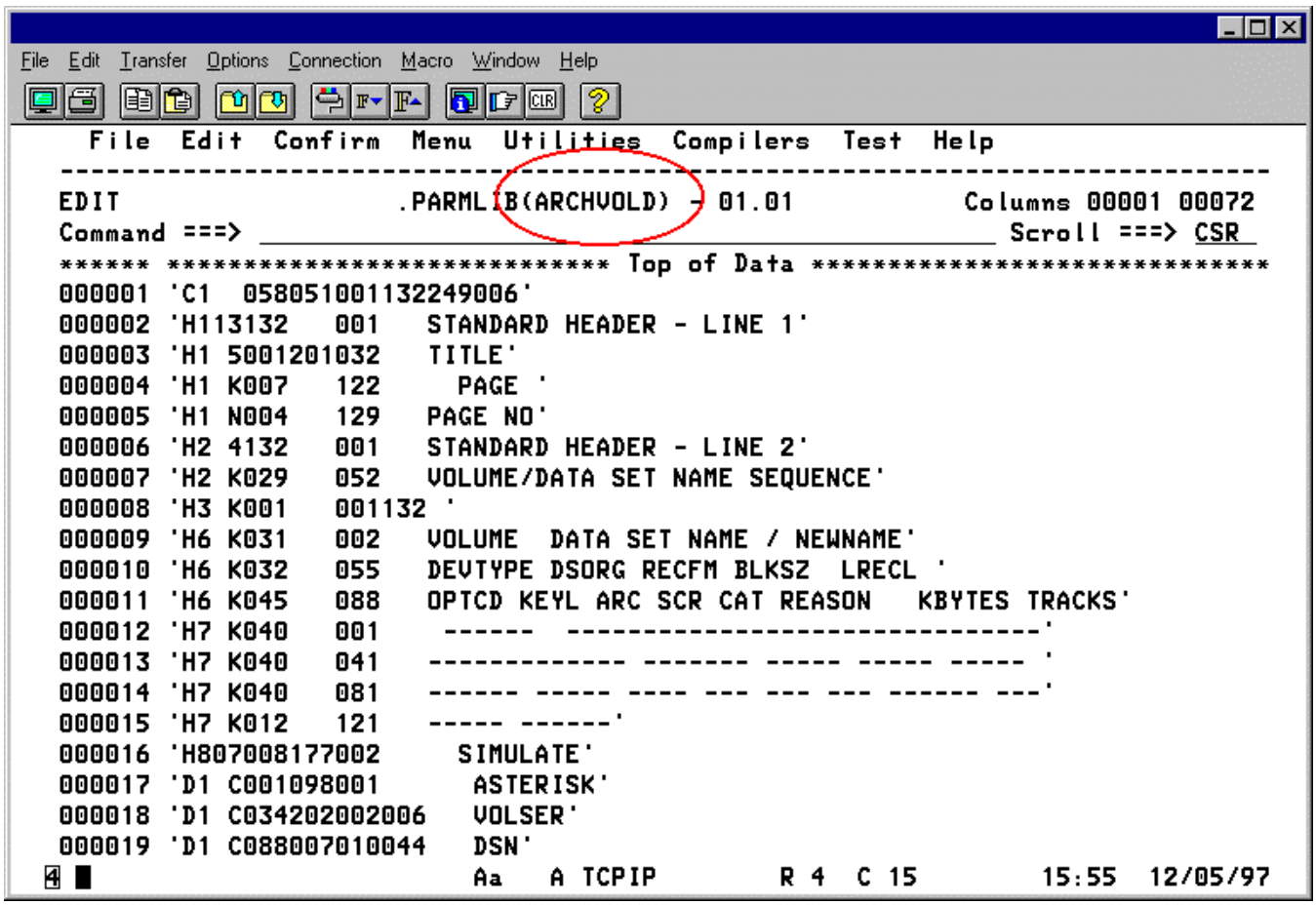
```
ARCHVOLDSN
*
*      Mapping of report record used to print ARCHIVE/RETAIN report
*      in VOL/DSN order
ARCHRECV DC    H'215,0'
RPTNAME  DC    CL10'ARCHVOLDSN'
VETAILIN DC    CL176' '
VVOLSER  DC    CL6' '            VOLUME SERIAL
VOLDSNAM DC    CL44' '           OLD DATA SET NAME
VNEWNAME DC    CL44' ',CL3' '    NEW DATA SET NAME
VASTRICH DC    C' '              ASTERISK *
VDEVTYPE DC    CL6' '            DEVICE TYPE
VDSORG   DC    CL3' '            DSORG
VRECFM   DC    CL2' '            RECFM
VBLKSIZE DC    CL6' '            BLKSIZE
VLRECL   DC    CL5' '            LRECL
VOPTCD   DC    CL2' '            OPTCD
VKEYLEN  DC    CL3' '            KEY LENGTH
VACT     DC    C' '              ACTION TAKEN
VSCRATCH DC    C' '              SCRATCHED
VCATALOG DC    C' '              CATALOGED
VCRIT1   DC    C' '              REASON
VCRIT2   DC    C' '              REASON
VCRIT3   DC    C' '              REASON
VCRIT4   DC    C' '              REASON
VKILOBYT DC    CL6' '            KILOBYTES
VTRACKS  DC    CL6' '            TRACKS
VSIMULAT DC    CL8' '            SIMULATE
VTOTARCH DC    XL2'0'            DATA SET ARCHIVED
VTOTSCR  DC    XL2'0'            DATA SET SCRATCHED
VTRKARCH DC    XL2'0'            TRACKS ARCHIVED
VTRKSCR  DC    XL2'0'            TRACKS SCRATCHED
VBYTARCH DC    XL4'0'            KILOBYTES ARCHIVED
VBYTSCR  DC    XL4'0'            KILOBYTES SCRATCHED
VTITLE   DC    C' '              REPORT TITLE
```

Table Used to Print Archive / Retain Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
//DSCL      DMS
//SYSIN      DD   *
SCAN REALVOLS
   SELECT CRITERIA=(USEDT,LT,TODAY-180)
      ARCHIVE
//SYSPARMS   DD   *
ARCHSORTNDV
```

FIGURE 58. Sample Parmlib Member ARCHVOLD



Report Record Format — (MIGRDSNVOL)

Migrate Report

```
MIGRDSNVOL
*
*      Mapping of report record used to print MIGRATE REPORT IN
*      DSN/VOL ORDER
MIGRRECN DC    H'178',H'0'
         DC    CL10'MIGRDSNVOL'
NETAILIN DS    0CL115
NOLDSNAM DC    CL44' '
NVOLSER  DC    CL34' '
NLASTHAF DS    0CL65
NDEVTYPE DC    CL6' ',C' '
NEXPDT   DC    CL8' ',CL1' '
NRECFM   DC    CL2' ',C' '
NBLKSIZE DC    CL6' ',C' '
NLRECL   DC    CL5' ',CL2' '
NOPTCD   DC    CL2' ',CL3' '
NKEYLEN  DC    CL3' ',CL2' '
NTAPEVOL DC    CL6' ',C' '
NSEQ     DC    CL3' ',CL2' '
NSCRATCH DC    C' ',CL2' '
NCATALOG DC    CL3' ',C' '
NREASON  DC    CL4' '
NKILOBYT DC    CL6' ',CL2' '
NTRACKS  DC    CL6' '
NSIMULAT DC    CL8' '
NTRKMIGR DC    XL2'0'
NBYTMIGR DC    XL4'0'
NTITLE   DC    C'6'
```

Table Used for Printing Migrate Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
//SEQMIGR EXEC MIGRATE
//SYSIN    DD  *
   MIGRATE DSNAME=,VOLUME=
//SYSPARMS DD  *
MIGRSORTNDN
```

FIGURE 59. Sample Parmlib Member MIGRDSNV

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                  .PARMLIB(MIGRDSNV) - 01.02          Columns 00001 00072
 Command ===> _                                            Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058051001132201'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 6001130032070TITLE'
 000004 'H1 K007   122     PAGE '
 000005 'H1 N004   129   PAGE NO'
 000006 'H2 4132   001   STANDARD HEADER - LINE 2'
 000007 'H2 K029   052   DATA SET NAME/VOLUME SEQUENCE'
 000008 'H3 K001   001132 '
 000009 'H4 K018   054   DEVICE         REC'
 000010 'H4 K004   097   TAPE'
 000011 'H5 K013   001   DATA SET NAME'
 000012 'H5 K006   046   VOLUME'
 000013 'H5 K036   054    TYPE  EXPDATE  FM BLKSZ LRECL OPTCD'
 000014 'H5 K043   090    KEYL VOLUME SEQ SCR CAT REAS KBYTE  TRACKS'
 000015 'H6 K040   001   ----------------------------------------'
 000016 'H6 K040   041   ---- ------  ------ ------- --- ----- --'
 000017 'H6 K040   081   --- ----- ---- ------ --- --- --- ---- -'
 000018 'H6 K012   121   ----  ------'
 000019 'H707008116001     SIMULATE LINE'
 4 ■                       Aa   A TCPIP        R 4  C 15        16:14  12/05/97
```

Report Record Format — (MIGRVOLDSN)

Migrate Report

```
MIGRVOLDSN
*
*      Mapping of report record used to print MIGRATE report in
*      VOL/DSN order
MIGRRECV DC    H'178',H'0'
         DC    CL10'MIGRVOLDSN'
VETAILIN DS    0CL115
VVOLSER  DC    CL34' '
VOLDSNAM DC    CL44' '
VLASTHAF DS    0CL65
VDEVTYPE DC    CL6' ',C' '
VEXPDT   DC    CL5' ',CL3' '
VRECFM   DC    CL2' ',C' '
VBLKSIZE DC    CL6' ',C' '
VLRECL   DC    CL5' ',CL2' '
VOPTCD   DC    CL2' ',CL3' '
VKEYLEN  DC    CL3' ',CL2' '
VTAPEVOL DC    CL6' ',C' '
VSEQ     DC    CL3' ',CL2' '
VSCRATCH DC    C' ',CL2' '
VCATALOG DC    CL3' ',C' '
VREASON  DC    CL4' '
VKILOBYT DC    CL6' ',CL2' '
VTRACKS  DC    CL6' '
VSIMULAT DC    CL8' '
VTRKMIGR DC    XL2'0'
VBYTMIGR DC    XL4'0'
VTITLE   DC    C'6'
```

Table Used for Printing Migrate Report in VOL/DSN Order

The following figure displays the first page of the source table used for printing when the following command is executed:

```
//SEQMIGR EXEC MIGRATE
//SYSIN    DD  *
   MIGRATE DSNAME=,VOLUME=
//SYSPARMS DD  *
MIGRSORTNNV
```

FIGURE 60. Sample Parmlib Member MIGRVOLD

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                  .PARMLIB(MIGRVOLD) - 01.02          Columns 00001 00072
 Command ===> _                                             Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058051001132201'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 6001130032070TITLE'
 000004 'H1 K007   122     PAGE '
 000005 'H1 N004   129   PAGE NO'
 000006 'H2 4132   001   STANDARD HEADER - LINE 2'
 000007 'H2 K029   052   VOLUME/DATA SET NAME SEQUENCE'
 000008 'H3 K001   001132 '
 000009 'H4 K018   054   DEVICE         REC'
 000010 'H4 K004   097   TAPE'
 000011 'H5 K006   001   VOLUME'
 000012 'H5 K013   008   DATA SET NAME'
 000013 'H5 K036   054    TYPE  EXPDATE  FM BLKSZ LRECL OPTCD'
 000014 'H5 K043   090    KEYL VOLUME SEQ SCR CAT REAS KBYTE  TRACKS'
 000015 'H6 K040   001   ------ ---------------------------------'
 000016 'H6 K040   041   -----------  ------ ------- --- ----- --'
 000017 'H6 K040   081   --- ----- ---- ------ --- --- --- ---- -'
 000018 'H6 K012   121   ----  ------'
 000019 'H707008116001     SIMULATE LINE'
 4 ■                       Aa   A TCPIP        R 4  C 15        16:29  12/05/97
```

Report Record Format — (RPTSTATS)

Restored Data Sets Statistics Report

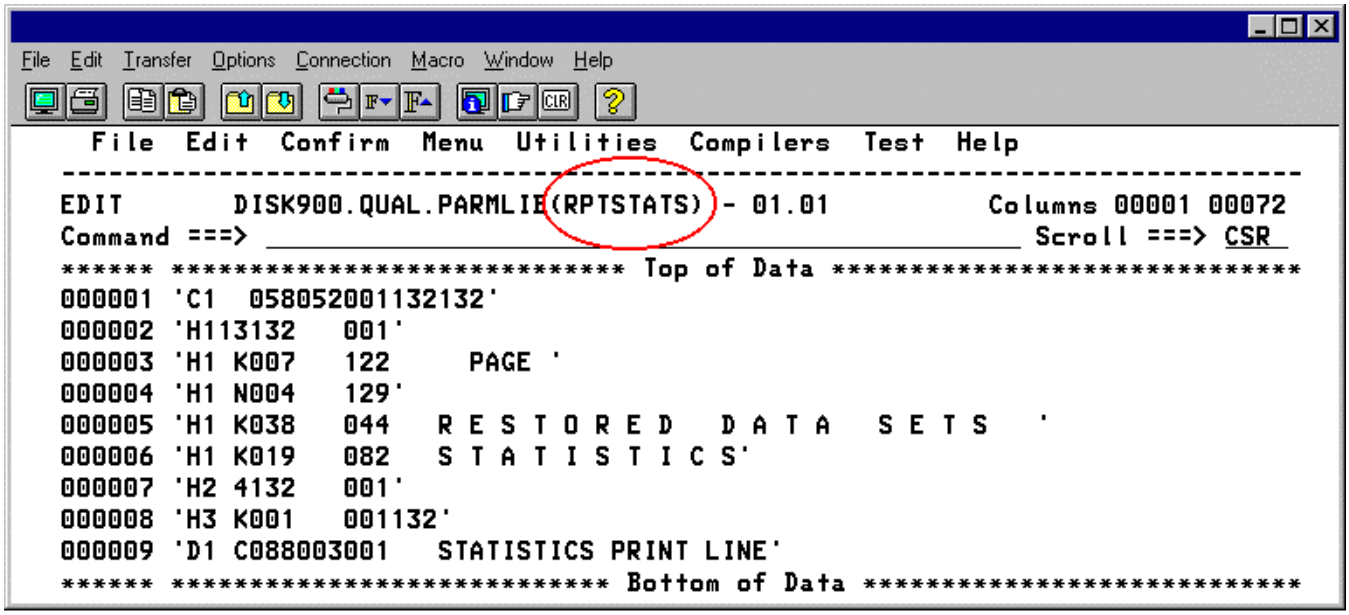
```
RPTSTATS
*
*      Mapping of report record used to print Restored Data Sets Statistics report
*
*
STAREC   DS    0F                    DEFINE RECORD HEADER
STACONLN DS    H                     LENGTH OF RECORD (STARECLN)
         DS    H                     ZEROS
STATABLE DS    CL10                  TABLE NAME ('RPTSTATS')
STADATA  DS    CL90                  REPORT RECORD
STARECLN EQU   *-STAREC              LENGTH OF REPORT RECORD
```

Table Used to Print Restored Data Sets Statistics Report

The following figure displays the source table used for printing when the following command is executed:

```
//DSCL     FMS
//SYSIN     DD  *
SET MOD=SIM
SCAN DSNINDEX
   SELECT VOLUMES=
      RECOVER REPORT
```

FIGURE 61. Sample Parmlib Member RPTSTATS



Report Record Format — (RPTPLST2)

Restored Data Sets Tape Pull List Report

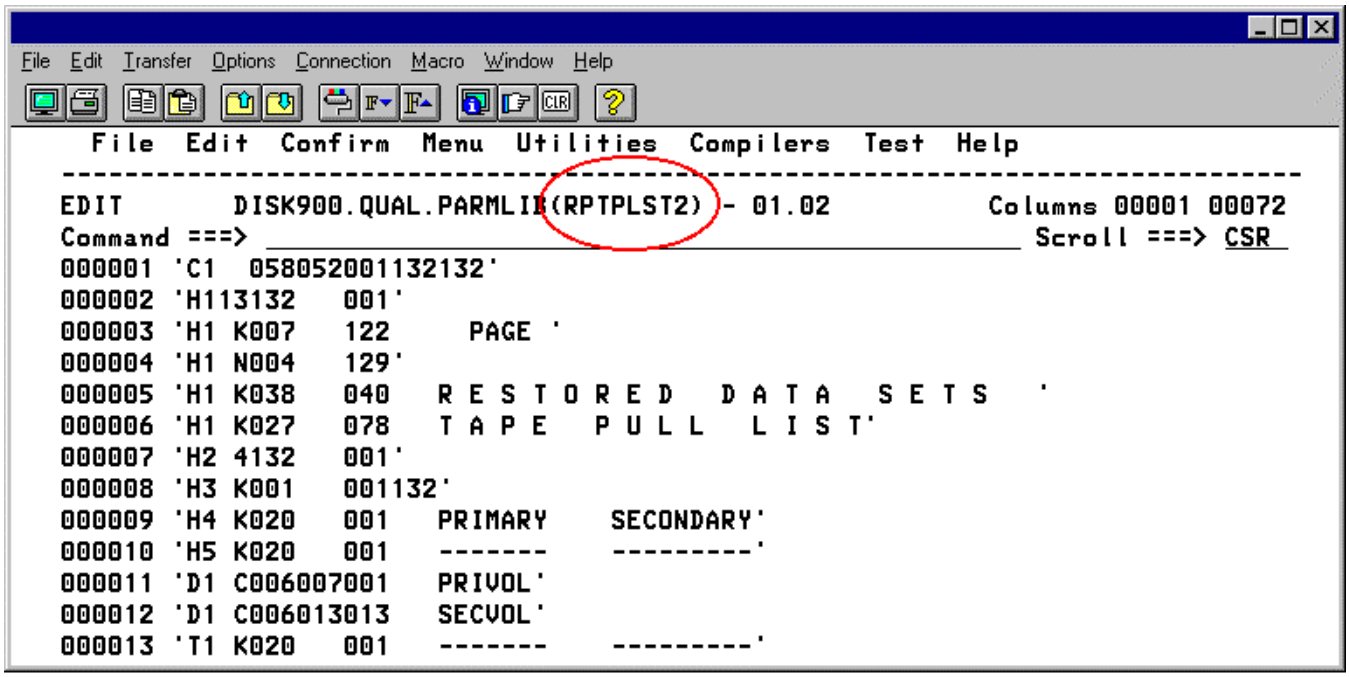
```
RPTPLST2
*
*      Mapping of report record used to print Restored Data Sets Tape Pull List report
*
TPREC    DS    0F                    DEFINE RECORD HEADER
TPCONLN  DS    H                     LENGTH OF RECORD (TPRECLN)
         DS    H                     ZEROS
TPTABLE  DS    CL10                  TABLE NAME ('RPTPLST2')
TPDATA   DS    0CL18
TPKEY    DS    CL6          1        KEY FOR SORTING
TPPVOL   DS    CL6          7        PRIMARY TAPE VOLSER
TPSVOL   DS    CL6         13        SECONDARY TAPE VOLSER
TPRECLN  EQU   *-TPREC               LENGTH OF REPORT RECORD
```

Table Used to Print Restored Data Sets Tape Pull List Report

The following figure displays the source table used for printing when the following command is executed:

```
//DSCL     FMS
//SYSIN     DD  *
SET MOD=SIM
SCAN DSNINDEX
   SELECT VOLUMES=
      RECOVER REPORT
```

FIGURE 62. Sample Parmlib Member RPTTPUL



Report Record Format — (VSAMCLDATA)

VSAM Cluster Data Record

```
VSAMCLDATA
*       Mapping of report record used to print VSAM CLD Report *
VSAMREC  DC    H''2157'',H''0''
         DC    CL2'**'
         DC    CL8'ADSVS026'        REPORT MODULE
C$NAME   DS    CL44                 CLUSTER NAME
C$CRDT   DS    XL4                  CLUSTER CREATION DATE
C$EXDT   DS    XL4                  CLUSTER EXPIRATION DATE
C$VOLS   DS    CL6                  PRIM. VOLSER OF DATA COMPONENT
C$VOLT   DS    XL2                  PRIM. VOLSER'S DEVICE TYPE
*       NOTE: THE CLUSTER'S PASSWORDS WILL NOT BE PRESENT IN THIS
*       RECORD IF THE USER IS NOT AUTHORIZED TO SEE THEM
C$PASS   DS    0CL32                CLUSTER'S PASSWORDS
         DS    4CL8
C$TRKS   DS    F                    TOTAL TRACKS ALLOC. TO CLUSTER
C$IDLE   DS    F                    TOTAL NUMBER OF IDLE TRACKS
C$PCTUSE DS    H                    CLUSTER PERCENT USED
C$PASATT DS    H                    PASSWORD ATTEMPTS ALLOWED
C$NOEXT  DS    H                    TOTAL NUMBER OF EXTENTS
*       THE FORMAT OF THE EACH EXTENT IS: *
*          1. TYPE OF EXTENT XL2 X'10' = INDEX *
*          2. LOW CCHH XL4 *
*          3. HIGH CCHH XL4 *
*          4. TRACKS IN EXTENT HW *
*          5. VOLSER CL6 *
*          6. DEVICE TYPE XL2 *
C$EXTENT DS    0XL640               DATA AND INDEX EXTENT INFO.
C$EXT    DS    32XL20
C$LEXT   DS    0C                   ADDRESS OF LAST EXTENT
D$ATTR   DS    XL4                  DATA AND CLUSTER ATTRIB. BYTES
         ORG   D$ATTR
D$$ATTR  DS    XL2                  ATTR1 AND ATTR2 OF DATA COMP.
D$ATTR3  DS    X                    ATTRIBUTE FLAGS
D$ATTR4  DS    X                    ATTR1 OF CLUSTER COMPONENT
D$UCAT   EQU   X'F0'                RECORD DESCRIBES A USER CATALOG
D$PAGE   EQU   X'01'                CLUSTER IS A PAGE DATA SET
D$SWAP   EQU   X'02'                CLUSTER IS A SWAP DATA SET
D$ALLOC  DS    XL7                  ALLOCATION QUANTITY AND TYPE
         ORG   D$ALLOC
D$PRIM   DS    XL3                  PRIMARY ALLOCATION AMOUNT
D$SEC    DS    XL3                  SECONDARY ALLOCATION AMOUNT
D$TYPALO DS    X                    ALLOCATION TYPE
D$CYLALO EQU   X'C0'                ALLOCATION IS IN CYLINDERS
D$TRKALO EQU   X'80'                ALLOCATION IS IN TRACKS
D$RECSZ  DS    F                    AVERAGE LOGICAL RECORD SIZE
D$AMD    DS    XL96                 AMDSB STATISTICS
D$HARBA  DS    FL8                  HIGH ALLOC. RBA OF DATA COMP.
D$HURBA  DS    FL8                  HIGH USED RBA OF DATA COMP.
D$TRKS   DS    F                    TRACKS ALLOCATED TO DATA COMP.
D$IDLE   DS    F                    IDLE TRACKS IN DATA COMP.
D$PCTUSE DS    H                    PERCENT USED OF DATA COMP.
```

```
I$ATTR   DS    XL4                 INDEX ATTRIBUTE BYTES
         ORG   I$ATTR
I$$ATTR  DS    XL2                 ATTR1 AND ATTR2 OF INDEX COMP.
I$ATTR3  DS    X                   NOT USED
I$ATTR4  DS    X                   NOT USED
I$ALLOC  DS    XL7                 ALLOCATION QUANTITY AND TYPE
         ORG   I$ALLOC
I$PRIM   DS    XL3                 PRIMARY ALLOCATION AMOUNT
I$SEC    DS    XL3                 SECONDARY ALLOCATION AMOUNT
I$TYPALO DS    X                   ALLOCATION TYPE
I$CYLALO EQU   X'C0'               ALLOCATION IS IN CYLINDERS
I$TRKALO EQU   X'80'               ALLOCATION IS IN TRACKS
I$AMD    DS    XL96                INDEX AMDSB STATISTICS
I$HARBA  DS    FL8                 HIGH ALLOC. RBA OF INDEX COMP.
I$HURBA  DS    FL8                 HIGH USED RBA OF INDEX COMP.
I$TRKS   DS    F                   INDEX COMPONENT TRACKS
I$IDLE   DS    F                   IDLE TRACKS IN INDEX COMP.
I$PCTUSE DS    H                   PERCENT USED OF INDEX COMP.
D$NAME   DS    CL44                DATA COMPONENT NAME
I$NAME   DS    CL44                INDEX COMPONENT NAME
C$RVOL   DS    CL6                 CRA VOLSER FOR CLUSTER
D$RVOL   DS    CL6                 CRA VOLSER FOR DATA COMPONENT
I$RVOL   DS    CL6                 CRA VOLSER FOR INDEX COMPONENT
C$TYPE   DS    C                   TYPE OF CLUS. (ENTYPE FR CAT.)
C$CLUSTR EQU   C'C'                RECORD DESCRIBES A BASE CLUSTER
C$AIX    EQU   C'G'                RECORD DESCRIBES AN AIX
C$SECF   DS    X                   SECURITY FLAG (X'80' = RACF)
D$SECF   DS    X                   SECURITY FLAG (X'80' = RACF)
I$SECF   DS    X                   SECURITY FLAG (X'80' = RACF)
C$CAT    DS    CL44                CAT. WHERE COMPONENT IS DEFINED
C$FLAGS  DS    XL10                FLAGS (REFER TO DMSFLAGS MACRO)
C$ASCCNT DS    H                   NUMBER OF ASSOCIATION FIELDS
C$ASC#   EQU   20                  NUMBER OF ASSOC FIELDS
C$ASCNAM DS    CL(C$ASC#*45)       ASSOC FIELDS - TYPE + NAME
D$USDT   DS    XL3                 LAST USE DATE (BINARY)
VSSELECT DS    CL50                GROUPS SPECIFIED ON RPT SELECT
```

Report Record Format — (RPTPLST1)

Restored Data Sets Disk Archive List Report

```
RPTDAPUL
*
*      Mapping of report record used to print Restored Data Sets Disk
*      Archive List report
*
*
DAKREC   DS    0F                          DEFINE RECORD HEADER
DAKCONLN DS    H                           LENGTH OF RECORD (DAKRECLN)
         DS    H                           ZEROS
DAKTABLE DS    CL10                        TABLE NAME ('RPTPLST1')
DAKDATA  DS    0CL56
DAKDVOL  DS    CL6    1                    DISK VOLSER
DAKKEY   DS    CL6    7                    DISK ARCHIVE KEY
DAKDSN   DS    CL44   13                   DISK ARCHIVE KEY
DAKRECLN EQU   *-DAKREC                    LENGTH OF REPORT RECORD
```

Table Used to Print Restored Data Sets Disk Archive List Report

The following figure displays the source table used for printing when the following command is executed:

```
//DSCL     FMS
//SYSIN     DD  *
SET MOD=SIM
SCAN DSNINDEX
   SELECT VOLUMES=
      RECOVER REPORT
```

FIGURE 63. Sample Parmlib Member RPTDAPUL

```
File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
--------------------------------------------------------------------------------
EDIT       DISK900.QUAL.PARMLIB(RPTPLST1) - 01.06          Columns 00001 00072
Command ===> _____________________________________________ Scroll ===> CSR
000001 'C1  058052001132132'
000002 'H113132   001'
000003 'H1 K007   122     PAGE '
000004 'H1 N004   129'
000005 'H1 K038   040   R E S T O R E D   D A T A   S E T S   '
000006 'H1 K033   078   D I S K   A R C H I V E   L I S T'
000007 'H2 4132   001'
000008 'H3 K001   001132'
000009 'H4 K021   001   VOLSER  ARCHIVE KEY  '
000010 'H4 K021   022   ARCHIVE DATA SET NAME'
000011 'H5 K050   001   ------  -----------  ----------------------------'
000012 'H5 K015   051   ---------------'
000013 'D1 C006001001   DSKVOL'
000014 'D1 C006007011   ARCKEY'
000015 'D1 C044013022   ARCDSN'
000016 'T1 K050   001   ------------------------------------------------'
000017 'T1 K015   051   ---------------'
000018 'T2 K023   001   TOTAL ARCHIVE DATA SETS'
000019 'T2 A000000024007Z  COUNTER'
****** **************************** Bottom of Data ****************************
4 ▌                      Aa   A TCPIP        R 4  C 15        10:58   4/13/98
```

Report Record Format — (RPTDSNEX)

VBACKUP Extent Report

```
RPTDSNEX
*
*      Mapping of report record used to print the DATA SET
*      EXTENT report created by VBACKUP (VSAM portion)
*
WRTVSAM  DS    0CL130
WRTVSLT  DC    XL2'00'                    LENGTH OF RECORD
WRTVSFL1 DC    XL2'00'                    FILLER
WRTVSRPT DC    CL10'RPTDSNEXNT'           REPORT NAME
WRTVSVOL DC    CL6' '                     VOLSER
WRTVSSDN DC    CL44' '                    DSN FOR SORT
WRTVSSDS DC    CL3' '                     DSORG FOR SORT
WRTVSSEX DC    XL8'0'                     EXTENT INFO FOR SORT
WRTVSDSN DC    CL44' '                    DSN FOR PRINT
WRTVSDSO DC    CL3' '                     DSORG FOR PRINT
WRTVSEXT DC    XL8'0'                     EXTENT INFO FOR PRINT
*
RPTDSNEX
*
*      Mapping of report record used to print the DATA SET
*      EXTENT report created by VBACKUP (non-VSAM portion)
*
WRTNVSM  DS    0CL130
WRTNVLT  DC    XL2'00'                    LENGTH OF RECORD
WRTNVFL1 DC    XL2'00'                    FILLER
WRTNVRPT DC    CL10'RPTDSNEXNT'           REPORT NAME
WRTNVVOL DC    CL6' '                     VOLSER
WRTNVSDN DC    CL44' '                    DSN FOR SORT
WRTVNSDS DC    CL3' '                     DSORG FOR SORT
WRTNVSEX DC    XL8'0'                     EXTENT INFO FOR SORT
WRTNVDSN DC    CL44' '                    DSN FOR PRINT
WRTNVDSO DC    CL3' '                     DSORG FOR PRINT
WRTNVEXT DC    XL8'0'                     EXTENT INFO FOR PRINT
```

Table Used for Printing VBACKUP Extent Report

The following figure displays the first page of the source table used for printing when the following command is executed:

```
//DSCL      DMS
//SYSIN      DD   *
SCAN REALVOLS
   SELECT VOLUMES=
      VBACKUP DSNENQ=,RESERVE=,RETPD=,EXPDT=,RPT=,LSTAR
```

FIGURE 64. Sample Parmlib Member RPTDSNEX

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT                    .PARMLIB(RPTDSNEX) - 01.01          Columns 00001 00072
 Command ===> _____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000001 'C1  058052001132130006'
 000002 'H113132   001   STANDARD HEADER - LINE 1'
 000003 'H1 K007   122     PAGE '
 000004 'H1 N004   129   PAGE NO'
 000005 'H1 K052   030   D A T A S E T   E X T E N T   R E P O R T   F O R   '
 000006 'H1 K013   082   V O L U M E  '
 000007 'H1 C006001096   VOLSER'
 000008 'H2 4132   001   STANDARD HEADER - LINE 2'
 000009 'H3 K001   001132'
 000010 'HC K013   001   DATA SET NAME'
 000011 'HC K003   048   DSO'
 000012 'HC K008   053   BEG CCHH'
 000013 'HC K008   063   END CCHH'
 000014 'HD K001   001132'
 000015 'D2 C044062003      DSN'
 000016 'D2 C003106048      DSORG'
 000017 'D2 X004109053004Z  BEGINNING CCHH'
 000018 'D2 X004113063004Z  ENDING CCHH'
 ****** ***************************** Bottom of Data ***************************
 4 ■                        Aa   A TCPIP        R 4  C 15        16:56  12/05/97
```

Subfile Record Format — (ARCHCMDS)

Deferred Archive Requests

```
ARCHCMDS DSECT
*
*      ARCHCMDS SUBFILE RECORD FORMAT (Deferred Archive Requests)
*
RCDSNLR  DS    0CL164
ARCNUNAM DS    CL44            0 NEW DATA SET NAME
         DS    AL1(0)         44
ARCJOBID DS    CL8            45 JOB ID
ARCREQTM DS    XL2            53 TIME REQUESTED
ARCREQDT DS    0XL3           55 DATE REQUESTED
ARCREQY  DS    X              55 YEAR REQUESTED
ARCREQDD DS    H              56 DAY REQUESTED
ARCRUNTM DS    XL2            58 ARCHIVE TIME
ARCRUNDT DS    XL3            60 ARCHIVE DATE
ARCFLAGS DS    XL2            63 FLAGS, X'0000' PENDING
                                        X'0001' COMPLETED OK
                                        X'0002' FAILED
                                        X'0004' ACTIVE
ARCMSGID DS    CL4            65 MESSAGE NUMBER
ARCVSLST DS    0CL30          69 VOLSERS FROM WHICH ARCHIVED
ARCVOLNO DS    5CL6           69
ARCDSNAM DS    CL44           99 DATA SET NAME
ARCEXPDT DS    XL3           143 EXPIRATION DATE FOR COMMAND
ARCRETPD DS    H             146 RETENTION PERIOD FOR COMMAND
ARCMBRCT DS    H             148 MEMBER COUNT
ARCDISP  DS    0CL3          150 DISPOSITION FROM COMMAND
ARCNOSCR DS    C             150 N=NOSCRATCH, S=SCRATCH
ARCNOCAT DS    C             151 N=NO ACTION, U=UNCAT, R=RECAT
ARCRECAT DS    C             152 E=EXPLICIT REQUEST
ARCPSWRD DS    CL8           153 PASSWORD
ARCDELDT DS    XL3           161 DATE TO DELETE FROM FILE
```

Subfile Record Format — (ARCHVOLS)

```
ARCHVOLS DSECT
*
*     archvols subfile record format (see archvol2 macro)
*
ARCKEY   DS    0CL6        + 0    ARCHVOL RECORD KEY (VOLSER,IF TAPE)
ARCID    DS    CL1         + 0    ARCHVOL KEY ID CHAR (LEN 1)
ARCNBR   DS    CL5         + 1    ARCHVOL KEY NUMERIC (LEN 5)
ARCKEYCH DS    CL6         + 6    KEY OF NEXT ARCHVOL REC CHAINED
ARCVOLUM DS    CL6         +12    ACTUAL VOLUME OF ARCHIVE DATA SET
ARCDVTYP DS    XL4         +18    DEVICE TYPE (FROM UCB)
ARCSEQNO DS    XL1         +22    SEQUENCE NUMBER OF THIS RECORD
ARCFORMT DS    XL1         +23    ARCHVOLS REC FORMAT (NEW = X'02')
ARC#BYTE DS    XL6         +24    # BYTES ON ARCHIVE VOLUME
*-----------------------------------------------------------------------
* THE ARC#BYTE FIELD IS NOT A TRUE 48 BIT SIGNED/UNSIGNED NUMBER.
* THE FIRST TWO BYTES ARE THE RESULT OF DIVIDING THE ACTUAL
* NUMBER OF BYTES BY 2147483648.  THE LOW ORDER WORD IS THE
* REMAINDER OF THE DIVISION.  THIS FIELD SHOULD NOT BE ADDED OR
* SUBTRACTED WITHOUT FIRST USING THE DCVD MACRO TO CONVERT IT TO
* DECIMAL AND THE ARITHMETIC OPERATION PERFORMED IN DECIMAL.
*
* THE TRUE NUMBER OF BYTES CAN BE COMPUTED BY MULTIPLYING THE
* FIRST TWO BYTES BY 2,147,483,648 AND THEN ADDING THE LOW ORDER
* WORD TO THE PRODUCT OF THE MULTIPLICATION.
*-----------------------------------------------------------------------
ARCDSNAM DS    CL44        +30    DATA SET NAME
ARCDSCNT DS    H           +74    NBR DATASETS ON TAPE
ARCBKCNT DS    F           +76    NBR BLOCKS ON VOLUMES
ARCBTCNT DS    F           +80    OLD NBR BYTES ON VOLUMES
ARCCTIME DS    XL4         +84    ARCHIVE TIME HHMMSSTH (UNSIGNED PKED)
ARCCDATE DS    XL3         +88    CREATION DATE YYDDDD (DISCONT BINARY)
ARCEXPDT DS    XL3         +91    EXPIRATION DATE YYDDDD
ARCFEET  DS    H           +94    FEET USED ON TAPE
ARCCOUNT DS    AL1         +96    VOLUME COUNT (COUNT OF VOLUMES AT +0)
ARCCPYKY DS    CL6         +97    COPY ARCKEY
*
ARCFLAG  DS    XL1         +103   INDICATORS
ARCFDISA EQU   X'80'              VOLUME IS DISABLED
ARCFCOPY EQU   X'40'              VOLUME IS COPY VOLUME
ARCFPASS EQU   X'20'              VOLUME IS PASSWORD PROTECTED
ARCFERR  EQU   X'10'              CLOSE FAILED OR I/O BKSPACE FAILED
ARCFNCLO EQU   X'08'              VOLUME NOT CLOSED, ARCHVOLS INCOMPLETE
ARCFFULL EQU   X'04'              VOLUME FULL (NEXT DSN DID NOT FIT)
ARCFMERG EQU   X'02'              VOLUME CREATED BY MERGE
ARCF6250 EQU   X'01'              VOLUME WRITTEN AT 6250 BPI
*
ARCFLAG2 DS XL1            +104
ARCDYN1  EQU X'80'                M860 or Redwood FLAGS
ARCDYN2  EQU X'40'
ARCDYN3  EQU X'20'
ARCVOLBK EQU X'10'                VOLUME LEVEL BACKUP
ARCIDRC  EQU X'08'                COMPACTION BIT
ARCXCOPY EQU X'04'                XCOPY HAS PROCESSED THIS ARCHVOL
ARCBYTFG EQU X'02'                ARC#BYTE PRESENT
ARCFDEVT EQU X'01'                ARCDDEVT PRESENT
```

```
*
ARCFLAG3 DS  XL1           +105
ARCP2    EQU X'80'                PRIMARY2 FLAG
*
         DS CL1            +106  (RESERVED FOR FUTURE USE)
ARCDYNN1 EQU B’00000001’         DYN1 SPECIAL DEVICE TYPE
ARCDYNN2 EQU B’00000010’         DYN2 SPECIAL DEVICE TYPE
ARCDYNN3 EQU B’00000011’         DYN3 SPECIAL DEVICE TYPE
ARCDYNN4 EQU B’00000100’         DYN4 SPECIAL DEVICE TYPE
ARCDYNN5 EQU B’00000101’         DYN5 SPECIAL DEVICE TYPE
ARCDYNN6 EQU B’00000110’         DYN6 SPECIAL DEVICE TYPE
ARCDYNN7 EQU B’00000111’         DYN7 SPECIAL DEVICE TYPE
ARCDYNN8 EQU B’00001000’         DYN8 SPECIAL DEVICE TYPE
ARCDYNN9 EQU B’00001001’         DYN9 SPECIAL DEVICE TYPE
*
ARCRATIO DS BL1            +107  IDRC COMPRESSION RATIO ON TAPE/CART
ARCDDEVT DS XL2            +108  TDDC DEVICE TYPE
```

Subfile Record Format — (DASDSPCB)

DASD BILLING

```
DASDSPCB DSECT
*
*      DASDSPCB SUBFILE RECORD FORMAT (DASD BILLING)
*
DSBWORK  DS    0CL84
DSBVOLNO DS    CL6            VOLSER
DSBDSNAM DS    CL44           DATA SET NAME
DSBNOSCR DS    X              SCRATCH COUNT (obsolete)
DSBNOSCN DS    X              SCAN COUNT
DSBACCUM DS    F              ACCUMULATED AMOUNT
DSBCREDT DS    XL3            CREATION DATE
DSBSCNDT DS    XL3            LAST SCAN DATE
DSBUSEDT DS    XL3            LAST USED DATE
DSBFLAGS DS    X              X'80' - SCRATCHED
DSBUSRCD DS    CL12           USER CODE
DSBDEVCL DS    XL2            DEVICE CLASS
DSBALLOC DS    H              CURRENT ALLOCATION IN TRACKS
DSBMAXUS DS    H              LARGEST NUMBER OF TRACKS USED
```

Subfile Record Format — (DMSPARMS)

TSO Dynamic Restore

```
DMSPARMS DSECT
*
*      DMSPARMS SUBFILE RECORD FORMAT (TSO DYNAMIC RESTORE)
*
DMSRECOR DS    0CL100        0
DMSIDENT DS    CL8           0 ID FIELD OF 'TSOTAPES'
DMSLIMIT DS    PL2           8 LIMIT OF CONCURRENT DYNAMIC RESTORES
DMSCOUNT DS    PL2          10 DYNAMIC RESTORES IN PROGRESS
DMSUSRID DS    11CL8        12 USER IDS DOING DYNAMIC RESTORES
```

Subfile Record Format — (DMSPOOLS)

For Tapes in the Tape Pools

```
DMSPOOLS DSECT
*
*      DMSPOOLS SUBFILE RECORD FORMAT (for tape pools)
*      OFFSET
POOLREC  DS    0CL90         0 DMSPOOLS SUBFILE RECORD
POOLKEY  DS    0CL26         0 KEY TO THE SUBFILE RECORD
POLKYNAM DS    CL8      +0     POOL NAME
POLKYRID DS    CL1      +8     RECORD ID. (S)CRATCH OR IN (U)SE
POLKYVOL DS    CL6      +9     'SCRTCH' OR ACTUAL VOLSER ,
*                              depends on TPOOLSEQ sysparm
POLKYDTO DS    ZL5      +15    (OLD) DATE OF UPDATE  (YYDDD)
POLKYTMO DS    PL4      +20    (OLD) TIME OF UPDATE (HHMMSSTH)
POLKYRSV DS    CL2      +24    (OLD) 2-BYTES RSRVD IN KEY (SPACES)
        ORG   POLKYDTO         REDEFINE DATE/TIME FIELDS FOR Y2K
POLKYDTE DS    ZL7      +15    DATE OF UPDATE  (YYYYDDD)
POLKYTIM DS    PL4      +22    TIME OF UPDATE  (HHMMSSTH) UNSIGNED
POLSERNO DS    CL6      +26    POOL VOLUME SERIAL NO.
POLFLAG1 DS    XL1      +32    FLAG BYTE
POLY2KDT EQU   X'80'           ON = KYDTE(YYYYDDD),KYTIM(PL4),KYRSV(NULL)
*                              OFF = KYDTE(YYDDD),KYTIM(PL4),KYRSV(SPACES)
POLBLANK EQU   X'40'           UNRELIABLE, OLD FORMAT ALWAYS CLEARED THIS
*                              FLAG BYTE TO BLANKS, SETTING THIS BIT ON.
POLRSV2  DS    CL57     +33    57-BYTES RESERVED DATA (SPACES)
```

Subfile Record Format — (DSNINDEX)

For Archived Data Sets

```
DSNINDEX DSECT
*
*     DSNINDEX SUBFILE RECORD FORMAT (SEE DSNINDEX MACRO)
*
         DS    0D
&NAME    DS    0CL256
DSNDSNAM DS     CL44        + 0     DATA SET NAME
DSNVOLNO DS     CL6         +44     VOLSER FROM WHICH ARCHIVED
DSNATIME DS     XL2         +50     ARCHIVE TIME (UNSIGNED PACKED=HHMM)
DSNMPBLK DS     XL2         +52     MAX PHYSICAL BLK IN DATA SET (OBSOLETE)
         ORG    DSNMPBLK
DSNDSKTT DS     XL2         +52     TT (OF TTRZ) OF DISK CNTRL RECORD
DSNARCDT DS     XL3         +54     ARCHIVE DATE (DISCON. BIN. YDD)
DSNEXPDT DS    0XL3         +57     EXPIRATION DATE (DISCON. BIN. YDD)
DSNEXPY  DS     XL1         +57     YEAR
DSNEXPDD DS     XL2         +58     DAYS
*
DSNDSORG DS    0XL2         +60     DATA SET ORGANIZATION (SAME AS F1 DSCB)
DSNDSRG1 DS     XL1         +60     FIRST BYTE OF DATA SET ORGANIZATION
DSNIS    EQU    X'80'               DSORG = IS (ISAM)
DSNPS    EQU    X'40'               DSORG = PS (PHYSICAL SEQUENTIAL)
DSNDA    EQU    X'20'               DSORG = DA (DIRECT ACCESS)
DSNUCAT  EQU    X'08'               ICF USER CATALOG
DSNDSDS  EQU    X'04'               DSS-dumped dataset (for ADSTH001)
DSNPO    EQU    X'02'               DSORG = PO (PARTITIONED)
DSNU     EQU    X'01'               DSORG = U (UNMOVABLE)
DSNDSRG2 DS     XL1         +61     SECOND BYTE OF DATA SET ORGANIZATION
DSNAM    EQU    X'08'               DSORG = AM (VSAM)
*
DSNDEVCL DS     X           +62     DEVICE CLASS
DSNSCALO DS     X           +63     TYPE OF ALLOCATION
DSNPRTRK DS     XL2         +64     PRIMARY TRACKS ALLOCATED
*      PRECEEDING SECTION OF 66     BYTES IS COMMON WITH ARC CONTROL RECORD
DSNDRBLK DS     XL2         +66     DIRECTORY BLOCK COUNT
DSNMMBRS DS     XL2         +68     MEMBER COUNT
DSNFILNO DS     XL2         +70     FILE NUMBER
DSNBLKNO DS    0XL4         +72     FIRST BLOCK (AND PRV FOR 3480'S)
DSNPRV   DS     XL1         +72     PRV FOR 3480'S
DSNRELBK DS     XL3         +73     ALWAYS FIRST RELATIVE BLOCK NUMBER
DSNBYTES DS     XL4         +76     BYTE COUNT
DSNBLKCO DS     XL2         +80     OLD BLOCK COUNT FIELD
DSNARVOL DS     CL6         +82     VOLSER ARCHIVED TO (PRE-RELEASE 7.0)
         ORG    DSNARVOL
DSNARKEY DS     CL6         +82     KEY TO ARCHVOLS REC (POST-REL 7.0)
*
DSNFLAGS DS    0XL10        +88     FLAGS
*
DSNFLAG1 DS     XL1         +88     FLAG BYTE 1
DSNFLGSC EQU    X'80'               DSN SCRATCHED WHEN ARCHIVED
DSNFLGUC EQU    X'40'               DSN UNCATALOGED WHEN ARCHIVED
DSNFLGRC EQU    X'20'               DSN RECAT TO PSEUDO VOLSER WHEN ARCHIVED
```

```
DSNFLGAR EQU    X'10'               ARCHIVED DUE TO EXPLICIT ARCHIVE REQUEST
*      IF BIT X'10' IS OFF, ARCHIVED BY RETAIN
DSNFLGEX EQU    X'08'               ARCHIVED USING DASD EXCP SUPPORT
DSNFLGBK EQU    X'04'               ARCHIVED WITH INCREMENTAL BACKUP
DSNFLGUV EQU    X'02'               DA-U/V DATASET OR PS-U/V BACKED UP AS DA
DSNFLGVV EQU    X'01'               DSN RESIDED ON VIRTUAL VOLUME
*
DSNFLAG2 DS     XL1         +89     FLAG BYTE 2
DSNFLGEN EQU    X'80'               HAD GENERIC PROFILE WHEN ARCHIVED
*      (GENERIC, NO DISCRETE PROTECTION)
DSNFLGRF EQU    X'40'               HAD DISCRETE PROFILE WHEN ARCHIVED
DSNFLGAX EQU    X'20'               DATA SET IS A VSAM ALTERNATE INDEX
DSNFLGCB EQU    X'10'               DATA SET IS A CATALOG BACKUP
DSNFLGIO EQU    X'08'               DATA SET BACKED UP WITH IO ERRORS
DSNFLGPE EQU    X'04'               DSORG IS PDSE (PARTITIONED EXTENDED)
DSNFLGBT EQU    X'02'               RACF BIT WAS ON WHEN ARCHIVED
DSNFLGET EQU    X'01'               EXTENT TRACK IMAGE BACKUP
*
DSNFLAG3 DS     X +90               FLAG BYTE 3
DSNBLKFG EQU    X'80'               NEW DSNBLKCT FIELD HAS BEEN UPDATED
DSNSTRIP EQU    X'40'               DSORG IS PSE (EXTENDED SEQUENTIAL)
DSNBSAM  EQU    X'20'               PSE ARCHIVED BY BSAM I/O
DSNMEMLA EQU    X'10'               MEMBER LEVEL ARCHIVE
DSNBYTFG EQU    X'08'               DSN#BYTE IS PRESENT
DSNSELMN EQU    X'07'               SAMS:SELECT MANAGED BACKUP
DSNSELIB EQU    X'04'               SAMS:SELECT INCR BACKUP
DSNSELDB EQU    X'02'               SAMS:SELECT DIFF BACKUP
DSNSELFB EQU    X'01'               SAMS:SELECT FULL BACKUP
*
DSNFLAG4 DS     X           +91     FLAG BYTE 4
UNCAT    EQU    X'80'               DS WAS NOT CATLGED AT BACKUP TIME
DSNDSS   EQU    X'04'               DSS-dumped dataset
DSNCC    EQU    X'02'               DS created using concurrent copy
*
DSNBFMT  DS     X           +92     FLAG BYTE 5
DSNBF00  EQU    0                   STANDARD FORMAT (USE ORIGINAL CODE)
DSNBF01  EQU    1                   NEW DFSMSdss BACKUP
*
DSNFLAG6 DS     X           +93     FLAG BYTE 6
DSNP2    EQU    X'80'               ..PRIMARY2 DSNINDEX
*
DSNFLAGN DS     XL4         +94     FLAG BYTE 7-10
*
         ORG    DSNFLAGS+9          ORG TO FLAG BYTE 10
DSNFLAGA DS     X           +97     FLAG BYTE 10
DSNERRXX EQU    X'40'               DO NOT USE! THIS FLAG WAS TURNED ON
*                                   IN ERROR IN THE EOFREC AND WAS FIXED
*                                   BY CHECKING FOR BYTES 9&10=X'4040'
*                                   AND ZEROING LAST 7 BYTES IF EQUAL.
DSNMSPFL DS     XL1         +98     MSP FLAGS
*
*-------------------------------------------------------------------
* THE DSNSEXPT FLAG IS USED TO CONVERT THE DSNARCDT FIELD INTO THE
* DSNLCOPY FIELD AND TO REFRESH THE DSNARCDT FIELD WITH ITS ORIGINAL
* CONTENTS. THE DSNUPDAT FLAG IS NOW USED TO IDENTIFY THE DSNLCOPY
* FIELD AS BEING UPDATED.
*-------------------------------------------------------------------
DSNSMSFL DS     XL1         +99     SMS FLAGS
```

```
DSNSEXPT EQU    X'80'                ARCDT ALTERED DUE TO LAST BK COPY
DSNMCBYP EQU    X'08'                MGMT CLASS BYPASSED
DSNEXPNS EQU    X'04'                ALTERED DATE FLAG
DSNUPDAT EQU    X'02'                ARCDT ALTERED DUE TO LAST BKUP CPY
*
DSNSMSIN DS    0CL32        +100    SMS CLASSES AND GROUP INFORMATION
DSNSMSDC DS     CL8         +100    SMS DATA CLASS
DSNSMSMC DS     CL8         +108    SMS MANAGEMENT CLASS
DSNSMSSC DS     CL8         +116    SMS STORAGE CLASS
DSNSMSSG DS     CL8         +124    SMS STORAGE GROUP
*
DSNDS1FL DS    0XL12        +132    FMT1 DATE FIELDS
DSNDS1CD DS     XL3         +132    CREATION DATE
DSNDS1MD DS     XL3         +135    LAST MOD DATE
DSNDS1UD DS     XL3         +138    LAST USED DATE
DSNDS1ED DS     XL3         +141    EXPIRATION DATE
DSNBLKCT DS     XL4         +144    DSNBLKCT REPLACES OLD BLOCK COUNT
DSNRESR1 DS     XL18        +148    RESERVED FOR FUTURE
DSNDS1JN DS     XL8         +166    JOBNAME
DSNLCOPY DS     XL3         +174    LAST BACKUP CPY DETECTED DATE
DSNRATIO DS     XL1         +177    IDRC REDUCTION RATE
DSN#BYTE DS     XL6         +178    # BYTES ON ARCHIVE VOLUME
*-----------------------------------------------------------------------
* THE DSN#BYTE FIELD IS NOT A TRUE 48 BIT SIGNED/UNSIGNED NUMBER.
* THE FIRST TWO BYTES ARE THE RESULT OF DIVIDING THE ACTUAL
* NUMBER OF BYTES BY 2147483648.  THE LOW ORDER WORD IS THE
* REMAINDER OF THE DIVISION.  THIS FIELD SHOULD NOT BE ADDED OR
* SUBTRACTED WITHOUT FIRST USING THE DCVD MACRO TO CONVERT IT TO
* DECIMAL AND THE ARITHMETIC OPERATION PERFORMED IN DECIMAL.
*
* THE TRUE NUMBER OF BYTES CAN BE COMPUTED BY MULTIPLYING THE
* FIRST TWO BYTES BY 2,147,483,648 AND THEN ADDING THE LOW ORDER
* WORD TO THE PRODUCT OF THE MULTIPLICATION.
*-----------------------------------------------------------------------
DSNRESR2 DS     XL25        +184    RESERVED FOR FUTURE
*
DSNALSTN DS     CL8         +209    USERID OF LAST UPDATE
DSNAREL# DS     CL6         +217    ARCHIVE RELEASE NUMBER
DSNRTDTE DS     XL3         +223    DATE OF LAST RESTORE
DSNRTIME DS     XL2         +226    RESTORE TIME (UNSIGNED PACKED=HHMM)
DSNRTCNT DS     XL1         +228    COUNT OF RESTORES
DSNVERS  DS     XL1         +229    VERSION NUMBER
DSNRESR3 DS     XL1         +230    RESERVED FOR FUTURE
DSNUSERF DS     XL25        +231    USER FIELD
```

Subfile Record Format — (MIGRECAT)

PS Migration to Tape

```
MIGRECAT DSECT
*
*      MIGRECAT SUBFILE RECORD FORMAT (PS Migration to Tape)
*      (control entries to recatalog datasets to backup tape)
*
CATREC   DS    0CL252        0
CATVOLNO DS     CL30         0 UP TO 5 PRIMARY VOLUMES
CATDSNAM DS     CL44        30 DATA SET NAME
CATDEVCL DS     XL4         74 DEVICE CLASS
CATVOLCT DS     H           78 VOLUME COUNT
CATCPYNO DS     CL30        80 UP TO 5 COPY TAPE VOLUME NUMBERS
CATNEWFS DS     H          110 FILE SEQUENCE NUMBER
CATCREDT DS     XL3        112 CREATION DATE OF MIGRECAT RECORD
CATEXPDT DS     XL3        115 EXPIRATION DATE
CATDDNAM DS     CL8        118 DDNAME FOR ORIGINAL VOLSER
CATVOLNO DS     CL6        126 ORIGINAL VOLSER
CATCPYN2 DS     CL120      132 REMAINING 20 COPY VOLUMES
```

Subfile Record Format — (RACFENCD)

RACF Encoded Profile Names

```
RACFENCD DSECT
*
*      RACFENCD SUBFILE RECORD FORMAT (for discrete profiles)
*
         DS    0CL90           (LAST 3 CHARS OF PROFILE DSN ALWAYS BLANK)
RACDSNAM DS     CL44       + 0 DATA SET NAME
RACARCDT DS     XL3        +44 ARCHIVE DATE (YDD)
RACATIME DS     XL2        +47 ARC TIME (UNSIGNED PACKED=HHMM)
RACPFNAM DS     CL44       +49 RACF PROFILE NAME
```

Subfile Record Format — (RESTCMDS)

Deferred Restore Requests

```
RESTCMDS DSECT
* RESTCMDS SUBFILE RECORD FORMAT (DEFERRED RESTORE REQUESTS)
         DS    0D
RESDSNLR DS    0CL164
RESNUNAM DS     CL44     + 0    NEW DATA SET NAME
RESFLAG1 DS     XL1      +44
RESCTLG  EQU    X'80'           SYSCATLG SPECIFIED ON CMD
RESY2KDT EQU    X'40'           ON  = RESTIME(PL2,UNSIGNED),DATE(XL3,BINARY)
                                OFF = RESTIME(ZL4,HHMM),RESDATE(ZL5,YYDDD)
*
RESJOBID DS     CL8      +45    JOB ID
RESREQTM DS     XL2      +53    TIME REQUESTED, QTIME
RESREQDT DS    0XL3      +55    DATE REQUESTED, QDATE
RESREQY  DS     XL1      +55    YEAR REQUESTED
RESREQDD DS     H        +56    DAY REQUESTED
RESRUNTM DS     XL2      +58    DISPOSE RESTORE TIME
RESRUNDT DS     XL3      +60    DISPOSE RESTORE DATE
RESFLAGS DS     XL2      +63    FLAGS. USED AS HALFWORD STATUS CODE.
RESPEND  EQU    X'0000'         0=REQUEST PENDING
RESCOMP  EQU    X'0001'         1=REQUEST COMPLETE
RESFAIL  EQU    X'0010'         2=REQUEST FAILED
RESACTV  EQU    X'0100'         4=REQUEST MARKED ACTIVE, MAY BE INUSE.
*
RESMSGID DS     CL4      +65    MESSAGE NUMBER
RESTOVOL DS     CL6      +69    VOLUME RESTORED TO
RESDSNAM DS     CL44     +75    ORIG DATA SET NAME FOR DSNINDEX GET
RESVOLNO DS     CL6     +119    ORIG FROM VOLUME
RESEXPDT DS     XL3     +125    EXPIRATION DATE FROM COMMAND
RESRETPX DS     H       +128    RETENTION PERIOD FROM COMMAND
RESMBRCT DS     H       +130    MEMBER COUNT
RESPOOL  DS     CL8     +132    POOL NAME
*           OLD ZONED FORMAT, PRE 134803
*  TIMDT DS    0XL5     +140    TIME AND DATE
*  TIME  DS     ZL4     +140    RESTORE TIME
*  DATE  DS    0CL5     +144    RESTORE DATE
*  DTYY  DS     CL2     +144    RESTORE YEAR
*  DTDDD DS     CL3     +146    RESTORE DAY
*           NEW XL2,XL3 FORMAT IN 134803
RESTIMDT DS    0XL5     +140    DSNINDEX TIME AND DATE
RESTIME  DS     XL2     +140    RESTORE TIME
RESDATE  DS     XL3     +142    RESTORE DATE
         DS     XL4     +145    OLD DATE FRAGMENT, NOW UNUSED
*
RESOPTNS DS    0CL4     +149   DISPOSITION OPTIONS FROM COMMAND
RESSCRAT DS     CL1     +149   U=NO SCRATCH,S=SCRATCH
RESNOCAT DS     CL1     +150   C=CATALOG,N=NOCATALOG,R=RECATALOG
RESCREAT DS     CL1     +151   N=LEAVE ORIG CREDT,C=RESET CREDT TO TODAY
RESERASE DS     XL1     +152   X'01=ERASE DIR
*
RESPSWRD DS     CL8     +153    PASSWORD
RESDELDT DS     XL3     +161    DATE TO DELETE RECORD FROM SUBFILE
```

Subfile Record Format — (RETEXCLD)

Grace Periods for Restored Data Sets

```
RETEXCLD DSECT
*
*      RETEXCLD SUBFILE RECORD FORMAT (Grace Periods for Restored
*      Data Sets)
*
EXCLDREC DS    0CL82
EXCDSNAM DS     CL44                    DATA SET NAME
EXCVOLNO DS     CL6                     VOLSER NUMBER
EXCEXPDT DS     XL3                     END DATE OF GRACE PERIOD
         DS     XL29                    FILLER
```

Record Formats — Files Data Set Log Records

Files Data Set Log Record Layout

RECORD 1:

```
AUDFDSNM DS    CL44       + 0          FILES DATA SET NAME
FORMATID DS    CL6        +44          161086
```

Note: FORMATID is for MFILES support. Each header contains 161086 in positions 45-50 so the new log format can be identified by ADSDM652 during FRECOVER, if SS04393 is applied, otherwise blank.

RECORD 2:

```
AUDITFIL DSECT
*
*      FILES DATA SET LOG RECORD FORMAT
*
AUDITDAT DS    XL4        + 0          DATE RECORDED (00YYDDDF)
AUDITIME DS    XL4        +04          TIME RECORDED (HHMMSSHH)
AUDJOBNM DS    CL8        +08          JOB NAME
AUDFUNCT DS    CL8        +16          DMS FUNCTION
AUDFDATE DS    XL4        +24          DMS FUNCTION START DATE
AUDFTIME DS    XL4        +28          DMS FUNCTION START TIME
AUDSFFUN DS    CL2        +32          SUBFILE FUNCTION
AUDSUBFL DS    CL8        +34          SUBFILE
AUDRECRD DS    XL0256     +42          THE UPDATED RECORD*
AUDXTRA  DS    XL02       +298         RESERVED
*      SEE SPECIFIC SUBFILE FOR SUBFILE RECORD DEFINITION
       If subfile function is 'SD' record can only contain
       the key for this entry.
```

Archive Tape Organization

The data sets archived during an execution of the archive support are stored as a single sequential data set with a record format of U (undefined). A file number, assigned to each data set at the time it is archived, is stored in the archive index and is used by CA-Disk to locate a given data set in order to restore it.

The access method used in accessing an archive tape is BSAM. Multiple buffering is employed during the reading and writing of data. The POINT macro is used to skip intervening blocks when positioning to a requested data set.

An internal algorithm is used to determine the approximate amount of data that can be carried on a single archive tape. If a data set for which an archive request is being processed will require more capacity than is estimated to remain on the current archive tape, a new archive tape will be mounted. This technique is used to confine each archived data set to a single reel of tape.

Note: Concurrent Copy is an exception to this rule. The BACKUPCC command will produce multi-volume ARCHVOLS, but only for one data set. In other words, if the current input data set will not fit on the current ARCHVOL, the BACKUPCC command will span across multiple volumes until the conclusion of the data set. At that time, the ARCHVOLS is closed and a new ARCHVOLS started. This is true with the exception of a VSAM sphere, which will span as many volumes as required.

Large data sets that exceed the capacity of a single tape volume will always start at the beginning of the first volume to which they are archived. The system will presently handle data sets that can be stored on up to 35 tape volumes. (For non-MVS users, the limit is five tape volumes.)

Archive Tape Format

The following section describes the format of data contained on the CA-Disk Archive Tape.

Archive tape volumes contain all information necessary to restore archived data sets. This information consists of both CA-Disk control information and physical blocks of the archived data. All information is stored on the archive volume as logical records.

Six different logical record types can occur in an archive file. These logical records are formatted together into physical blocks and are then written to the tape volume. More than one physical block can be needed to store all data for an archived data set.

Each physical block can contain both CA-Disk control records and user data blocks. Blocks are filled with as many logical records as will fit into the blocksize specified. Logical records will span physical blocks. Each partial record in the new block has a split record prefix. For integrity purposes, logical records belonging to different archived data sets will never be stored in the same physical block.

Each physical block contains an 8-byte prefix area. This area contains control and block verification data. Its contents are:

Offset	Length	Description
+0	2	Halfword length of physical block

Offset	Length	Description
+2	2	Halfword File Number. The file number starts at 1 and is incremented by one for every data set archived to the volume.
+4	4	Fullword Block Count. This count is the block number for this block within the archive volume.

Following the block prefix are the logical records that contain the control information and user data. Each logical record has a prefix that identifies its type and length. Following the logical record prefix is the logical record data. The first 8 bytes contain the record prefix; it will appear in all logical records. The format of logical record data differs for each logical record type. They are described later. The logical record prefix is formatted as follows:

Offset	Length	Description
+0	4	Fullword length of the logical record. This length includes the length of the record prefix.
+4	1	Record Type Indicator. A 1-byte identifier is assigned to each record type. Based on this record type, CA-Disk can determine the contents and format of the remaining data in the record.
+5	3	TTR information

The following is a list of the valid logical record types:

TABLE 186. List of Valid Logical Record Types

Type	Description
C	CA-Disk Control Record. Its the first record generated for each data set archived and the first record for each data set.
1	Format-1 DSCB Record. This record always follows the control record. It contains an image of the format-1 DSCB for the archived data set.
V	Volume/Catalog Data Record. This record contains volume and catalog information necessary to allocate non-VSAM data sets or define VSAM clusters.
D	PDS Directory Data (PDS data sets only). One record is generated for every directory block.
R	Data Set, Data Block. One logical record is generated for every physical block contained on the original data set that was archived.
A	Alias Record. One record is generated for every PDS alias entry in the PDS archived. This record type is only generated when non-EXCP support is used.
E	Logical End of Data Set. This is the last record generated for every archived data set.
S	Split record of Data Set. This is a partial record that has been split between blocks.
P	PDS information record. This record contains information relating to PDS anomalies and directory statistics.
U	User catalog (ICF) record. This record type contains the alias pointers connecting a master catalog to an ICF catalog, extracted during ICF catalog backup.
I	Encapsulated DFSMSdss data and control information.

Depending on the organization of the data set being archived and on the use of the full-track EXCP code or standard access method used to archive the data set, only certain record types will be present in the records on the archive tape. The following information indicates what records will be generated in the different cases:

TABLE 187. Record Types Generated for PS and DA

Type	Description
C	Control record
1	Format-1 DSCB
V	Volume/Catalog Data Record - allocation attributes
R	One or more physical data blocks
E	End of data set

For HFS data sets using the DFSMSdss Data Mover Interface, the following record types will be generated:

TABLE 188. Record Types Generated for HFS

Type	Descriptioin
C	Control record
1	Format-1 DSCB
V	Volume/Catalog Data Record - allocation attributes
P	Records format and version information
I	Encapsulated DFSMSdss data and control information
E	End of data set

For PO data sets using EXCP support, the following record types will be generated:

TABLE 189. Record Types Generated for PO

Type	Description
C	Control record
1	Format-1 DSCB
1	Volume/Catalog Data Record - allocation attributes
P	One or more directory statistics and anomalies block
D	One or more directory block
R	One or more physical data blocks. A data block with all x'EE' bytes indicates the end of data for a member.
E	End of data set

For PO (PDS and PDSE) data sets using non-EXCP support, the following record types will be generated:

TABLE 190. Record Types Generated for PDS and PDSE

Type	Description
C	Control record
1	Format-1 DSCB
V	Volume/Catalog Data Record - allocation attributes
D	Directory entry for each member

TABLE 190. Record Types Generated for PDS and PDSE

Type	Description
A	Entry for each alias entry
R	One or more physical data blocks. A series of D, A and R type records will be repeated.
E	End of data set

For ISAM data sets, the following record types will be generated:

TABLE 191. Record Types Generated for ISAM

Type	Description
C	Control record
1	Format-1 DSCB
V	Volume/Catalog Data Record - allocation attributes
R	Logical records from QISAM access method
E	End of data set

For AM (VSAM cluster) data sets, the following record types will be generated:

TABLE 192. Record Types Generated for VSAM

Type	Description
C	Control record
1	Format-1 DSCB (dummy)
V	Volume/Catalog Data Record - cluster attributes
R	One or more logical data records. The first is cluster catalog information used to allocate the cluster. Others are logical records returned by VSAM.
U	One or more alias entries from the master catalog (present only for backup copies of ICF catalogs)
E	End of data set

The following section describes the record formats for each logical record type on the archive volume:

TABLE 193. Record Format of Logical Record Types

Type	Offset	Length	Description
C — Control Record	+0	44	Data set name
	+44	6	Volume serial
	+50	2	Archive time
	+52	2	(reserved)
	+54	3	Date archived
	+57	3	Archive expiration date
	+60	2	DSORG, from format-1 DSCB
	+62	1	DEVCL
	+63	1	DS1SCALO, from format-1 DSCB
	+64	2	Primary space quantity
	+66	2	Number of tracks used
	+68	2	Space allocated
	+70	2	Secondary space allocation
	+72	1	Cylinder overflow (F2 + 52) 1... .... data set in compressed fmt .xxx xxxx Trks per ISAM cyl overflow
	+73	1	CTLNOTRK (F2 + 66)
1 — FMT-1 DSCB	This is an exact copy of the format-1 DSCB found in the VTOC of the original volume. Its format is described in the IBM publication “Data Areas”.		
D — Directory Record	+0	256	EXCP support. Directory block from PDS. (See System Data Areas.)
D — Directory Record	+0	8	Non-EXCP support. Member name
	+8	4	TTRF
	+12	v	User data
A — Directory Record	+0	8	Non-EXCP support only. Member name
	+8	4	TTRF
	+12	v	User data
R — User Data Blocks	+0	v	User data physical blocks or logical records
E — End Of Data Set	+0	10	’ END FILE ’
	+10	2	Directory count block
	+12	2	member count
	+14	1	’*’
	+15	2	DSNFLAGS - same as in DSNINDEX record
	+17	53	spaces
S — Split Record	+0	v	remainder of logical record

Archive Tape Format (Volume-level Processing)

Volume-level backup tapes (that is, those created by the VBACKUP command) contain all information necessary to restore an entire DASD volume. This includes both CA-Disk control information and physical blocks of the DASD volume that was backed up. All information is stored on the volume-level backup tapes as track image records.

TABLE 194. Archive Tape Format (Volume-level)

Type		Offset	Length	Description
M — Extent Record	Created during volume-level backup processing. They contain the data set name and extent information for each data set.	+0	44	Non-VSAM. DSNAME from format-1 DSCB.
		+44	2	DSORG from format-1 DSCB
		+46	v	Data set extents from 1 to 16
		+0	44	VSAM. Cluster name from VVDS reader.
		+44	2	DSORG from format-1 DSCB
		+46	44	Catalog name from VVDS reader
		+90	44	Component name from VVDS reader
		+134	v	Data set extents from 1 to 123
I — I/O Error Record	Created during volume-level backup processing. They are written to the tape when an I/O error occurs on the DASD volume being backed up.They contain the TTR and pertinent sense data.	+0	v	TTR and sense data for I/O errors

Archive Tape Format for Volume-level Backup Tapes

The following information indicates what records will be generated in the volume-level backup tape format.

TABLE 195. Tape Format for Volume-level Backups

Type	Description
C	control record
1	format-4 DSCB which imitates the format-1 DSCB
M	extent records
R	one or more physical data blocks
I	I/O error record
E	end of data set

CHAPTER 14

PARMLIB

This section describes the user-defined members of the CA-Disk parameter library (parmlib). You add members to parmlib when you want to modify or tailor the way a particular CA-Disk function works; for example, a list of data sets that you want to exempt from processing.

CA-Disk also keeps much of its own control information in the parmlib data set. These members are not described here, but in the appropriate section(s) of this manual instead. Appendix B "Contents of INSTALL Library" on page 502 also identifies the parmlib members.

Purpose

All functions of CA-Disk use various control parameters during their execution. These parameters provide many different kinds of information. Some indicate how CA-Disk is to operate, some define the format of reports, some define data sets for which CA-Disk is to skip processing, and others provide user-dependent information to CA-Disk functions.

These parameter lists are specified as members of a partitioned data set known as the parameter library, or parmlib for short. The parmlib implementation allows CA-Disk to better tailor its operation to each user's requirements. The contents of a parmlib member can easily be changed to reflect the desired options. Once they are changed, the next execution of CA-Disk will use them.

The library is ready for use as soon as it is loaded from tape during the installation process. The supplied members are regarded as parameter lists to be used for internal CA-Disk system functions and should not be changed. Parameter lists to contain user information are placed in new members of the library. The documentation for each function of the system explains what user-defined members from parmlib it uses and the purpose of each.

Most CA-Disk functions that require user-created members will indicate specific names to be assigned when the member is created. Some functions allow this name to be overridden by parameters on the CA-Disk commands. This allows you to create several different parmlib members for use by the same function. Then you specify the member to be used at execution time.

See "User PARMLIB Members" on page 479 for explanations of common CA-Disk parameter lists and the purpose of each. The documentation for each function of CA-Disk further defines the parmlib members used. More information is found there.

Updating Member Data

Since the CA-Disk parameter library is a partitioned data set, it can be updated like any other PDS. CA-Disk does not provide special facilities to update it. Most installations have on-line editors to change members of PDSs. The standard IBM-supplied utility for maintenance of PDS members is the IEBUPDTE utility. If you are unsure how to update this library, see your systems programmer.

Backup Considerations

Since the parameter library is a conventional PDS, it can be managed by the same means. Note, however, that if the parameter library is lost, CA-Disk restores cannot be done. Therefore it is advisable to occasionally take a backup copy of this data set using the IEBCOPY utility. Following is sample JCL to backup the CA-Disk PARMLIB with the IBM IEBCOPY utility. A copy of this JCL is in the CA-Disk installation library in member name PARMBKUP.

FIGURE 65. Sample JCL from Installation Member PARMBKUP

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT          .DMS.R900.INSTALL(PARMBKUP) - 01.06        Columns 00001 00072
 Command ===> _                                            Scroll ===> CSR
 000100 //JOBNAME  JOB (ACCT INFO)
 000200 //*
 000300 //*    JCL TO BACKUP PARMLIB
 000400 //*
 000500 //BACKUP   EXEC   PGM=IEBCOPY
 000600 //SYSPRINT DD     SYSOUT=*
 000700 //SYSUT1   DD     DISP=SHR,DSNAME=SAMS.DISK.PARMLIB
 000800 //SYSUT2   DD     DISP=(,CATLG),DSNAME=DISK.BACKUP.PARMLIB,UNIT=TAPE
 001000 //*
 001100 //*    JCL TO RESTORE PARMLIB FROM BACKUP TAPE
 001200 //*
 001300 //RESTORE  EXEC   PGM=IEBCOPY
 001400 //SYSPRINT DD     SYSOUT=*
 001500 //SYSUT1   DD     DISP=OLD,DSNAME=DISK.BACKUP.PARMLIB
 001600 //SYSUT2   DD     DISP=(,CATLG),DSNAME=SAMS.DISK.PARMLIB,
 001700 //                UNIT=DISK,SPACE=(CYL,(2,2))
 4 ■                       Aa   A TCPIP        R 4  C 15        17:49   5/28/98
```

PARMLIB Specifications

These are the syntax rules for entries in members of parmlib:

1. A single entry can be from one to seventy-one characters long and can be continued if longer.
2. An entry must be enclosed in quotes if it contains embedded blanks or is continued. Any entry can be enclosed in quotes.
3. To continue an entry, add a comma after the ending quote.
4. To have quotes within an entry enclosed in quotes, use two quotes.
5. To include a comment, add an asterisk in column one. For a comment on a command line, it should follow one or more blanks after the entry.

Examples

FIGURE 66. Sample Syntax for PARMLIB Entries

```
File  Edit  Transfer  Options  Connection  Macro  Window  Help

 File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
EDIT                    .PARMLIB(EXAMPLES) - 01.01                                  Columns 00001 00124
Command ===>                                                                         Scroll ===> CSR
=COLS> ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8----+----9----+----0----+----1----+----2----
000001 1234567890                                                  | AN ENTRY WITH NO EMBEDDED BLANKS
000002                                                             |
000003 'A SAMPLE ENTRY'                                            | AN ENTRY WITH EMBEDDED BLANKS
000004                                                             |
000005 'A SAMPLE ENTRY THAT HAS BEEN CONTINUED ',                  | A TWO LINE CONTINUED ENTRY
000006 'THROUGH TWO LINES'                                         |
000007                                                             |
000008 'A SAMPLE ENTRY THAT HAS BEEN CONTINUED ',                  | A THREE LINE CONTINUED ENTRY
000009 'THROUGH THREE ',                                           |
000010 'LINES'                                                     |
000011                                                             |
000012 'A SAMPLE ENTRY THAT HAS '' QUOTES '                        | AN ENTRY ENCLOSED IN QUOTES USING QUOTES
000013                                                             |
000014 12345'67890                                                 | AN ENTRY NOT ENCLOSED IN QUOTES USING QUOTES
000015                                                             |
000016 * THIS IS A COMMENT                                         | A COMMENT
000017                                                             |
000018 'A SAMPLE ENTRY FOLLOWED WITH COMMENT' COMMENTED HERE       | AN ENTRY FOLLOWED BY A COMMENT ON THE SAME LINE
000019                                                             |
000020 * THIS IS A COMMENT                                         | SEVERAL COMMENTS
000021 * THAT HAS MORE THAN ONE LINE                               |
****** ************************************************** Bottom of Data ***************************************************
                          Aa   B TCPIP        R 4  C 15        14:44  11/26/97
```

Special Rules for NAMEREFS

NAMEREFS is a member in parmlib that allows users to specify alternate names for CA-Disk tables. It is intended for systems that have used previous releases of CA-Disk that supported parmlib members with 10-character names. This was before parmlib was converted to a partitioned data set format. The member contains the name in the old 10-character format and the new character name to be used in its place. This allows CA-Disk users to continue to specify old format names in the CA-Disk commands and still be properly accessed in the CA-Disk parameter library.

If this member is not used and a 10-character name is used on a CA-Disk command, CA-Disk truncates the 10-character name to 8 characters and searches parmlib for the character name.

Each entry in NAMEREFS has this format:

```
'OLDTABLENM NEWMEMBR'
```

where:

OLDTABLENM - the old table name - maximum of 10 characters

NEWMEMBR - the new member name - maximum of 8 characters

Note: The entry must be enclosed in quotes because of the embedded blank between the two names.

Special Rules for POOLDEFS

The POOLDEFS member defines up to 150 pool names to be used by the archive/backup functions by associating a poolname with the name of the archive data set being created. If the data set is being created on tape, the pool name defines a tapepool that must exist within the CA-Disk files data set. If the data set is being created on disk (archival to disk), the pool name defines a diskpool that must exist as a member in the parmlib data set.

Each entry in POOLDEFS has the following format. Note that it must be enclosed in quotes because of the embedded blank between the two names.

```
'POOLNAME DATASETNAME'
```

where:

POOLNAME - a pool name — must be 8-characters in length

DATASETNAME - a data set name/pattern, 1- to 44-characters, that is to be mapped to the specified pool

PARMLIB Syntax Validation Utility

This utility validates the syntax of the specified members or all members in the parmlib data set. A copy of the execution JCL is in the installation library in member PARMVALD.

FIGURE 67. Sample JCL of Installation Member PARMVALD

```
File  Edit  Transfer  Options  Conrection  Macro  Window  Help

  File  Edit  Confirm  Menu  Utilities  Compilers  Test  Help
 ------------------------------------------------------------------------------
 EDIT          .DMS.R900.INSTALL(PARMVALD) - 01.09        Columns 00001 00072
 Command ===> ____________________________________________ Scroll ===> CSR
 ****** ***************************** Top of Data ******************************
 000100 //JOBNAME  JOB (ACCT INFO)
 000200 //VALIDATE EXEC PGM=ADSMI002,PARM=ADSTS152,REGION=2048K
 000300 //STEPLIB   DD DISP=SHR,DSN=SAMS.DISK.LOADLIB
 000400 //PARMLIB   DD DISP=SHR,DSN=SAMS.DISK.PARMLIB
 000500 //CMDPRINT  DD SYSOUT=A
 000600 //MSGPRINT  DD SYSOUT=A
 000700 //SYSPRINT  DD SYSOUT=A
 000800 //SYSUDUMP  DD SYSOUT=A
 000900 //ABNLDUMP  DD DUMMY
 001000 //SYSIN     DD *
 001100  VALIDATE MEMBERS=
 001200 /*
 ****** ***************************** Bottom of Data ***************************
 4 ▌                      Aa   A TCPIP       R 4  C 15       14:04   5/28/98
```

A list of from 1 to 50 member names to be validated for syntax rules.

Alternate DDname for User Tables

There are 4 TABLE= parameters used in CA-Disk:

1. The DSCL FIND command, described in “TABLE=” on page 50 in the User’s Guide
2. The DSCL SELECT command, described in “TABLE=” on page 52 in the User’s Guide
3. The DSCL EXCLUDE command, described in “TABLE=” on page 75 in the User’s Guide
4. The FMS SELECT command, described in “TABLE=” on page 117 in the User’s Guide

These tables are *user-defined,* and by default are stored as members in PARMLIB. In some environments where protecting PARMLIB from unauthorized use is normal practice, managing these tables can become a nuisance.

To alleviate this problem, an alternate DDname can be used for the sole purpose of storing user-defined tables. This is accomplished by specifying the system parameter TABLESDD with a value other than PARMLIB. Doing so allows CA-Disk to use a different DDname for retrieving these user-defined table entries. This allows your users to store and manage their own tables without the need for access to PARMLIB.

For details and an example of how to specify this system parameter, see “TABLESDDparmlib” on page 181.

User PARMLIB Members

Member Name: RETENTAU
Alternate Name: Specify on AUTABLE= parameter on command
Used By: RETAIN COMMAND - IMPLICIT ARCHIVAL

Description

The purpose of this member is to permit data set retention control where only data sets having one of a common set of data set name or volume prefixes are permitted to reside on a volume or class of volumes.

Each member entry must consist of a character volume serial and a 1- to 44-character data set name.

While the volume serial numbers do not have to be in any particular sequence, all entries for a given volume serial number or prefix must be contiguous in the member.

Format

```
'VVVVVVDDDD....'
```

Where:

VVVVVV - A 1- to 6-character volume serial number or a CA-Disk pattern value in positions 1 through 6.

DDDD - A 1- to 44-character data set name or CA-Disk pattern name in positions 7 through 50.

Example

The first sample entry indicates that all “SYS /” data sets are authorized only for volumes beginning with the three characters “SYS”. The second entry indicates that all “USERA.” data sets must exist on volumes beginning with “TSO”.

```
'SYS/ SYS./'
'TSO/ USERA./'
```

Billing Members in PARMLIB

Member Name: DSBCATEG
DSBRATEC
DSBUSERD
DSBEXCLU

Used By: ACCUM COMMAND - DASD BILLING FUNCTION
EXTEND COMMAND - DASD BILLING FUNCTION

Description

These members are used by the DASD Billing function of CA-Disk. Please turn to "Billing Rates" on page 402 in the User's Guide for a description of their use and format.

Billing User Code Definition Member

Member Name: USERCODT

Used By: ACCUM COMMAND - DASD BILLING FUNCTION
EXTEND COMMAND - DASD BILLING FUNCTION

Description

This member assigns user codes to data set names or prefixes. It is used by the DASD Billing function.

Entries are used to define DASD Billing user code values to data set names. Data set names can be coded as full names or prefix names. If a data set name is encountered and no entry exists for it in the member, a value of spaces will be assigned. To allow a default value to be used instead of spaces, specify the last member entry with the default value desired and a data set name value of a single slash character.

Format

```
'CCCCCCCCCCCCDDDD....'
```

Where:

CCCCCC - A 1- to 12-character user code to be assigned to data set names matching the following data set name or pattern in positions 1 through 12.

DDDD - A 1- to 44-character data set name or CA-Disk pattern name in positions 13 through 56.

Example

The following example shows three entries in a user code definition member. The first entry indicates that all data sets that begin with the characters "PROD" are assigned the user code "PRODUCTION". All data sets beginning with "TEST" are assigned the user code "DEVELOPMENT". All other data sets are assigned the user code "UNKNOWN".

```
'PRODUCTION PROD./'
'DEVELOPMENT TEST./'
'UNKNOWN /'
```

DASD Pool List

Member Name: DASDPOOL

Used By: RECOVER COMMAND - RECOVER FUNCTION
RESTORE COMMAND - RESTORE FUNCTION

Description

Entries are listed in this member to define one or more pools of DASD volumes that share a common role. This member is used by the restore function to obtain an alternate volume for allocation when the data set being restored cannot be placed on the target volume. It is also used when the POOL parameter is specified on the RESTORE command.

The DASD pool list is a single member in the parmlib data set. Each entry contains the name of the volume to be associated with a DASD pool and the pool name. Multiple volumes can be associated with the same pool name by having multiple entries with the same pool name and different volume serials. Multiple DASD pools can be generated by using different pool names on groups of entries. Note that all entries comprising a pool must be grouped together.

Up to 60 different DASD pool names can be defined. Sysparm DASDVMAX specifies the total number of volumes that can be included in the defined pools (sum of volumes in all different pools). This is also the maximum number of volumes that can be specified for a single pool name. The default value is 100; for example, one pool of 100 volumes, or two of 50, etc.

Format

```
'VVVVVVPPPPPPPP'
```

Where:

VVVVVV - The 6-character volume serial of a volume in a pool. If it is fewer than six characters in length, trailing blanks must be provided and the entry enclosed in quotes in position 1 through 6.

PPPPPPPP - The 1- to 8-character alphanumeric pool name. This name must be identical for all volume entries comprising a specific pool in positions 7 through 14.

Example

The following entries define two pools, a volume pool named RJEWORK and a volume pool named TSODATA.

```
'VOL003RJEWORK'
'VOL004RJEWORK'
'VOL006RJEWORK'
'PACK01TSODATA'
'PACK12TSODATA'
'PACK15TSODATA'
'PACK20TSODATA'
```

Note: To implement ARCHIVE to DISK, review the step-by-step instructions under the topic "Archiving to Disk: Requirements and Recommendations" on page 62.

Auto-Restore DSNAME Pool Member

Member Name: name specified by user in sysparm ARDSNPOL

Used By: AUTO-RESTORE DRIVEN FROM CATALOG MANAGEMENT

Description

Entries are listed in this member to define one or more data set pattern names that get associated with pools of DASD defined in the member DASDPOOL. This member is used by the auto-restore function to obtain a pool of volumes for allocation selection, rather than forcing the restore to a specific target volume.

The entries in this table relate either explicit or pattern data set names with a specific DASD pool to which to restore. All standard CA-Disk pattern-matching capabilities are supported for the data set names. When a data set is being restored, the entries in this table are searched from top to bottom to find a match. The first match that is found will have its pool name used. With this in mind, you should place the more specific entries at the start of the table.

If you always wanted to force a default pool to be used when no other matches are found, use the following entry as the last entry in the table:

```
'DFLTPOOL/ '
```

There is currently no limit to the number of entries permitted in this table. The only requirement is that all pool names referenced must exist in the member called DASDPOOL. The same pool name can be used in multiple entries in the table.

Format

```
'PPPPPPPPDDDDDD....DDD'
```

Where:

PPPPPPPP - The 8-character pool name defined in member DASDPOOL in positions 1 through 8.

DD...DD - The 1- to 44-character data set name pattern to compare against in positions 9 through 52.

Example

The following entries define several patterns to compare against. Note that the entries go from most specific to most generalized.

```
'POOL3350LABS.DMSOS.*.LOADLIB '
'POOL3380LABS.DMSOS.*.FILES '
'POOLWORKLABS.TEST/ '
'POOLPRODLABS.PROD/ '
'POOLWORKLABS.TJP/ '
'POOL3380LABS./ '
'POOL3350/ '
```

Auto-Restore Source Volume Pool Member

Member Name: name specified by user in sysparm ARVOLPOL

Used By: AUTO-RESTORE DRIVEN FROM CATALOG MANAGEMENT

Description

Entries are listed in this member to define one or more source volume pattern names that get associated with pools of DASD defined in the member DASDPOOL. This member is used by the auto-restore function to obtain a pool of volumes for allocation selection, rather than forcing the restore to a specific target volume.

The entries in this table relate either explicit or pattern source volume names with a specific DASD pool to which to restore. All standard CA-Disk pattern-matching capabilities are supported for the volume names. When a data set is being restored, the entries in this table are searched from top to bottom to find a match. The first match that is found will have its pool name used. With this in mind, you should place the more specific entries at the start of the table.

If you always wanted to force a default pool to be used when no other matches are found, use the following entry as the last entry in the table:

```
'DFLTPOOL/ '
```

There is currently no limit to the number of entries permitted in this table. The only requirement is that all pool names referenced must exist in the member called DASDPOOL. The same pool name can be used in multiple entries in the table.

Format

```
'PPPPPPPPVVVVVV'
```

Where:

PPPPPPPP - The 8-character pool name defined in member DASDPOOL in positions 1 through 8.

VVVVVV - The 1- to 6-character pattern to compare against the source volume of the data set being restored in positions 9 through 14.

Example

The following entries define several patterns to compare against. Note that the entries go from most specific to most generalized.

```
'POOL3350LABS5/'
'POOL3380LABS8/'
'POOLWORKWRK/ '
'POOLPRODPROD/ '
'POOLTEMP/ '
```

Exclusion Table for Data Compression

Member Name: Specified in sysparm DCEXCTBL_____

Used By: ARCHIVE AND BACKUP FUNCTION

Description

Place the data set names and patterns that are to be excluded from data compression during archive or backup into this member. A match causes data compression to be bypassed.

Format

```
'DDDD.....'
```

DDDD... - A 1- to 44-character data set name or CA-Disk pattern data set name. Located in positions 1 through 44.

Example

The following entries cause data sets with a prefix of “SYS1” and data sets with only three nodes and the third node is “PRINT” to be excluded from compression processing.

```
'SYS1/'
'*.*.PRINT'
```

Inclusion Table for Data Compression

Member Name: Specified in sysparm DCINCTBL_____

Used By: ARCHIVE AND BACKUP FUNCTION

Description

Place the data set names and patterns that are to be included for data compression during archive or backup into this member. A match causes the data set to be compressed.

Format

```
'DDDD.....'
```

Where:

DDDD..... - A 1- to 44-character data set name or CA-Disk pattern data set name. Located in positions 1 through 44.

Example

The following entries cause data sets with a prefix of “SYS1” and data sets with only three nodes and the third node is “PRINT” to be compressed.

```
'SYS1/'
'*.*.PRINT'
```

Exemption Lists - Generalized

Member Name: Specified with command parameters;
EXEMPT=
EXTABLE=
TABLE=

Used By: RELEASE COMMAND (IDLE SPACE RELEASE)
SCAN COMMAND (PDS COMPRESS)
EXCLUDE COMMAND (DSCL)
MOVE/COPY COMMANDS

Description

Place the data set names and patterns that are to be exempt from CA-Disk volume scan processing in this member of parmlib. When CA-Disk is searching DASD volumes for data sets to process, it will check to ensure that the data set name is not in this list before attempting processing for it. If the data set name is in this list, it will be exempted from processing.

Note: By default, these tables are stored in PARMLIB. However, you can store them in an alternative data set if you desire. For details, see "Alternate DDname for User Tables" on page 478.

Format

```
'DDDD.....'
```

Where:

DDDD..... - A 1- to 44-character data set name or CA-Disk pattern data set name. Located in positions 1 through 44.

Example

The following entries cause data sets beginning with the five characters "SYS./", and data sets with a third node level of "PRINT" to be excluded from CA-Disk processing.

```
'SYS /'
'*.*.PRINT'
```

Exemption List for Implicit Archive (Retain)

Member Name: RETENTXM

Alternate Name: Specify on EXTABLE= parm on command

Used By: RETAIN COMMAND - IMPLICIT ARCHIVAL

Description

Data sets that are to be exempted from retention control processing must have their names or a matching name prefix placed in this member. When EXEMPT is coded on the RETAIN command, this member is searched for a matching data set name for each data set being examined. A match causes no further processing; that is, the data set is exempt.

Member entries can be a combination of data set names and patterns with a length of 1 to 44 characters.

Note: A different member name can be used by specifying it in the EXTABLE= parameter. No exemption member is ever examined, however, unless the EXEMPT parameter is also present.

Format

```
'DDDD.....'
```

DDDD..... - A 1- to 44-character data set name or CA-Disk pattern data set name. Located in positions 1 through 44.

Example

The following entries cause data sets beginning with the five characters “SYS /” and data sets with a third node level of “PRINT” to be excluded from processing.

```
'SYS /'
'*.*.PRINT'
```

GDG DASD Generations List

Member Name:	GDGDASDG (for sequential migrate function) xxxxxxxx (specified on RETGDGNM sysparm for the retain function)
Used By:	SCAN COMMAND - SEQUENTIAL MIGRATE TO TAPE
	RETAIN COMMAND - IMPLICIT ARCHIVAL

Description

Used by the sequential migration to tape function, and the implicit archival function, this member is used to specify the number of generations of each GDG data set that are to be kept on disk.

For the sequential migrate to tape function, the older generations are copied to tape and recataloged. The entries in this member are retrieved and used only when the GENER= parameter is specified with the SCAN command. The member entries override the value specified on the GENER= parameter.

For the implicit archival function, the older generations are processed according to the options specified with the G selection type entry in the volspec table, with the member entries overriding the limit value specified in that same entry.

This data set name list is searched from top to bottom during CA-Disk processing. The first data set or pattern name in the list that matches the data set being processed will be used.

Format

```
'NNNNDDDD.........'
```

NNNN - A 4-digit generation count indicating the number of generations of a GDG data set to keep on disk. Located in positions 1 through 4.

DDDD.... - A 1- to 36-character GDG base name or pattern to which this version count applies. Do not include the GggggVvv as part of your pattern! Use all or a portion of the base name with an appended slash. Located in positions 5 through 40.

Example

The following sample entries indicate to keep the two current generations of all GDGs starting with PROJ.GDG.INDX1, and to keep three copies of those starting with DEPT75.GDG.INDX2. All other data sets will have one copy kept on DASD.

```
'0002PROJ.GDG.INDX1/'
'0003DEPT75.GDG.INDX2/'
'0001/'
```

Migration Exemption List

Member Name:	MIGRATXM
Alternate Name:	Specify on METABLE= parameter of command
Used By:	SCAN COMMAND - SEQUENTIAL MIGRATE TO TAPE

Description

Place the data set names and patterns that are to be exempt from the sequential migration to tape function in this member. Any data set that matches a specification in this list will be ignored by CA-Disk. Each entry consists only of a 1- to 44-character data set name or pattern.

Format

```
'DDDD.....'
```

Where:

DDDD..... - A 1- to 44-character data set name or CA-Disk pattern data set name. Located in positions 1 through 44.

Example

The following entries cause data sets beginning with the five characters “SYS /” and data sets with a third node level of “PRINT” to be excluded from sequential migration processing.

```
'SYS /'
'*.*.PRINT'
```

PDS Anomaly Table

Member Name:	Specify on sysparm PDSANMLY
Used By:	Archive, backup, restore and recover of a PDS

Description

This member is optionally used by the CA-Disk archive and restore processing for a PDS. It defines specific types of anomalies found in partitioned data sets, and what options to use when they are encountered.

See “Processing PDSs that Contain Anomalies” on page 75 for information about the format and use of this member.

Reblocking Member

Member Name: Specify on BLKSIZE= parm of MIGRATE command

Used By: MOVE/COPY COMMAND

Description

This member is optionally used by the Move/Copy function when the user indicates that data sets are to be reblocked as they are moved to new DASD volumes.

See "Reblocking" on page 363 in the User's Guide for information about the format and use of this member.

Retention Period By Prefix List

Member Name: RETENTPR

Alternate Name: Specify on RPTABLE= parameter of command

Used By: RETAIN COMMAND - IMPLICIT ARCHIVAL

Description

This member is used by retention control processing. When the R selection type is specified in the volspec entry, each data set being processed is matched against the names in this member to retrieve a retention period. This retention period value is added to the data set's creation date or last used date to yield an expiration date. If this date has been reached, the data set is disposed according to the disposal actions in the volspec entry. Also see the sysparm description for "RETPREFTc" on page 170.

The entries are loaded into memory on the first call and remain for the duration of retention control processing. The list is searched serially until an entry is found that corresponds to the passed volume serial number and data set name.

To have a retention period applied to a given data set name or pattern regardless of volume of residence, /xxxxx should be specified as the volume prefix.

Format

```
'NNNNVVVVVVDDDD....'
```

NNNN - The 4-digit retention period to be added to the creation or last used date of any data set with the name or pattern name given in the (d-d) field. Located in positions 1 through 4.

VVVVVV - The 1- to 6-character volume serial number of the volume on which the data set exists. This can be a CA-Disk pattern specification. Located in positions 6 through 10.

DDDD.... - The 1- to 44-character data set name or CA-Disk pattern data set name to which this entry applies. Located in positions 11 through 54.

Example

The first sample entry indicates that all "SYS /" data sets are to be kept for 365 days, no matter what volume they exist on. The second entry indicates that all data sets beginning with "TEST./" will be kept for only 15 days.

```
'0365/ SYS /'
'0015TSO/ TEST./'
```

Retention Control Volume Specification Table

Member Name: RETVOLSP
Alternate Name: Specify on VSTABLE= parameter on command
Used By: RETAIN COMMAND - IMPLICIT ARCHIVAL

Description

This member is used to indicate to CA-Disk implicit archival what data set selection criteria are in effect and what actions are to be taken against selected data sets. See the Retention Control section for a complete explanation of this member and how it is specified.

Security List for CA-Disk Data Set Protection

Member Name: SECURDSN
Used By: ALL CA-Disk functions

Description

This member is used to define data set names to be associated with CA-Disk resources that are to be protected. The member is currently used for the protection of subfiles within the CA-Disk files data set, and CA-Disk command names. To activate the CA-Disk features to protect these resources, specify sysparms SECURFIL and SECURCMD with a value of Y.

Each entry in this member describes a resource type to be protected, the name of the resource, and the data set name and volume to be associated with the resource.

For more information on how this member is used, see the topic "Resources Protected by the Security Interfaces" on page 314.

Note: If a resource is not defined in this member and security checking has been activated for the resource type, the resource is considered not to be protected and access to the resource is allowed.

Format

```
'CRR-RRVVVVVVDDDD....'
```

' - A single quote used as a delimiter. An * asterisk indicates a comment line. Located in position 1.

C - A single character indicator specifying the type of resource to which the entry applies. Located in position 2.

The following values are valid:

C — For CA-Disk command name.
F — For a subfile of the files data set.

RR-RR - A 1- to 15-character name of the resource to be protected (for example, a command or subfile name). This name can be specified with CA-Disk pattern matching notation. Located in positions 3 through 17.

VVVVVV - A 1- to 6-character volume serial to be passed to the security system with the data set name. (This field is required for Top Secret and RACF, and is optional for CA-ACF2.) Located in positions 18 through 23.

DDDD.... - A 1- to 44-character data set name known to the installation security system. Located in positions 24 through 67.

Examples

Following are examples of entries for this member.

Example 1 - The following entry protects CA-Disk command name ARCHIVE. Data set name "SAMS.DISK.ARCHIVE.COMMAND" will be interrogated whenever the user attempts to use the ARCHIVE command. The data set name exists on volume=SYSRES. If the user does not have OUTPUT authority for this data set name, the command will be rejected.

```
'CARCHIVE SYSRESSAMS.DISK.ARCHIVE.COMMAND'
```

Example 2 - The following entry protects CA-Disk subfile name DSNINDEX. Data Set name "SAMS.DISK.FILE.DSNINDEX" will be interrogated whenever the user attempts to use the subfile. The data set name exists on volume=SYSRES. If the user does not have appropriate authority to process the subfile (that is, to archive a data set the user must have UPDATE authority to the DSNINDEX subfile), processing will be terminated.

```
'FDSNINDEX SYSRESSAMS.DISK.FILE.DSNINDEX'
```

Selection Lists — Generalized

Member Name:	Specified with command parameters TABLE=
Used By:	SELECT COMMAND - DSCL
	FIND COMMAND - DSCL

Description

Place the data set names and patterns that are to be included in processing in this member of parmlib. When CA-Disk is searching DASD volumes for data sets to process, it will check to ensure that the data set name is in this list before attempting to process it. If the data set name is not in this list, it will be exempted from processing.

Note: If the intention of the table is for use with either the FIND or RECOVER command, then each data set name specified must include a complete HLQ (High Level Qualifier). For a further explanation, refer to the third example below.

Note: By default, these tables are stored in PARMLIB. However, you can store them in an alternative data set if you desire. For details, see "Alternate DDname for User Tables" on page 478.

Format

```
'DDDD.....'
```

DDDD..... - A 1- to 44-character data set name or CA-Disk pattern data set name. Located in positions 1 through 44.

Examples

The following entries cause data sets beginning with the five characters “SYS /” and data sets with a third node level of “PRINT” to be processed by CA-Disk.

```
'SYS/'
'*.*.PRINT'
'PROG.!PROD'
```

System Parameter List

Member Name: SYSPARMS

Used By: ALL CA-Disk FUNCTIONS

Description

The CA-Disk system parameter list is a user-specified list of options used by CA-Disk during its operation. They indicate how the user wants CA-Disk to operate in their environment. These parameters are explained completely in Table 24, “System Parameters,” on page 107.

Each entry in the SYSPARMS member specifies a separate option to CA-Disk. System parameters have 8-character names that identify them. Each entry contains the character name followed by the option specification. Option specification is variable depending on the definition of the option.

Format

```
'XXXXXXXXVVVV....'
```

XX..XX - System parameter name. Located in positions 1 through 8.

VVVV.... - System parameter information. Located in positions 9 through nn.

Example

The following entries define two system parameters. The first parameter indicates that the VSAM support is installed on the CA-Disk system. The second option indicates that default retention periods for implicit archival processing is 15 days.

```
'VSAMSUPPY'
'RETRETPD0015'
```

TSO/ISPF USERID Authorization List

Member Name: TSOUSERI

Used By: ALL CA-Disk TSO FUNCTIONS

ALL CA-Disk ISPF FUNCTIONS

Description

This member contains the user IDs of those TSO users who are to be given the ability to archive and restore data sets of any name, not just those prefixed with their user ID. CA-Disk system defaults allow all users this privilege. This member is used if either sysparm TSOUSRIDY or SPFUSRIDY is present, restricting users to accessing only those data sets prefixed with their user ID.

Format

```
'UUUUUUU'
```

UUUUUUU - The 1- to 7-character TSO user ID of the TSO users authorized to process any data set with CA-Disk ISPF or TSO support. Located in positions 1 through 7.

Example

The following entries indicate that users “USERA” and “PGMMGRS” are authorized to process any data sets.

```
'USERA'
'PGMMGRS'
```

User-Specified Condition Codes

Member Name: CCSET

Used By: ALL CA-Disk FUNCTIONS

Description

The purpose of this member is to provide the user with the capability to override or not override the condition code generated by CA-Disk. If the user chooses to override the CA-Disk-generated condition code, one of four possible alternative condition codes can be selected and assigned per message number. The first entry in this member is required. The format is as follows:

```
$FLG FORMAT
```

Format

```
'$FLGXyz'
   or
'nnnnbxy'
```

Where:

$FLG	$FLG		
X	BLANK		
y	BATCH FLAG	B	display message under BATCH
		BLANK	do not display message under BATCH
z	TSO FLAG	T	display message under TSO
		BLANK	do not display message under BATCH

or

nnnn	4-digit CA-Disk message code. Located in positions 1 through 4.			
b	blank, located in position 5.			
x	O	override	Use CA-Disk condition code. Located in position 6.	
	N	no override		
y	override condition codes (located in position 7):	G	0	good
		I	4	informational message
		R	8	resource error
		E	12	error
		C	16	command error

Example

The following entry indicates that the user wishes to override the CA-Disk-generated condition code for message 3179 with a user-specified condition code of 16.

```
'3179 OC'
```

CHAPTER 15

JCL DD Statements

This section provides an explanation of all DD statements within the PROCS. All DD statements (except STEPLIB) are arranged alphabetically. This allows overriding DD statements to be used much more easily, since all you need to remember is to provide them in alphabetical order.

TABLE 196. Symbolic Parameters and Recommendations

Symbolic PARMS	Recommended Defaults	Explanations
P	SAMS.DISK.Rnn.INSTALL	Library for control statements
Q	SAMS.DISK.Rnn	Qualifier for data sets
S	*	Sysout class
W	SYSDA	Workfile default unit name

Where Rnn represents the release number that is installed. For example, Release 1.1 or 9.9 would be represented as R11 or R99, respectively.

Note: These default values are recommended, but can be different due to your local naming conventions. Please examine the actual procedures as installed and being used at your location and consult your local CA-Disk support personnel if you need assistance.

DD Statement Explanations

//AMSRFILE

A temporary sequential data set used by the MOVE function. Critical events are logged as they occur in VSAM data set move processing. This log is used to recover the cluster in the event CA-Disk abends when the data set is being moved.

//ARCHIVEC

This dd statement should be provided only when duplicate backup copies of the archive tapes are desired. It can be either omitted or dummied out if duplicate tapes are to be suppressed.

DCB attributes are automatically copied from the ARCHIVE0 dd statement. It can also be dynamically allocated based upon the values of sysparms ARCCTYPE and ARCCNAME.

//ARCHIVES

This dd statement is used by CA-Disk to allocate a tape drive when read access of the archives is necessary. The RESTORE and RECOVER functions and REBUILD access the CA-Disk archives for read only.

//ARCHIVE0

The output medium upon which archived data sets are to be stored is referenced by this dd statement. It can be provided in the JCL, or it can also be dynamically allocated based upon the values of sysparms ARC0TYPE and ARC0NAME. If the expiration date is not supplied by the label parameter or if its value is zero, the highest expiration date from all of the data sets to be written on it will be placed in the label as the expiration date.

//ARCHIVC1 ... ARCHIVC5

Output devices for the duplicate copies created by the MERGE function, which distributes the archived data sets based upon their expiration dates.

//ARCHIVE1 ... ARCHIVE5

Output devices for the primary copies created by the MERGE function, which distributes the archived data sets based upon their expiration dates.

//ARCHOUT

A sequential data set dynamically allocated to receive the DSCL Archive commands generated with the DASD Threshold Manager.

//COMMANDS

A temporary work file used by the RESTORE function to order the explicit RESTORE requests. This is done such that the archive tape will be mounted only once, and processed only in the forward direction.

//CMDPRINT

A transaction listing of the input command stream and any command processing messages.

//COMPWORK

A temporary work file used by the PDS Compress function. It holds the PDS member data during a SCRATCH and REALLOCATE of the data set.

//COPYFILE
A data set containing a list of CA-Disk archive volumes that are to be copied.

//DATEWORK
A record is written in this data set for each sequential data set to be migrated. The records are subsequently retrieved in descending expiration date order.

//DISPHOLD
A temporary data set used to suspend data set disposition until a threshold value can be verified. This allows certain CA-Disk functions to operate in a “fail safe” mode.

//DISPWORK
A disposition record is written in this data set for subsequent reference in determining the recatalog and scratch action after successful tape creation.

//DSNEXPDT
This file is REQUIRED (live or simulate mode) only if:

- Data sets that satisfy the selection criteria specify ARCHIVE as one of the disposal actions and
- No expiration date is provided on the //ARCHIVE0 dd statement.

The file is used to determine the highest expiration date from all the data sets being archived. This date is then assigned as the expiration date for the entire archive volume.

The file is used much like an internally generated deferred request. All disposal records for any archives to be done are temporarily stored here. The highest expiration date is found, the archive volume is opened, and the disposal records are then processed to cause the actual archiving of the data sets.

//DSNIIN
This dd statement is used as a SORTIN data set.

//DSNINDEX
This data set is used to contain the sorted ARCHIVE index records used for MERGE input.

//DSNIOUT
This dd statement is used as a SORTOUT data set.

//DSNIWK01 ... WK03
These dd statements are used as SORT work files.

//DSCBS
This dd statement is used as a SORTIN data set. It contains the format-1 DSCBs selected by the billing function.

//FILEIN

This dd statement is used as a SORTIN data set.

//FILEOUT
This dd statement is used as a SORTOUT data set.

//FILES
This dd statement allocates the CA-Disk files data set. It contains subfiles for the DSNINDEX, ARCHVOLS and other files.

//FILEWK01 ... WK03
These dd statements are used as SORT work files.

//FILPRINT
This dd statement defines a SYSOUT data set CA-Disk will use to produce a files status report.

//INDEXREC
A temporary work file used by the RECOVER function to order the DSNINDEX records. This is done such that an ARCHIVE tape will be mounted only once, and processed only in the forward direction.

//MIGRATE
Data sets are migrated to this device from DASD.

//MIGRATEC
When the COPY option has been specified to obtain duplicate tapes, this dd statement is required. The DCB attributes, including expiration date for this data set, are obtained from the MIGRATE dd statement.

//MFILES
This dd statement allocates one or more CA-Disk files data sets for autorestore, restore, and deferred restore processing. Each files data set allocated in the concatenation is searched to find the first DSNINDEX record for a data set to be restored. The FILES dd statement will not be used if the MFILES dd statement is allocated.

//MOVPRINT
A formatted report of data sets processed for the DSCL MOVE function.

//MSGPRINT
Execution time messages are directed to this data set.

//MTFTIME - A formatted report of module performance statistics, including Elapsed time and CPU time.

//PARMLIB
This dd statement defines the CA-Disk parameter library. It contains internal system parameter lists as well as any user-defined parameters.

//PARMLIBD (TSO/ISPF only)

This statement defines the CA-Disk parameter library. It is an alternate ddname that can only be used in an online (TSO/ISPF) environment if the ddname PARMLIB is used by another product. For more information, turn to the topic “Step 1. Identify Files, Parmlib, and Activation Codes” on page 18 in the Installation Guide.

//RELPRINT
A formatted report of data sets processed for the DSCL RELEASE function.

//REPORTS
This ddname is used to reference the data set that receives the detail records in any report generating step, and the data set containing the sorted records in the print step.

//RESTART
This data set is required by the Extend function of DASD Billing. It is used to clear the billing file and for proper recovery after an abend.

//SEQFILES
This data set contains records from a FILES unload and is used for the FILES reload.

//SMFDATA
This ddname is used in the SMFRPT procedure to reference a sequential data set containing SMF type 14 and 15 records. These records are used to produce the “Utilization by Date and Time” report.

//SMSPRINT
CA-Disk generates diagnostic messages explaining the processing done for the CA-Disk function as it pertains to IBM's DFSMS. These messages are written out to this data set.

//STEPCAT
The STEPCAT dd statement should be used only to override the normal catalog search techniques.

//SYSPARMS
The SYSPARMS dd statement can be used to supply overriding sysparm values at execution time.

//SYSPRINT
A formatted listing of all actions taken during the run are directed to this data set.

//TAPEPULL
This data set contains a copy of the generated “tape pull” list and restricts CA-Disk to only those tapes.

//VSAM
DD statement required for VSAM cluster reference.

//VSAMKEY
DD statement used to reference a VSAM index component when it resides on a different volume from the cluster's data component.

//VSAMRVOL
DD statement used to reference a volume that contains a VSAM recoverable catalog required to successfully define a VSAM cluster.

APPENDIX A

DSCB Appendage Fields

Allocation information regarding individual data sets that is not available in the format-1 DSCB is maintained as an appendage to the DSCB, which is passed to all non-VSAM control processing modules.

Format-1 Fields

The fields computed for the format-1 DSCB are described below. A binary format is used for all data.

TABLE 197. Format-1 DSCB Appendage Fields

Offset			
DEC	**HEX**	**Length**	**Description**
0	(00)	140	Data Set Control Block - DSCB
140	(8C)	8	mbbcchhr of DSCB (reports only)
148	(94)	2	Tracks Allocated
150	(96)	2	Tracks Idle
152	(98)	2	PDS Directory Blocks used (a value of x’7FFF’indicates it has not been calculated yet)
154	(9A)	2	PDS Directory Blocks empty
156	(9C)	2	PDS Member count (a value of x’7FFF’ indicates it has not been calculated yet)
158	(9E)	2	PDS Member average directory entry length (will be zero if not calculated yet)
160	(A0)	2	PDS Member directory capacity
166	(A6)	2	Tracks in Primary Allocation
168	(A8)	2	Extent 1 tracks
170	(AA)	2	Extent 2 tracks
172	(AC)	2	Extent 3 tracks
174	(AE)	2	Extent 4 tracks
176	(B0)	2	Extent 5 tracks
178	(B2)	2	Extent 6 tracks

TABLE 197. Format-1 DSCB Appendage Fields

Offset			
DEC	**HEX**	**Length**	**Description**
180	(B4)	2	Extent 7 tracks
182	(B6)	2	Extent 8 tracks
184	(B8)	2	Extent 9 tracks
186	(BA)	2	Extent 10 tracks
188	(BC)	2	Extent 11 tracks
190	(BE)	2	Extent 12 tracks
192	(C0)	2	Extent 13 tracks
194	(C2)	2	Extent 14 tracks
196	(C4)	2	Extent 15 tracks
198	(C6)	2	Extent 16 tracks
200	(C8)	2	PDS Member Directory entries available
202	(CA)	2	Primary allocation by type
204	(CC)	2	Number of Extents
206	(CE)	2	PDS - number ALIAS entries in directory
208	(D0)	2	PDS - dir entries with user TTR pointers
210	(D2)	2	PDS - dir entries with note lists
212	(D4)	1	ISAM - Number of tracks per cyl overflow
213	(D5)	1	ISAM - Number of tracks for highest index
214	(D6)	2	PDS - Max directory entry length
216	(D8)	2	PDS - Min directory entry length

Format-4 Fields

Allocation information that has been accumulated from the processing of all types of DSCBs is available as an appendage to the format-4 DSCB, which is passed as one of the parameters to the report assemble modules. The fields that are described below are carried in a binary format unless otherwise indicated.

TABLE 198. Format-4 DSCB Appendage Fields

Offset			
DEC	**HEX**	**Length**	**Description**
0	(00)	140	Format-4 DSCB - VTOC (volser in pos 1-6)
140	(8C)	8	mbbcchhr of DSCB
148	(94)	6	Volume serial
154	(9A)	4	Unit address
158	(9E)	2	Device class
160	(A0)	1	UCBSTAT

TABLE 198. Format-4 DSCB Appendage Fields

Offset			
DEC	**HEX**	**Length**	**Description**
161	(A1)	1	UCBSTAB
164	(A4)	4	Date - Julian (xxyydddf)
168	(A8)	4	Time - (xhhmmssf)
172	(AC)	2	DSCB count - blank
174	(AE)	2	DSCB count - F1
176	(B0)	2	DSCB count - F2
178	(B2)	2	DSCB count - F3
180	(B4)	2	DSCB count - F4
182	(B6)	2	DSCB count - F5
184	(B8)	2	DSCB count - F6
186	(BA)	2	DSORG count - IS
188	(BC)	2	DSORG count - PS
190	(BE)	2	DSORG count - DA
192	(C0)	2	DSORG count - PO
194	(C2)	2	DSORG count - AM
196	(C4)	2	DSORG count - other
198	(C6)	4	Category count - Device total
202	(CA)	4	Category count - allocated
206	(CE)	4	Category count - free space
210	(D2)	4	Category count - idle
214	(D6)	4	Category count - lost
218	(DA)	4	Free space extent - largest
222	(DE)	4	Free space extent - second-largest
226	(E2)	4	Free space extent - third-largest
230	(E6)	2	DSCBs in VTOC
232	(E8)	2	DSCBs allocated
234	(EA)	2	Free space extent counter
236	(EC)	2	Free space extent total
238	(EE)	2	blank DSCB total
240	(F0)	2	Data set allocation - 0 tracks
242	(F2)	2	Data set allocation - 1 to 50 tracks
244	(F4)	2	Data set allocation - 51 to 125 tracks
246	(F6)	2	Data set allocation - 126 to 300 tracks
248	(F8)	2	Data set allocation - over 300 tracks

APPENDIX B

Contents of INSTALL Library

The following procedures are created by executing member IEBUPDATE against the member PROCS in the PROCLIB data set on the distribution tape:

TABLE 199. List of PROCS in CA-Disk Proclib

Member	Description
ARCHIVE	Explicit archive
BILLING	DASD billing (Select and Accumulate)
COMPRES	PDS compression
CONFIGR	Volume configurator (DASD-to-DASD Migration)
DEFVOLS	Generate “tape pull list” for deferred restores
DERASE	Batch delete of deferred archive/restore cmds
DMS	DSCL invoked function
DMSAR	Started task for auto-restores
DMSGTF	Run a GTF trace of an application
DMSLDSD	List CA-Disk-generated RACF profiles
DMSLRAC	List alternate RACF profile x-ref
DMSPOOL	Tape pool maintenance
DMSPROF	CA-Disk RACF profile maintenance utility
EXTEND	DASD billing (EXTEND Totals and Clear)
FMS	RECOVER function (DSCL-like)
IXCATLG	Recatalog data sets in the archives
IXMAINT	Index maintenance
IXUPDATE	Index update utilities
LISTD	List archive indexes
LISTREQ	List deferred archive and restore requests
MERGCOPY	Create copy of new merge archvols (post-step)
MERGE	Archive merging
MIGRATE	Migrate sequential data sets to tape

TABLE 199. List of PROCS in CA-Disk Proclib

Member	Description
PDS2SEQ	Create sequential data set from PDS members
PROCS	IEBUPDTE stream containing all JCL PROCS
RACFCHK1	RACF synchronization utility
REBUILD	Rebuild index entries from tape
RECOVER	Full volume recovery
RELOAD	Reload files data set
REORG	Reorganize/resequence the files data set
REPORT	General DASD reports
RESTART	Restart after REORG shutdown
RESTORE	Explicit restore
RETAIN	Retention control (auto archival and backup)
RLSE	Release idle (over-allocated) space
SMFRPT	SMF-based report
THRSHMGR	Volume Threshold Manager Utility
UNLOAD	Unload files data set
VSAMBILL	VSAM DASD billing (select and accumulate)

The following is a list of members included with the PARMLIB data set on the distribution tape:

TABLE 200. List of PARMLIB Members and Descriptions

Function	Member	Description
Message	DMSMSGES	CA-Disk Messages
System Parameters	PARMEDIT	Sysparm defaults/validation
Files data set	FDSAMPLE	Sample Definition for All Device Types

TABLE 200. List of PARMLIB Members and Descriptions

Function	Member	Description
Reports	REPORTNM	Reportname (Control Modules for each Report)
	RPTNALLO	RPT=ALLOCS
	RPTNATTR	RPT=ATTRBS
	RPTNDIST	RPT=DISTRS
	RPTNDSIN	RPT=DSINDX
	RPTNDSUT	RPT=DSUTIL
	RPTNADSU	RPT=ADSUTI
	RPTNFREE	RPT=FREESP
	RPTNISAM	RPT=ISAMDS
	RPTNLAYO	RPT=LAYOUT
	RPTNMAPP	RPT=MAPPER
	RPTNMEMB	RPT=MEMBER
	RPTNMVDI	RPT=MVDICT
	RPTNAMVD	RPT=AMVDIC
	RPTNBMVD	RPT=BMVDIC
	RPTNPDST	RPT=PDSTAT
	RPTNSVDI	RPT=SVDICT
	RPTNDAYW	SMF Report by day and time
DASD Billing Report	RPTNDEVD	Summary By Device Type
	RPTNDEVY	Summary By Device Type (if sysparm DSBALTRT”1”)
	RPTNVOLD	Summary By Volume
	RPTNVOLY	Summary By Volume (if sysparm DSBALTRT”1”)
	RPTNDSND	Detail By Data Set Name
	RPTNDSNY	Detail By Data Set Name (if sysparm DSBALTRT”1”)
	RPTNODSD	Detail By Dsn Owner
	RPTNOWND	Summary By Owner
Merge Report	RPTNDSNM	Data Sets Merged Forward
	RPTNVOLM	Merged Volumes Status
Deferred Disposition Report	RPTNDEFD	Report From Retain Defer
PDS Compress Report	RPTNCOMP	PDS Compress Results
Archive/Retain Sorted Report	ARCHDSNV	Data Set Name Sequence
	ARCHVOLD	Volume / Data Set Name Sequence
Sequential Migration Sorted Report	MIGRDSNV	Data Set Name Sequence
	MIGRVOLD	Volume / Data Set Name Sequence
RACF Profile Report	RPTNRFDN	Data Set Name To CA-Disk-saved Profile Name XREF
	RPTNRFPF	CA-Disk-Saved Profile Name To Data Set Name XREF

TABLE 200. List of PARMLIB Members and Descriptions

Function	Member	Description
Sample Parmlib Members	SAMPARMS	Sample entries for mbr=SYSPARMS
	SAMPCSET	Sample entries for mbr=CCSET
	SAMPDBRC	Sample entries for data base names for DBRC
	SAMPDEFS	Sample entries for mbr=POOLDEFS
	SAMPPDSA	Sample entries for mbr=PDSANMLY
	SAMPREFS	Sample entries for mbr=NAMEREFS
	SAMPSPEC	Sample entries for volspec table for RETAIN
	SAMPSPFO	Sample entries for ISPF options
	SAMPZDMS	Sample entries for authorized CA-Disk Parmlibs
ISPF Functions	FMT1DICT	Format-1 dictionary (online reporting)
	FMT1FLDS	Format-1 fields (online reporting)
	FMT4DICT	Format-4 dictionary (online reporting)
	FMT4FLDS	Format-4 fields (online reporting)
	MODEL	Model commands (online reporting)
	SPFMENU	Menu options and descriptions
Source Code	ADSARnnn	Auto-restore
	ADSDMnnn	General system
	ADSDSnnn	Data security
	ADSFHnnn	DASD EXCP
	ADSMEnnn	Message editors
	ADSMInnn	Module Intercept/Transfer (not distributed)
	ADSSPnnn	ISPF support
	ADSTHnnn	Tape handling
	ADSTSnnn	Table support
	ADSUTnnn	Utility
	ADSVCnnn	Volume Configurator
	ADSVSnnn	VSAM

APPENDIX C

ENQ/DEQ/Reserve Usage

Integrity processing within CA-Disk is performed in two distinct ways. The first covers Data Sets, Index Files, Auto-Restore, Tape, and Cartridges.

FIGURE 68. Enqueue/Dequeue Reserve for Data Set Processing

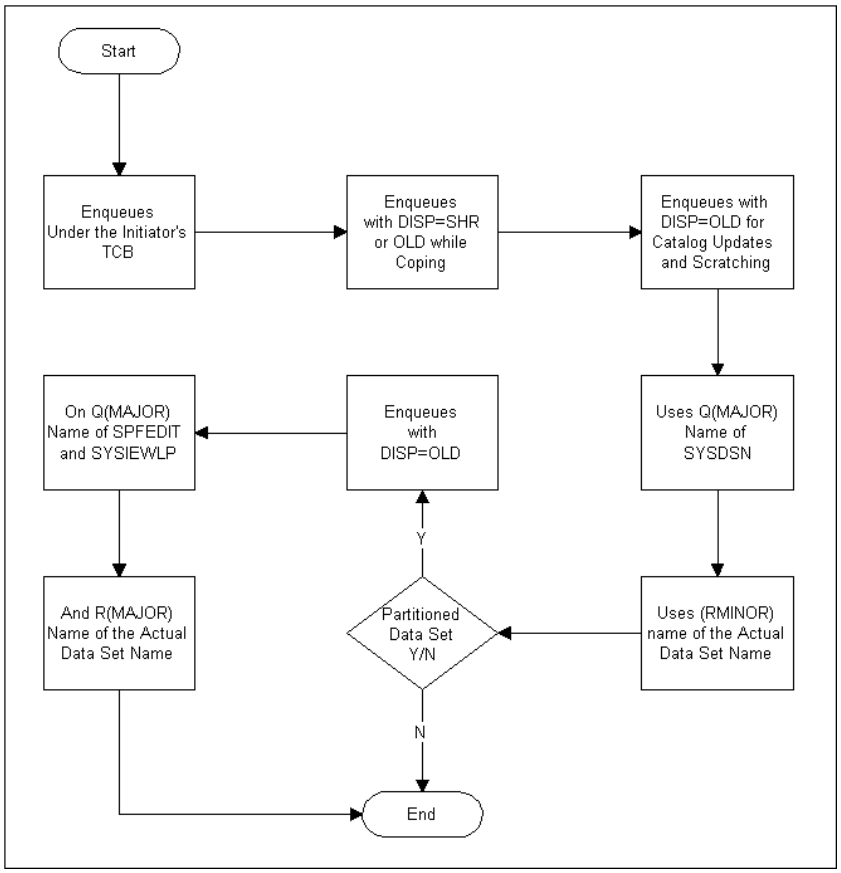


FIGURE 69. Enqueue/Dequeue Reserve for Protection of the SUBFILES

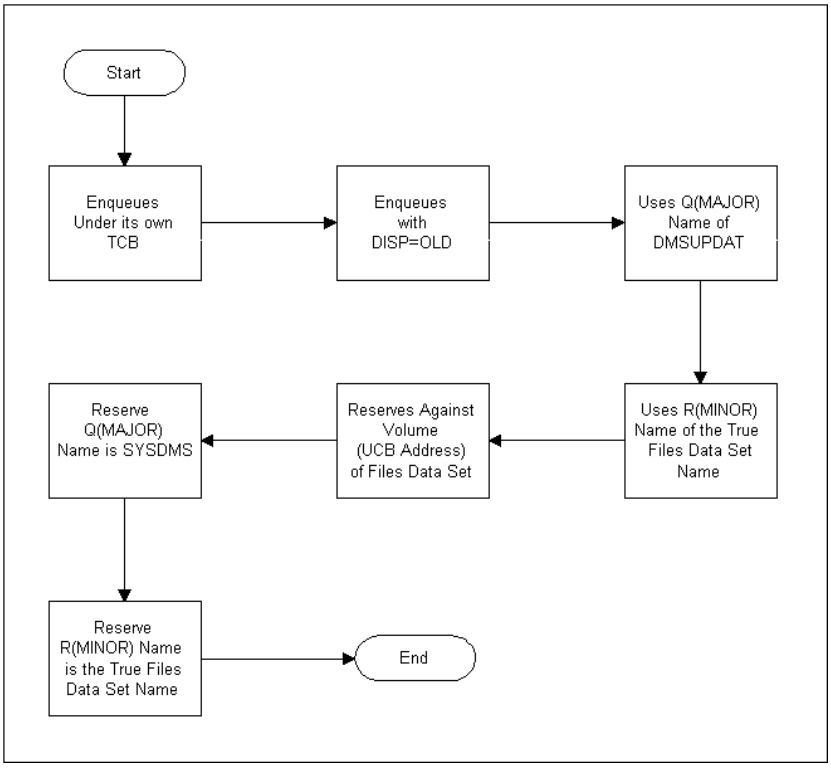


FIGURE 70. Enqueue/Dequeue Reserve for Auto-Restores

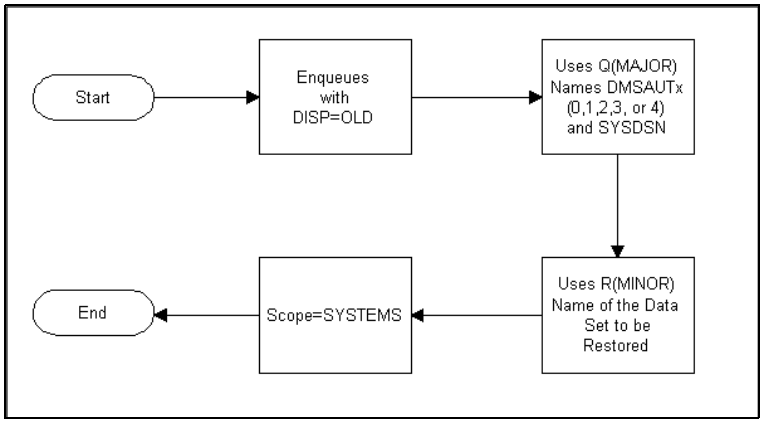
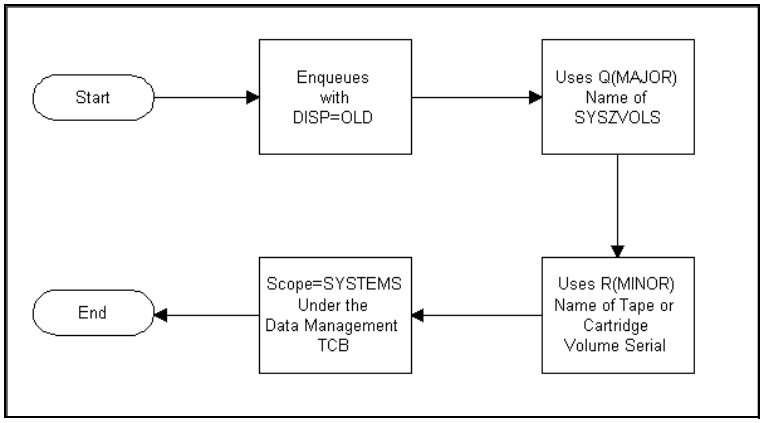
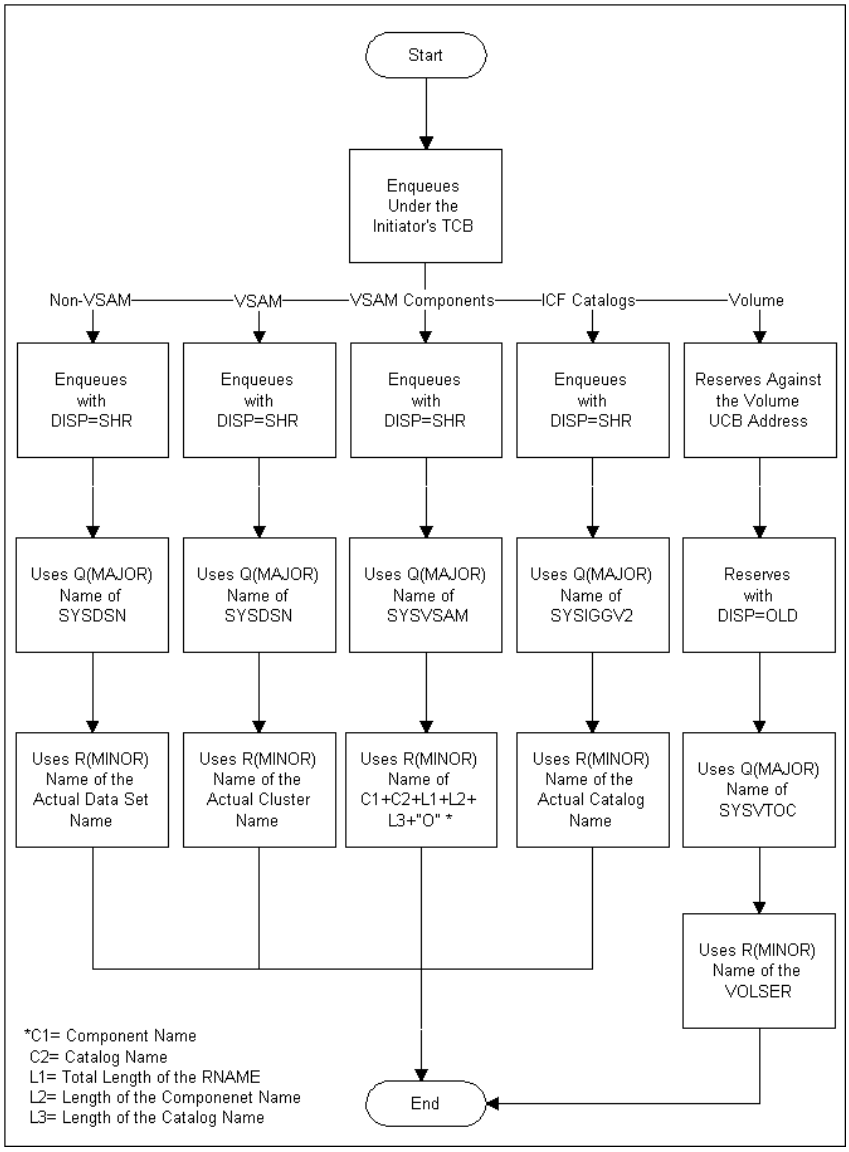


FIGURE 71. Enqueue/Dequeue Reserve for Tapes and Cartridges



The other way CA-Disk performs integrity is for VBACKUP and VRECOVER, which is illustrated below.

FIGURE 68. Enqueue/Dequeue Reserve for VBACKUP and VRECOVER



Index

Numerics

3420 50
3480/3490 51

A

abends
  PDS error handling 76
  tape status 74
ACF2FORCn 122
ACF2SUPPn 122

ADPCTOTHnnn 122
ADPCTPDSnnn 123
ADSUT028 266
ADTRACKSnnnn 123
allocation considerations 40
ALLOCPRIy 123
ALLOEXITmmmmmmmm 123
ALLOVBLKn 123
ALOCATEXmmmmmmmm 123
ALOPOSEXmmmmmmmm 123
ALTTRKASy 123
ARBACKUPy 124
ARC0NAMEnnn 128
ARC0TYPEtape 128
ARCATOPTu 124
ARCBLKSIn 124
ARCCNAMEnnn 124
ARCCPYDVnnnn 124
ARCCTYPEtape 125
ARCDSKBZbbbbb 125
ARCDSNEXmmmmmmmm 125
ARCDSORGn 125
ARCEMPTYy 125
ARCEODEXmmmmmmmm 125
ARCFDEFRn 125
ARCHCMDS subfile record format 458
ARCHDSNVOL record format 442
archive

abends 74
archiving the archives 72
archiving to disk requirements 62
assigning expiration dates 62
blocksizes for tapes 42
cataloging the tapes 44
data compression 57
expiration dates 45
extract utility 284
fix utility 287
integrity 73
media used 60
medium used 59
multiple jobs running concurrently 44
multivolume tape datasets 43
naming the archives 61
overwriting valid tapes 73
PDS error handling 76
producing duplicate tapes concurrently 42
protection of the disk archives 75
SAMS:Compress 58
security 73
simulate mode 44
space allocation when archiving to disk 389
system parameters 55
tape analysis utility 281
tape format 473
tape organization 468
tape protection 73
tape recovery utility 296
ARCHIVEXmmmmmmmm 125
ARCHPACKy 125
ARCHSORTudv 126
ARCHVOLDSN record format 444
ARCHVOLS
record format of the subfile 459
repair utility 289
ARCMODELn 126
ARCMULTIn 126
ARCNOSTDn 126
ARCPLENQe 126
ARCRETPDnnnnn 126
ARCSEPC0y 126
ARCTNAMEd 127
ARDDNEXC_ddname 128
ARDIAGNM_jobname 128
ARDSNPOLxxxxxxxx 128

ARESCRDTn 128
ARESPREXmmmmmmmm 128
ARESTENQy 128
ARESUNICxxxxxxxx 128
ARESUNInxxxxxxxx 128
ARESUNITxxxxxxxx 128
ARJOBEXC_jobname 129
ARPOOLEXmmmmmmmm 129
ARSECUREn 129
ARTAPEOKy 130
ARTASKNMdmsar 130
ARVOLPOLxxxxxxxx 130
assembly of SAMS:Disk modules 299
authority level translation tables 327, 335
auto-restore
  DSNAME pool member 482
  security 317
  source volume pool member 483

B

backup
  abends 74
  assigning expiration dates 62
  blocksizes for tapes 42
  cataloging the tapes 44
  data compression 57
  expiration dates 45
  integrity 73
  media used 60
  medium used 59
  multiple jobs running concurrently 44
  multivolume tape datasets 43
  naming the archive 61
  overwriting valid tapes 73
  PDS error handling 76
  protection of the disk archives 75
  RACF discrete profiles (RACFBKUP) 345
  SAMS:Compress 58
  security 73
  simulate mode 44
  system parameters 55
block splits 21
blocksizes for archive and backup tapes 42
bytes that fit on an output track 384

C

calling customer support 270

capacity
- considerations of the files dataset 21
- estimating the requirements of the files dataset 20
- of the files dataset 20

CARTCALCn 130

CARTFEETnnnn 131

CARTPRCTnnn 131

cartridge switching considerations 49

cataloging the tape archives 44

CATBKDELn 132

CATBKNAMnnnnnnnnnnnnnnnnnnn 132

CATBKSPC500 132

CATBKUNTunitname 132

CATLOCWAnnnnnnn 132

CATPURGEn 132

CCDRESETn 133

CCINITEXmmmmmmmm 133

CCSET 492

CNVATCYLn 133

coding for an interactive environment 301

command
- ADD (tapepool processing) 48
- CHECK (RACF) 348
- CREATE (RACF) 357
- DELETE (RACF) 357
- DELETE (tapepool processing) 49
- DISPLAY (RACF) 356
- ERASE (tapepool processing) 49
- FRECOVER (files data logging) 29
- LIST (tapepool processing) 48
- RENAME (RACF) 358

common file handler interface 33

compression flow 386

condition codes
- CCSET member of PARMLIB 492

considerations
- allocation type considerations 40
- backing up the PARMLIB dataset 475
- capacity of the files dataset 21
- cartridge length 49
- DASD threshold manager 98
- data compression 57
- I/O buffering 56

memory 56
PDS error handling 76
performance of the files dataset 21
SAMS:Compress 58
tape length 49
tapepool processing 45
threshold manager 98
control block structure 308
customer support 270

D

DARCHIEXmmmmmmmm 133
DARCKEEPxxx 133
DASD threshold manager 91
DASDPMAXnnn 133
DASDPOOL 481
DASDSPCB subfile record format 460
DASDVMAXnnnn 133
data compression 57
data compression techniques 387
DATECUTO75 133
DBDSNCFDx 134
DBDSNINXn 134
DBRCDBSZ20000 134
DBRCICSZ80000 134
DBRCPREXnnnnnnnn 134
DBRCRLIBnnn 134
DBRCRLSEnnn 135
DBRCSMUT3400 135
DBRCSUPPn 135
DBRCVCUT3380 135
DCAVGCMPnn 135
DCCMPEX0ADSUT371 135
DCCMPEX1ADSUT410 135
DCCMPEXnmmmmmmmm 135
DCCOMPTCn 135
DCDATACPn 136
DCDCPEX0ADSUT372 136
DCDCPEX1ADSUT411 136
DCDCPEXnmmmmmmmm 136
DCDSNDEXmmmmmmmm 136
DCEXCTBLxxxxxxxx 136
DCINCTBLxxxxxxxx 136
DCRTSTATn 136

DD statement explanations 495
DD statements 295
DDNAMES 40
decompression flow 386
DEFARCEXmmmmmmmm 136
DELETEOKn 137
DERASEEXmmmmmmmm 137
DFLTPOOLxxxxxxxx 137
DIAG0001n 137
DIAG0002n 137
DIAGALOCn 137
DIAGAUTHn 137
DIAGDCTCn 137
DIAGENQDn 137
DIAGENQFn 137
DIAGENQVn 137
DIAGEXCPn 137
DIAGLSTRn 137
DIAGNOSEn 137
diagnostics
  DD statements 295
  DMS diagnostic area 276
  tools 272
DIAGRECDn 138
DIAGSHRKn 138
DIAGSTAEn 138
DIAGSTMPy 138
DIAGTHRSn 138
DIAGVSRLn 138
DINXUFEXmmmmmmmm 138
discrete profile handling 344
DISKPROTy 138
DLVOLMSGy 138
DMSPARMS subfile record format 461
DMSPOOLS subfile record format 461
DRESARCHy 138
DRESKEEPxxx 138
DRESTOEXmmmmmmmm 139
DSBALTRTn 139
DSBCATSZxxxxx 139
DSBSELVSn 139
DSBTRKSZt 139
DSBUNITSxxx 139

DSBUSREXmmmmmmmm 139
DSBVOLCTnnnn 140
DSCBCVEXmmmmmmmm 140
DSCBJBNMxxx 140
DSCBLMODnnnp 140
DSCBLUSDnnnp 140
DSCBOPCDnnnh 140
DSCBSVMDnnn 140
DSCBUSRDxxx 140
DSCLMSGSy 140
DSCLOWRCn 140
DSCLRPLYn 141
DSCLSCEXmmmmmmmm 141
DSDISPEXmmmmmmmm 141
DSNDELEXmmmmmmmm 141
DSNDLPEXmmmmmmmm 141
DSNDUPVLn 141
DSNINDEX
  record format of the subfile 462
DSNSELREn 141
DSSEIOTRn 142
DSSMSGDDxxxxxxxx 142
DSUCATLGy 142
DSUTILCF0 142
DSUTILSQd 142
DYNALDSNn 142
dynamic allocation 40
DYNCUNITxxxxxxxx 142
DYNEXPDTE99365 143
DYNnUNITxxxxxxxx 144
DYNTUNITxxxxxxxx 143
DYNUKEEPn 144
DYNVREQDn 144

E

ENQADDTLn 145
ENQDSSIMn 145
ENQGDGBYn 145
ENQJES3Tdyn 145
ENQMSGESn 146
ENQSTPOKn 146
enqueue reserves 506
ENQVOLWTnnn 146

ENQWAITDn 146
exclusion table for data compression 484
exemption lists - generalized 485
EXPARSOPy 146
EXPASCOPs 146
EXPDTUSEn 146
expiration dates
  in archive datasets 62
  in tapes 45
extract utility 284

F

Fast Path 275
FASTCOMPy 147
files dataset
  activating the logging feature 26
  allocation control record 14
  block splits 21
  capacity 20
  capacity considerations 21
  common file handler interface 33
  control blocks 14
  estimating capacity requirements 20
  file control block 15
  formatting of 17
  general information 13
  integrity 13
  logging 26
  logical blocks 14
  logical data block 15
  logical index block 15
  logical records 14
  management 17
  merging files datasets 32
  naming the log datasets 27
  overflow blocks 21
  performance 56
  performance considerations 21
  physical blocksize 14
  RELOAD utility 25
  Reload utility 22
  REORG utility 26
  Reorg utility 22
  restrictions 16
  Sample Recovery report 30
  Status report 18
  Structure 14

subfile dump utility 30
subfiles 15
system parameters used for logging 54
unload utility 22
utilities 22
FILESCNT12 147
FILESPAC1500 147
FILEUNITnnnnnnnn 147
FILOGNAMxxxxxxxxxxxxxxxxxxxxxxxxxx 147
fix utility 287
FLYCOUNT2 147
FMSSAFEXmmmmmmmm 147
FMSSPREXmmmmmmmm 148
FORCPOOLnnnnnnnn 147
Format-1 / Format-4 DSCB Appendage Fields 362
Format-1 Fields 499
Format-4 Fields 500
formatting messages 267

G

GDG DASD Generation List 486
GDGCATEXmmmmmmmm 148
general information
DASD threshold manager 91
files dataset 13
IDRC support 43
parmlib dataset 474
tailoring SAMS:Disk 52
threshold manager 91
user exits 201

H

HONORTTRn 148

I

I/O buffering requirements 56
IBM’s Magstar support 41
ICHEKAEXmmmmmmmm 148
ICHEKPEXmmmmmmmm 148
idle space calculation for VSAM 384
IDRC
overview 43
IDRCRATE00 148
IDRCSIZE005 148
IDRCSTATv 149

IDRCUSEDx 149
IGNLSTARy 149
implementing
  Masstor M860 81
inclusion table for data compression 484
installation library 502
integrity in the archives 73
integrity within the files dataset 13
intercept control stack 272
intercept tracing 276
interfaces
  common file handler 33
  execuring without a security interface 318
  module call interface 302
  PARMLIB interface 302
  security 314
  system parameter interface 304
IOERRLIMnnnnn 149
IOMAXRECnnnnnn 150
IOTRACKSxx 150
ISAM report Logic 378
ISAMRECOc 150
ISPF
  adding user functions 305
  modifying the SAMS:Disk functions 307
  report modules 87
  tapepool updates 47
  Userid authorization list 492
IXCTLGEXmmmmmmmm 151
IXMCLEANy 151
IXMDUPDTn 151
IXMQTIME0000 151
IXMREVEXmmmmmmmm 151
IXMVOLONn 151
IXUSRNOTn 152

L

link-edit of SAMS:Disk modules 299
LISTDMEXmmmmmmmm 152
LISTDVEXmmmmmmmm 152
LISTREEXmmmmmmmm 152
LISTSORTn 152
LISTVTOCn 152
load library status utility 292

logging activity of the files dataset 26
LUSEDATEn 152

M

M860 support 41, 81
macros 350
Magstar support 41
managing the files dataset 17
Masstor support 41, 81
media used in archive and backup 60
medium used in archive and backup 59
memory requirement 56
MERCnNAMnnnnnnnnnnnnnnnnnnnnn 152
MERCnTYPtape 153
MERDSNEXmmmmmmmm 153
MERFASTSy 153
merging files datasets 32
MERPnNAMnnnnnnnnnnnnnnnnnnnnn 153
MERPnTYPtape 153
MERRELTMnn 153
MERREPLYn 153
MERVOLEXmmmmmmmm 153
message formatting 267
MI Fast Path 275
MIGBYPASn 154
MIGCATLGy 154
MIGCREDTn 154
MIGCTGDGn 154
MIGDISEXmmmmmmmm 154
MIGDSMAXnnnn 154
MIGDSORGn 154
MIGEXPDTn 154
MIGFXPDTy 154
MIGGDGEXmmmmmmmm 154
MIGMODELn 154
MIGNOEXPn 154
MIGNXPDTy 154
MIGPSCOPn 154
MIGPSDENddddd 155
migration exemption list 487
MIGRDSNVOL record format 446
MIGRECAT subfile record format 465
MIGREXITmmmmmmmm 155

MIGRSORTndv 155
MIGRVOLDSN record format 448
MIGYRCHGn 155
modifying SAMS:Disk ISPF functions 307
module call interface 302
module intercept control stack 272
multivolume tape datasets 43
MVDCATLGy 155

N

NAMEREFS 476
naming the archive and backup tapes 61
NEWPATHNy 156
NOBLKMSGn 156

O

OFFLDASDn 156
ONCCFAILcancel 156
overflow blocks 21
overview
  DASD threshold manager 91
  IDRC support 43
  ISAM report Logic 378
  parmlib dataset 474
  report processing 360
  security 312
  space allocation for archival to disk 389
  threshold manger 91

P

PARMEDIT 304
PARMLIB
  auto-restore DSNAME pool member 482
  auto-restore source volume pool member 483
  backup considerations 475
  billing members 480
  CCSET member 492
  DASDPOOL 481
  exclusion table for data compression 484
  exemption lists 485
  GDG DASD Generation List 486
  inclusion table for data compression 484
  ISPF Userid authorization list 492
  migration exemption list 487
  NAMEREFS 476
  parameter library support 302
  PDS Anomaly Table 487

POOLDEFS 477
purpose of 474
reblocking member 488
retention period list 488
RETVOLSP member 489
SECURDSN member 489
selection lists 490
specifications 476
syntax validation utility 478
SYSPARMS member 491
TSOUSERI 492
updating member data 475
user members 479
validation 304
PASSNEWNn 156
password-indicated dataset support 318
PASSWORDn 156
PDS
anomaly code descriptions 77
compression space calculations 394
compression threshold calculations 392
error handling 76
PARMLIB Anomaly Table 487
PDSANMLYmmmmmmmm 157
PDSELOADn 157
PDSENQSHn 157
PDSEXSIMy 157
PDSMAXPR00000 157
PDSMAXSE99999 157
PDSMINDB00000 158
PDSMINPR00000 158
PDSMINSE00000 158
PDSRRMAXnnnnn 158
PDSSETBSy 158
performance tips 55
files dataset 56
files dataset consideratins 21
I/O buffering 56
memory 56
PDS compress 55
report processing 55
POINTABSy 158
POOLDEFS 477
PREALLOCn 158
PRIALLOCn 159
PRTSORTFy 159

PSUDEVTPn 159

R

RAADDAEXmmmmmmmm 160
RAADDPEXmmmmmmmm 160
RACF
authority level translation tables 335
backing up discrete profiles 345
changing the value of RACFDVOL 359
CHECK command 348
conversion of SAMS:Disk RACF profile encoding 353
CREATE command 357
DELETE command 357
discrete profile handling 344
DISPLAY command 356
encode profile (Format 2) 352
format of SAMS:Disk saved profiles 352
Installation exits 351
macros 350
RACFDVOL 359
reconsiling with SAMS:Disk 346
RENAME command 358
restoring datasets that have no profiles 346
user exits 350
using RACF macros 350
VSAM datasets 345
RACFALLOa 160
RACFALWZy 160
RACFBITSy 161
RACFBKUPn 161
RACFDAEXmmmmmmmm 161
RACFDPEXmmmmmmmm 161
RACFDVL2dmsxxx 161
RACFDVOLvvvvvv 161
RACFEAEXmmmmmmmm 162
RACFENCD subfile record format 465
RACFENTY2 162
RACFEPEXmmmmmmmm 162
RACFMDSNxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 162
RACFMODLn 162
RACFMVOLvvvvvv 162
RACFNEWNn 162
RACFPDSWn 162
RACFPREDn 163
RACFPROCy 163

RACFSEQMn 163
RACFSEQTy 163
RACFSMEXmmmmmmmm 163
RACFSUPPn 163
RACFTESTn 163
RACFUSIDxxxxxx 163
RACFVCAVn 164
RACGDGVLvvvvvv 164
RACHKAEXmmmmmmmm 164
RACHKPEXmmmmmmmm 164
RADEFAEXmmmmmmmm 164
RADEFPEXmmmmmmmm 164
RADELAEXmmmmmmmm 165
RADELPEXmmmmmmmm 165
RADELSVCnnn 165
REBLIMITnnnnn 165
REBLKBITn 165
reblocking member in PARMLIB 488
REBYPASSy 166
RECATPSUr 166
RECATVOLvvvvvv 166
RECCHCATn 166
reconsiling SAMS:Disk and RACF records 346
record format
  ARCHCMDS subfile 458
  ARCHDSNVOL 442
  ARCHVOLDSN 444
  ARCHVOLS subfile 459
  DASDSPCB subfile 460
  DMSPARMS subfile 461
  DMSPOOLS subfile 461
  DSNINDEX subfile 462
  files dataset Logging Records 467
  MIGRDSNVOL 446
  MIGRECAT subfile 465
  MIGRVOLDSN 448
  RACFENCD subfile 465
  RESTCMDS subfile 466
  RETEXCLD subfile 467
  RPTDSNEX 456
  RPTNALLOCS 398
  RPTNAMVDIC 400
  RPTNATTRBS 402
  RPTNBMVDIC 404
  RPTNDAYWEK 419

RPTNDEVDSB 406
RPTNDISTRS 408
RPTNDSINDX 415
RPTNDSNDSB 410
RPTNDSNX 412
RPTNDSUTIL 417
RPTNFREESP 421
RPTNLAYOUT 423
RPTNMAPPER 425
RPTNMEMBER 427
RPTNMVDICT 429
RPTNPDSTAT 431
RPTNPO1REF 433
RPTNPO2REF 433
RPTNPOCCHH 433
RPTNPODUMP 433
RPTNPOTTRD 433
RPTNPOXREF 433
RPTNRACFDN 434
RPTNRACFPF 436
RPTNSVDICT 438
RPTNVOLDSB 440
RPTNVOLX 414
RPTPLST1 454
RPTPLST2 451
RPTSTATS 450
VSAMCLDATA 452
RECPRIEXmmmmmmmm 166
REDEVDEFn 166
Redwood support 41
release
idle space calculation for VSAM 384
relinking TSO and ISPF control modules 301
repair archvols utility 289
reports
formatting tables 362
ISPF conversion modules 87
logic diagram 373
overview 360
performance 55
processing - ISAM 375
REPORTNM Table 361
RESAFTEXmmmmmmmm 166
RESCHCATy 166
RESCOPYTn 167
RESCRCATn 167
RESCRNEXmmmmmmmm 167

RESIXDLBn 167
RESIXRPDnn 167
RESPRIEXmmmmmmmm 168
RESRETPDnnnnn 169
RESTCMDS subfile record format 466
RESTCTLGy 169
restoring RACF-indicated datasets 346
restrictions
  archiving the archives 72
  files dataset 16
  ISAM Space 377
RETAFTEXmmmmmmmm 169
RETDSNCDccasc 169
RETDSNCLccas 169
RETDSNDFccddd 169
RETDSNLUccasc 169
retention control volume specification table 489
retention period by prefix list 488
RETENTPR 488
RETEXCLD subfile record format 467
RETEXCLDn 169
RETEXCMDn 170
RETGDGEXmmmmmmmm 170
RETGDGNMxxxxxxxx 170
RETMULTIn 170
RETNODSPyy 170
RETNOMIGyy 170
RETPREFTc 170
RETPRIEXmmmmmmmm 170
RETRETPDnnnnn 170
RETVOLSP 489
RLSEAFEXmmmmmmmm 171
RLSEDIAGn 171
RLSEMAXSnnnnn 171
RLSEMINSnnnnn 171
RLSESECRa 171
RLSESYS1n 171
RLSEUNMVy 171
RPTCLSELxx...x 171
RPTDSBEXmmmmmmmm 171
RPTDSNEXmmmmmmmm 172
RPTDSUEXmmmmmmmm 172
RPTLINES 172

RPTMVDEXmmmmmmmm 172
RPTNMTSZxxx 172
RSCHGBITn 172
RSUPPRESn 172

S

SAMSYSNMaim 172
SCRASYS1n 172
SCRTCHEXmmmmmmmm 172
SECARVOLarcive 173
SECFRVOLn 173
SECURAEXmmmmmmmm 173
SECURCMDn 173
SECURDRSaaaaaaaa 173
SECURDSN 489
SECURFILn 173
security
  archive disk protection 75
  archive tape protection 73
  auto-restore function 317
  executing without a security interface 318
  overwriting valid tapes 73
  RACF 335
  resources not protected by SAMS:Disk 316
  resources protected by the SAMS:Disk base system 313
  resources protected by the security interfaces 314
  statement of direction 311
  top secret 327
security in the archives 73
SECURLIMnnnn 173
SECURLOGy 173
SECURPEXmmmmmmmm 174
SECURRPTn 174
SECURVOLy 174
SECUSUPPn 174
SELECTAMn 175
selection lists 490
SELECTREy 175
sending information to the technical support center 272
SHRNKFDTdmsosfdt 175
SHRNKSYSxxxx 175
SMFCDATDnnn 175
SMFCTIMDnnn 175
SMFDFLAGnnn 175

SMFDSNMDnnn 175
SMFJDATDnnn 175
SMFJTIMDnnn 175
SMFTYP14nnn 176
SMFTYP15nnn 176
SMFVLSRDnnn 176
SMSACSDFn 176
SMSALLOCn 176
SMSBYBKVn 177
SMSCACHEn 177
SMSCPROCy 177
SMSDPROCy 177
SMSFORCEn 177
SMSGBRST99 177
SMSIPROCy 178
SMSLOWRCn 178
SMSMCBYPn 178
SMSSGBYPn 178
SMSSTGADddd 178
SMSUPDATy 178
SORTPADDy 179
space allocation algorithm 390
SPACECEXmmmmmmmm 179
SPACEPRInnn 179
SPACESECnnn 179
SPFAUNITnnnnnnnn 179
SPFDSNAMnn....nn 179
SPFENQQNnnnnnnnn 179
SPFENQTYr 179
SPFENQWTnn 179
SPFHDGEXmmmmmmmm 180
SPFHILVNnnnnnnnn 180
SPFORDEXmmmmmmmm 180
SPFORSEXmmmmmmmm 180
SPFORSPCnnnn 180
SPFORSPEnnnn 180
SPFRPTDTy 180
SPFSYSINnnnnnnnn 180
SPFUSRIDn 180
statement of security direction 311
storagetek support 41
subfile dump utility 30

subfiles of the files dataset 15
support
  IBM's Magstar 41
  Masstor M860 41, 81
  storagetek redwood 41
SVCNODMSnnn 180
SVDCATLGy 180
SYSOUTEXmmmmmmmm 180
SYSPARMOn 181
SYSPARMS 491
SYSPARMSn 181
system parameters
  archive and backup 55
  definitions 122
  files dataset logging 54
  for PDS error handling 76
  lines per page 54
  list of sysparms available 107
  messages 54
  overriding 106
  reports 53
  restore/recover 55
  separtor pages 54
  sysparms interface 304
  SYSPARMS member of PARMLIB 491
  validation 304

T

TABLESDDparmlib 181
TAOPPOEXmmmmmmmm 182
TAOPPREXmmmmmmmm 182
TAPCTLEXmmmmmmmm 182
tape
  3420 devices 50
  3480/3490 devices 51
  analysis utility 281
  archive tape organization 468
  format 51, 473
  recovery utility 296
  status after abends 74
  switching considerations 49
TAPEDLEXmmmmmmmm 182
TAPEFEETnnnn 183
TAPEFSCRn 183
tapepool processing
  ADD command 48

considerations 45
DELETE command 49
ERASE command 49
LIST command 48
update utility 48
updates from ISPF 47
TAPEPOOLn 183
TAPEPROTn 183
TAPEPRTRy 183
TAPEVVERy 183
TAPMFEETnnnn 184
technical support 270
TERMCLERy 184
TESTMSGSn 184
threshold manager 91
TMSCTLEXmmmmmmmm 184
top security
authority level translation tables 327
TOPSALWZy 184
TOPSFORCn 184
TOPSSUPPn 184
TPMGMTEXmmmmmmmm 184
TPOOLCHGn 184
TPOOLSEQs 185
tracing 276
track dump utility 294
TRKOVCHGn 185
TSCHKAEXmmmmmmmm 185
TSCHKPEXmmmmmmmm 185
TSODACATu 185
TSODARESy 185
TSODASCRs 185
TSOTULMTnn 185
TSOTUNITnnnnnnnn 185
TSOUSERI 492
TSOUSRIDn 185
TSOVOLOFn 186
TTRULMOVn 186

U

UCBALLOCxx 186
UCC1BYPSn 186
UCC1EXPRy 186
UCC1EXTNnnnn 186

UCC1SCRTn 186
UCC6CPDSn 186
UNCATARCn 187
UNCATDELn 187
UNCATFDRn 187
UNCATPSUn 187
UNLOADEXmmmmmmmm 188
UNMOVEARn 188
UNMOVEVCn 188
USEDSSIO 188
user exits
  archive analysis 284
  DASD threshold manager 99
  descriptions 206
  fix utility 289
  general information 201
  list of available user exits 203
  RACF 350
  threshold manager 99
  using previous releases 202
USERSENQmmmmmmmm 189
USERVENQmmmmmmmm 189
USRCDLENxx 189
USRCODEXmmmmmmmm 189
utilities
  archive extract utility 284
  archive fix utility 287
  archive tape analysis 281
  archive tape recovery utility 296
  extract utility 284
  fix utility 287
  load library status utility 292
  of the files dataset 22
  PARMLIB syntax validation 478
  RELOAD 22
  RELOAD utility 25
  REORG 22
  REORG utility 26
  REPARCH utility 289
  subfile dump utility 30
  tape recovery utility 296
  tapepool update utility 48
  track dump utility 294
  UNLOAD 22
  UNLOAD utility 22

V

VBRCHCATn 189
VBRIVSAMn 190
VCBYPASSn 190
VCCHGBITn 190
VCCOPYEXmmmmmmmm 190
VCDMIGEXmmmmmmmm 190
VCUDSORGn 190
VOLSELREn 190
VSACCESSc 191
VSACESEXmmmmmmmm 191
VSAIXDEFy 191
VSALLOEXmmmmmmmm 191
VSALOSHRn 192
VSAM
  idle space calculation 384
  new name algorithm 385
  RACF 345
VSAMCLDATA record format 452
VSAMSUPPn 192
VSAPPENDn 192
VSARCAIXn 192
VSARCFMTl 192
VSATTREXmmmmmmmm 193
VSATTRIBn 193
VSBIXCATxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 193
VSBIXPSWnnnnnnnn 193
VSBLDINXy 193
VSCATMAXnnnn 193
VSCATPTNy 194
VSCIACATy 194
VSCIADETn 194
VSCOMPNMn 194
VSCRIMAXnnnn 194
VSDATEFTn 194
VSDATSUFxxxxxxxx 194
VSDEFCATy 195
VSDSPACEn 195
VSINXSUFxxxxxxxx 195
VSMAFTEXmmmmmmmm 196
VSMPRIEXmmmmmmmm 196
VSMPURGEy 196
VSMSGEMPn 196

VSMSGTOTn 196
VSMSGWRNn 196
VSMSPWEXmmmmmmmm 196
VSMSTATSn 196
VSNEWNEXmmmmmmmm 196
VSNOERASn 196
VSNOSCROn 197
VSONLINEn 197
VSPREDEFn 198
VSRESCLDn 198
VSSEQCNTnnn 198
VSSEQERRy 198
VSSPEEDOy 198
VSSPUSEDy 198
VSSTUMPLnnnnnn 199
VSSYSVERn 199
VSUNOPENn 199
VSUSEBLKn 199
VSUSEMODn 199
VSVERIFYn 199
VTOC
  ENTRIES- format 1, 2 and 3 DSCBs 377
VTOCREADe 200

W

WTORESRLy 200

X

XCPARCEXmmmmmmmm 200
XCPDSNEXmmmmmmmm 200

List of Figures

FIGURE 1. Sample Files Data Set STATUS Report 18
FIGURE 2. Sample UNLOAD JCL and Command 23
FIGURE 3. Sample RELOAD JCL and Command 25
FIGURE 4. Files Data Set LOGGING Generated DSNAME 27
FIGURE 5. Sample FRECOVER Report 30
FIGURE 6. Sample SUBFILE Dump Utility 31
FIGURE 7. Call to Retrieve DSNINDEX Record 33
FIGURE 8. Predicting Output Device Capabilities 50
FIGURE 9. Generated 22 Character DSNAME 61
FIGURE 10. Sample Source of POOLDEFS Member in PARMLIB 64
FIGURE 11. PDS Anomaly Table 76
FIGURE 12. Sample JCL Using the TABLESDD Statement 182
FIGURE 13. Sample Report produced by the MTFTIME DD Statement 278
FIGURE 14. Sample JCL for the Archive Tape Analysis Utility 281
FIGURE 15. Sample Archive Tape Verification Report 283
FIGURE 16. Sample JCL for the EXTRACT Utility 286
FIGURE 17. Sample JCL for the Archive Fix Utility 288
FIGURE 18. Sample JCL for the Repair Archvols Utility (REPARCH) 290
FIGURE 19. Sample JCL for the Load Library Status Utility 292
FIGURE 20. Sample Load Library Status Report 293
FIGURE 21. Sample JCL for the Archive Tape Recovery Utility 296
FIGURE 22. Sample Archive Tape Recovery Utility Report 298
FIGURE 23. Sample Control Block Structure 310
FIGURE 24. Multi-level approach to security 313
FIGURE 25. RACF Checking Logic Summary Flowchart 349
FIGURE 26. Racf Encode Profile Generated Dataset Name 352
FIGURE 27. Sample DMSLRAC Report 355
FIGURE 28. MEMBER=RPTNOMVD in the PARMLIB data set 369

FIGURE 29. MEMBER=RPTNMVDI in the PARMLIB data set 370
FIGURE 30. Report Logic Diagram 373
FIGURE 31. Record Collection portion of Report Logic 374
FIGURE 32. Detail Records Grouped Together Example 375
FIGURE 33. General flow of data compression 386
FIGURE 34. Data Compression Technique 0 Flowchart 388
FIGURE 35. Sample Parmlib Member RPTNALLO 399
FIGURE 36. Sample Parmlib Member RPTNAMVD 401
FIGURE 37. Sample Parmlib Member RPTNATTR 403
FIGURE 38. Sample Parmlib Member RPTNBMVD 405
FIGURE 39. Sample Parmlib Member RPTNDEVD 407
FIGURE 40. Sample Parmlib Member RPTNDIST 409
FIGURE 41. Sample Parmlib Member RPTNDSND 411
FIGURE 42. Sample Parmlib Member RPTNDSNX 412
FIGURE 43. Sample Parmlib Member RPTNVOLX 414
FIGURE 44. Sample Parmlib Member RPTNDSIN 416
FIGURE 45. Sample Parmlib Member RPTNDSUT 418
FIGURE 46. Sample Parmlib Member RPTNDAYW 420
FIGURE 47. Sample Parmlib Member RPTNFREE 422
FIGURE 48. Sample Parmlib Member RPTNLAYO 424
FIGURE 49. Sample Parmlib Member RPTNMAPP 426
FIGURE 50. Sample Parmlib Member RPTNMEMB 428
FIGURE 51. Sample Parmlib Member RPTNMVDI 430
FIGURE 52. Sample Parmlib Member RPTNPDST 432
FIGURE 53. Sample Parmlib Member RPTNRFDN 435
FIGURE 54. Sample Parmlib Member RPTNRFPF 437
FIGURE 55. Sample Parmlib Member RPTNSVDI 439
FIGURE 56. Sample Parmlib Member RPTNVOLD 441
FIGURE 57. Sample Parmlib Member ARCHDSNV 443
FIGURE 58. Sample Parmlib Member ARCHVOLD 445
FIGURE 59. Sample Parmlib Member MIGRDSNV 447
FIGURE 60. Sample Parmlib Member MIGRVOLD 449
FIGURE 61. Sample Parmlib Member RPTSTATS 450
FIGURE 62. Sample Parmlib Member RPTTPUL 451
FIGURE 63. Sample Parmlib Member RPTDAPUL 455
FIGURE 64. Sample Parmlib Member RPTDSNEX 457
FIGURE 65. Sample JCL from Installation Member PARMBKUP 475
FIGURE 66. Sample Syntax for PARMLIB Entries 476
FIGURE 67. Sample JCL of Installation Member PARMVALD 478
FIGURE 68. Enqueue/Dequeue Reserve for Data Set Processing 506
FIGURE 69. Enqueue/Dequeue Reserve for Protection of the SUBFILES 507
FIGURE 70. Enqueue/Dequeue Reserve for Auto-Restores 507
FIGURE 71. Enqueue/Dequeue Reserve for Tapes and Cartridges 508
FIGURE 68. Enqueue/Dequeue Reserve for VBACKUP and VRECOVER 509

List of Tables

TABLE 1. FILEDEFN Record Format 16
TABLE 2. Possible Values for the KEY Parameter of the Subfile Dump Utility 31
TABLE 3. DSUTIL Report Sorting Options 53
TABLE 4. Message Suppression Sysparms 54
TABLE 5. Files Data Set LOGGING Activation Sysparms 54
TABLE 6. Command ARCHIVE/BACKUP Sysparms 55
TABLE 7. Common RESTORE/RECOVER Sysparms 55
TABLE 8. Data Compression Sysparms 57
TABLE 9. Required Data Compression Sysparms 59
TABLE 10. Optional Data Compression Sysparms 59
TABLE 11. Output Media Sysparms 60
TABLE 12. Available Values for Media Option Sysparms 60
TABLE 13. PDS Anomaly Codes and their Descriptions 77
TABLE 14. Possible M860 Values for ARC0TYPE 82
TABLE 15. Possible REDWOOD Values for ARC0TYPE 83
TABLE 16. ISPF Custom Report Field Names 85
TABLE 17. Parameters Passed to Dictionary Module 87
TABLE 18. Parameters Passed to Dictionary Module for Sorting 88
TABLE 19. Format of $FLGXyz 90
TABLE 20. Format of MMMMBsc 90
TABLE 21. Valid entries for MMMMBsc 90
TABLE 22. Threshold Manager Syntax Error Messages 92
TABLE 23. Device Type Character Code Values 98
TABLE 24. System Parameters 107
TABLE 25. Possible Values for ARBACKUP 124
TABLE 26. Possible Values Available for ARCHSORT 126
TABLE 27. Possible Values Available for ARCTNAME 127
TABLE 28. List of Possible Value for DYNCUNIT 143

TABLE 29. List of Possible Values for DYNTUNIT 143
TABLE 30. List of Possible Value for DYNnUNIT 144
TABLE 31. Possible Values Available for ISAMRECO 150
TABLE 32. Sample Values for Sysparm IXMQTIME 151
TABLE 33. List of Possible Values for MIGPSDEN 155
TABLE 34. Possible Values for sysparm MIGRSORT 155
TABLE 35. Possible Values for sysparm NEWPATHN 156
TABLE 36. Possible Values Available for RACFALLO 160
TABLE 37. Possible Results when using RESCRCAT 167
TABLE 38. Possible Values for RESPRISQ 168
TABLE 39. SECURDRS Authorization Levels 173
TABLE 40. Possible Values for SMSFORCE 177
TABLE 41. Possible Values Available for SYSPARMS 181
TABLE 42. Possible Values Available for USEDSSIO 189
TABLE 43. Common IDC3009I Return Codes 197
TABLE 44. VTOC Reader Names and Descriptions 200
TABLE 45. User Exits 203
TABLE 46. Parameters Passed to ALLOEXIT 206
TABLE 47. Parameters Passed to ALOCATEX 207
TABLE 48. Parameters Passed to ALOPOSEX 207
TABLE 49. Parameters Passed to ARCDSNEX 208
TABLE 50. Parameters Passed to ARCEODEX 208
TABLE 51. Parameters Passed to ARCHIVEX 209
TABLE 52. Parameters Passed to ARESPREX 210
TABLE 53. Parameters Passed to ARPOOLEX 211
TABLE 54. Parameters Passed to CCINITEX 211
TABLE 55. Parameters Passed to DARCHIEX 212
TABLE 56. Parameters Passed to DBRCPREX 213
TABLE 57. Parameters Passed to DCCMPEXn 213
TABLE 58. Parameters Passed to DCDCPEXn 214
TABLE 59. Parameters Passed to DCDSNDEX 214
TABLE 60. Parameters Passed to DEFARCEX 214
TABLE 61. Parameters Passed to DERASEEX 215
TABLE 62. Parameters Passed to DINXUFEX 216
TABLE 63. Parameters Passed to DRESTOEX 216
TABLE 64. Parameters Passed to DSBUSREX 217
TABLE 65. Parameters Passed to DSCBCVEX 218
TABLE 66. Parameters Passed to DSCLSCEX 219
TABLE 67. Parameters Passed to DSDISPEX 219
TABLE 68. Parameters Passed to DSNDELEX 220
TABLE 69. Parameters Passed to DSNDLPEX 220
TABLE 70. Parameters Passed to FMSSAFEX 221
TABLE 71. Parameters Passed to FMSSPREX 221
TABLE 72. Parameters Passed to GDGCATEX 222
TABLE 73. Parameters Passed to ICHEKAEX 222

TABLE 74. Parameters Passed to ICHEKPEX 223
TABLE 75. Parameters Passed to IXCTLGEX 224
TABLE 76. Parameters Passed to IXMREVEX 224
TABLE 77. Parameters Passed to LISTDMEX 225
TABLE 78. Parameters Passed to LISTDVEX 225
TABLE 79. Parameters Passed to LISTREEX 226
TABLE 80. Parameters Passed to MERDSNEX 226
TABLE 81. Parameters Passed to MERVOLEX 227
TABLE 82. Parameters Passed to MIGDISEX 227
TABLE 83. Parameters Passed to MIGGDGEX 228
TABLE 84. Parameters Passed to MIGREXIT 228
TABLE 85. Parameters Passed to RAADDAEX 228
TABLE 86. Parameters Passed to RAADDPEX 229
TABLE 87. Parameters Passed to RACFDAEX 230
TABLE 88. Parameters Passed to RACFDPEX 230
TABLE 89. Parameters Passed to RACFEAEX 231
TABLE 90. Parameters Passed to RACFEPEX 231
TABLE 91. Parameters Passed to RACFSMEX 232
TABLE 92. Parameters Passed to RACHKAEX 232
TABLE 93. Parameters Passed to RACHKPEX 233
TABLE 94. Parameters Passed to RADEFAEX 233
TABLE 95. Parameters Passed to RADEFPEX 234
TABLE 96. Parameters Passed to RADELAEX 235
TABLE 97. Parameters Passed to RADELPEX 235
TABLE 98. Parameters Passed to RECPRIEX 236
TABLE 99. Parameters Passed to RESAFTEX 236
TABLE 100. Parameters Passed to RESCRNEX and RESPRIEX 237
TABLE 101. Parameters Passed to RETAFTEX 238
TABLE 102. Parameters Passed to RETGDGEX 239
TABLE 103. Parameters Passed to RETPRIEX 239
TABLE 104. Parameters Passed to RLSEAFEX 240
TABLE 105. Parameters Passed to RPTDSBEX 240
TABLE 106. Parameters Passed to RPTDSNEX 241
TABLE 107. Parameters Passed to RPTDSUEX 241
TABLE 108. Parameters Passed to RPTMVDEX 241
TABLE 109. Parameters Passed to SCRTCHEX 242
TABLE 110. Parameters Passed to SECURPEX 243
TABLE 111. Parameters 5 - 8 (DSNAME) for SECURPEX 244
TABLE 112. Resource Type = 3 (VOLSER) for SECURPEX 244
TABLE 113. Resource Type = 4 (VSAM Catalog Name) 244
TABLE 114. Resource Type = 5 (Commands) for SECURPEX 245
TABLE 115. Resource Type = 6 (VSAM Component) 245
TABLE 116. Resource Type = 7 (Subfile) for SECURPEX 245
TABLE 117. Parameters Passed to SPACECEX 246
TABLE 118. Parameters Passed to SPFHDGEX 246
TABLE 119. Parameters Passed to SPFORDEX 247

TABLE 120. Parameters Passed to SPFORSEX 247
TABLE 121. Parameters Passed to SYSOUTEX 247
TABLE 122. Parameters Passed to TAOPPOEX 248
TABLE 123. Parameters Passed to TAOPPREX 248
TABLE 124. Parameters Passed to TAPEDLEX 249
TABLE 125. Parameters Passed to TMSCTLEX 250
TABLE 126. Parameters Passed to TPMGMTEX 253
TABLE 127. Parameters Passed to TSCHKAEX 254
TABLE 128. Parameters Passed to TSCHKPEX 255
TABLE 129. Parameters Passed to UNLOADEX 255
TABLE 130. Parameters Passed to USERSENQ 256
TABLE 131. Parameters Passed to USERVENQ 256
TABLE 132. Parameters Passed to USRCODEX 257
TABLE 133. Parameters Passed to VCCOPYEX 257
TABLE 134. Parameters Passed to VCDMIGEX 258
TABLE 135. Parameters Passed to VSACESEX 260
TABLE 136. Parameters Passed to VSALLOEX 260
TABLE 137. Parameters Passed to VSATTREX 261
TABLE 138. Parameters Passed to VSMAFTEX 262
TABLE 139. Parameters Passed to VSMPRIEX 263
TABLE 140. Parameters Passed to VSMSPWEX 264
TABLE 141. Parameters Passed to VSNEWNEX 264
TABLE 142. Parameters Passed to XCPARCEX 265
TABLE 143. Parameters Passed to XCPDSNEX 265
TABLE 144. Parameters Available to Module ADSUT028 266
TABLE 145. Format of the DMSMSGES Member of PARMLIB 267
TABLE 146. Module Intercept Control Stack Fields 274
TABLE 147. MI Trace Facility DD Statements 276
TABLE 148. Format of the MI Trace Control Statement 279
TABLE 149. MI Trace Options and Definitions 280
TABLE 150. MI Trace Output Explanation 280
TABLE 151. Archive Analysis User Exit 284
TABLE 152. Archive Fix Utility User Exit 289
TABLE 153. Parameters Passed to REPARCH User Exit 291
TABLE 154. Archive Tape Recovery Condition Codes 298
TABLE 155. Description of the MI CALL Structure 303
TABLE 156. Field Descriptions within DM175 304
TABLE 157. Format of PARMEDIT Entries 305
TABLE 158. Protection of Subfiles Under ACF2 321
TABLE 159. Protection of Commands Under ACF2 322
TABLE 160. Protection of Subfiles Under Top Secret 328
TABLE 161. Protection of Commands Under Top Secret 329
TABLE 162. Protection of Subfiles Under RACF 336
TABLE 163. Protection of Commands Under RACF 337
TABLE 164. List of RACF User Exits 350

TABLE 165. REPORTNM Content Descriptions 361
TABLE 166. REPORTNM Example 361
TABLE 167. Control Entry Description 362
TABLE 168. Header Entry Description 362
TABLE 169. Detail Entry Description 363
TABLE 170. Possible Line Type Values and Descriptions 363
TABLE 171. Total Entry Description 363
TABLE 172. Special Field Types for Position 4 364
TABLE 173. RPTNOMVD Example 364
TABLE 174. Header Entry Example 365
TABLE 175. Detail Entry Example #1 366
TABLE 176. Detail Entry Example #2 366
TABLE 177. Total Entry Example #1 367
TABLE 178. Total Entry Example #2 367
TABLE 179. Parameters Passed to module ADSDM033 368
TABLE 180. Record Format for the Owner Record 371
TABLE 181. Record Format of the Standard MVD 371
TABLE 182. Four Major Areas of ISAM data sets 380
TABLE 183. ISAM Variable Combinations 381
TABLE 184. Data Compression Techniques available 387
TABLE 185. List of Record Formats 396
TABLE 186. List of Valid Logical Record Types 469
TABLE 187. Record Types Generated for PS and DA 470
TABLE 188. Record Types Generated for HFS 470
TABLE 189. Record Types Generated for PO 470
TABLE 190. Record Types Generated for PDS and PDSE 470
TABLE 191. Record Types Generated for ISAM 471
TABLE 192. Record Types Generated for VSAM 471
TABLE 193. Record Format of Logical Record Types 472
TABLE 194. Archive Tape Format (Volume-level) 473
TABLE 195. Tape Format for Volume-level Backups 473
TABLE 196. Symbolic Parameters and Recommendations 494
TABLE 197. Format-1 DSCB Appendage Fields 499
TABLE 198. Format-4 DSCB Appendage Fields 500
TABLE 199. List of PROCS in CA-Disk Proclib 502
TABLE 200. List of PARMLIB Members and Descriptions 503