

CA Disk™ Backup and Restore

User Guide

r12.5, Second Edition



Third Edition

This Documentation, which includes embedded help systems and electronically distributed materials (hereinafter referred to as the "Documentation"), is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2015 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

CA Technologies Product References

This document references the following CA Technologies products:

- CA 1® Tape Management (CA 1)
- CA Allocate™ DASD Space and Placement (CA Allocate)
- CA Compress™ Data Compression (CA Compress)
- CA Datacom®/DB (CA Datacom/DB)
- CA Disk™ Backup and Restore (CA Disk)
- CA TLMS® Tape Management (CA TLMS)

Documentation Changes

The following documentation updates have been made since the last release of this documentation:

- [Topic Name]—[Description of new chapter, topic, or description of changes to an existing topic].
- [Topic Name]—[Description of new chapter, topic, or description of changes to an existing topic].

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At <http://ca.com/support>, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at <http://ca.com/docs>.

Contents

Chapter 1: General Information	25
About CA Disk.....	25
Learning to Use CA Disk	25
Basic Conventions and Requirements.....	26
Terminology	26
Control Statements	27
Data Set Names and Patterns	28
Tailoring to User Needs.....	30
Date Formats.....	30
Year 2000 Considerations	32
Separator Pages	34
Messages.....	34
User Abend Codes.....	34
Storage Memory Requirements.....	34
JCL Requirements.....	35
Data Set Allocation.....	35
Special Considerations	35
Types of Data Sets Processed.....	35
Direct Access and Physical Sequential Data Sets	36
PSE Data Sets.....	38
IMS and DB2 Databases	38
DBRC Support.....	39
IDMS Support.....	39
GDG Data Sets and Generation Wrapping	39
RESTORE/RECOVER and Compression Products	42
Abnormalities Detected	42
Password-Protected Data Sets.....	43
Multivolume Data Sets.....	43
StorageTek 4400 Automated Cartridge System	44
StorageTek Redwood Device	44
User Data Set Integrity.....	45
Concurrent Copy Support	48
General Restrictions	49
CA Disk and DFSMS — Overview.....	50
The Relationship Between CA Disk and DFSMS	52
DFSMS Hierarchy.....	52
Exploitation of DFSMS.....	54

Miscellaneous	55

Chapter 2: Basic System DMS PROC 59

Commands Executed by the DMS JCL Procedure.....	59
Data Storage Command Language	60
Coding Multiple Action Commands	61
DSCL Graceful Shutdown.....	61
Condition Codes	63
Messages.....	64
Commands	64
SCAN and FIND Rules	66
SELECT and EXCLUDE Rules	70
SET Command	74
SET Command Example.....	75
PASSWORD Command (For VSAM Only).....	75
SCAN Commands.....	77
SCAN REALVOLS Command.....	78
SCAN CATALOGS Command	78
FIND Command	79
Processing Considerations	80
Syntax.....	81
SELECT Command.....	82
Criteria Operators and their Meanings	88
DSCL SELECT CRITERIA Field Name Descriptions.....	92
DSCL Select Criteria Boolean (AND/OR) Logic.....	117
EXCLUDE Command	119
ACTION Commands	123
Complex DSCL Command Sequences.....	124
DFSMS and DSCL	125
Additional Select Parameter	125
SMS Data Class Fields.....	126
SMS Storage Class Fields	129
SMS Management Class Fields.....	130
SMS Storage Group Fields.....	132
SMS Volume Definition Fields	137
Criteria for CA—Select Interface	142

Chapter 3: Basic System FMS PROC 147

About FMS PROC.....	147
Condition Codes	148
Commands	148

FMS Command Sequence	149
General Rules Pertaining to the FMS Command Structure.....	149
SET Command	150
SET Command Syntax.....	150
PASSWORD Command (For VSAM Only).....	152
PASSWORD Command (For VSAM Only) SYNTAX	152
SCAN Command	153
SCAN Command SYNTAX.....	153
SELECT Command.....	153
SELECT Command SYNTAX	154
Criteria Statement Operands	155
FMS Selection Criteria Field Names	156
FMS Select Criteria Boolean (AND/OR) Logic.....	165
EXCLUDE Command	166
ACTION Commands	166

Chapter 4: FMMSGEN PROC 167

About FMMSGEN.....	167
REXX Program—ADSRX010	167
Logic Summary	167
Sample Templates.....	168
FMSSAMP1.....	169
Creating Multiple RESTORE Jobs.....	171
FMSSAMP2.....	172
Creating Customized Reports.....	173
Substitution symbolics	174
Calculated symbolics	176
Special symbolics	176
FMSSAMP3.....	177

Chapter 5: Reports 179

About Reports	179
General Information.....	181
REPORT Condition Codes	182
Restrictions	182
REPORT Commands and Parameters	183
REPORT Command Names	183
VREPORT Command Names.....	190
REPORT Field Legend.....	192
Using the ISAM Reports	194
ISAM REPORT Legend.....	195

CLDATA Description	198
Cluster Name Line	198
Data Component Detail Group (D)	199
Index Component Detail Group (I)	199
Extent Detail Line (R)	200
Space Allocation Detail Line (S)	200
Use Statistical Detail Group (U)	201
Password Detail Group (P)	201
Attribute Detail Group (A, Z)	202
Catalog Name Line (C)	203
Blank Line (B)	203
Association Lines (X)	203
CIA Report Description	203
CIA Report KSDS Catalog Information Section	205
Index Component Catalog Information	207
CIA Report Sequence Set Graphic Analysis	208
Description of Sequence Set Graphic Headings	210
Unreferenceable Data Control Intervals	211
CIA Report Data Component Graphic Analysis	214
Control Information Overhead	215
Description of Data Component Graphic Headings	216
CIA Report KSDS Data Component	217
Usable Free Bytes	218
Estimated New Record Additions	219
Estimated Dead Space	220
CIA Report ESDS Catalog Information Section	223
CIA Report ESDS Graphic Analysis	224
CIA Report RRDS Catalog Information Section	224
CIA Report RRDS Graphic Analysis	225
CIA Report RRDS Data Component Summary	226
Analyzing the CIA Report	227
Analyzing RRDS Relative Record Data Sets	227
Analyzing ESDS Entry-Sequenced Data Sets	228
Analyzing KSDS Key-Sequenced Data Sets	229
SMF Report	230
Syntax	231
JCL Example	231

Chapter 6: Backup/Archive 233

About Backing Up and Archiving	233
General Information	234

Terminology	235
Concurrent Copy Support	236
DFSMSdss Data Mover Interface	237
BACKUP and ARCHIVE Considerations	238
ARCRELTMSYSPARM Releases Tapes When Restore is Waiting	239
Change Bit	239
Midnight "Rapid Aging" Problems	239
Exempting Data Sets	240
Exempting Restored Data Sets	240
Backup of Volume VTOCS	241
Task Performance	242
Graceful Shutdown	242
Task Restart Considerations	243
ARCHIVE Report Sequencing Options	243
Data Set Compression and Reorganization	244
Fast Archive Processing	244
Non-VSAM Considerations	244
Backup of SYSCTLG	244
PDS Anomalies	245
Model DSCB Processing	245
PSE Considerations	245
VSAM Considerations	245
Verify of VSAM Clusters	246
Backing Up or Archiving Empty Clusters	246
Processing Clusters with Alternate Indexes	247
Deleting Unexpired Data Sets	247
Processing Clusters in Use by Other Tasks	248
Recataloging for VSAM Auto-Restore	248
VSAM Backup and Archive Restrictions	248
Access Methods for VSAM Clusters	249
ICF Catalog Backup and Recovery	250
Processing IMS Databases	252
Processing Controlled by CA Compress	253
Alternate Index Support	253
DBRC Support	254
Processing DB2 Databases	254
Processing Clusters that have Non-Standard Formats	256
Physical Sequential Data sets that have Non-Standard Formats	257
Incremental Backups	257
Volume Level Full Backups	258
Implementing Incremental Backup	259
BACKUP Command	262

Deferred Requests	263
Member Level Backup.....	263
BACKUP Command Syntax	264
SMS and BACKUP	268
BACKUPCC Command.....	269
DFSMSdss Data Mover	269
BACKUPCC Command Limitations.....	270
BACKUPCC Command Syntax.....	271
BACKUPCC Command Example	271
ARCHIVE Command	272
Deferred Requests	272
Deferred ARCHIVE Command Restrictions.....	273
ARCHIVE Command Syntax	273
SMS and ARCHIVE	277
ARCHIVE Command Examples.....	278
ARCHIVE Command Example 1	278
ARCHIVE Command Example 2	278
ARCHIVE Command Example 3	279
ARCHIVE Command Example 4	279
DELETE Command	280
DELETE Command Syntax.....	280
SMS and DELETE.....	281
DELETE Command Example 1.....	281
DELETE Command Example 2.....	281
DEL3435 Command	282
DEL3435 Command Syntax	282
DEL3435 Command Examples.....	282
Troubleshooting DEL3435	283
EXPIRE Command	283
EXPIRE Command Syntax	284
SMS and EXPIRE	284
EXPIRE Command Example 1	286
EXPIRE Command Example 2	286
EXPIRE Command Example 3	286
Sample Backup/Archive/Delete Report	287
Explicit ARCHIVE	291
Device Independence and Ease of Use	291
Explicit ARCHIVE Restrictions	293
Explicit ARCHIVE and DARCHIVE Syntax.....	296
Catalog Actions	297
Delete Actions	298
Reset Change Bit Actions	298

Processing Queued ARCHIVE Requests	298
Chapter 7: RESTORE/RECOVER	301
About Restoring and Recovering	301
General Information	302
Using the Bestfit facility for RESTORE/RECOVER of SMS Data sets	302
Multiple Files Data set for Restore	303
Restore Condition Codes	303
Non-VSAM Restore Volume Selection	304
Multi-Volume Restore	305
Deferred Restores (Queued Restore Requests)	306
Restore of SYSCTLGS	306
Restore Grace Periods	307
Restore To Preallocated Data Sets	307
PDS Member Restore	307
Restore to Unlike Device Types (Block Size Concern)	308
Restore Space Allocation	308
Restore of ISAM Data Sets	308
Restore Considerations for GDG Data Sets	308
PSE Considerations	309
Concurrent Copy Considerations	310
DFSMSdss Data Mover Interface	310
Restore and Recover of VSAM Clusters	311
Restrictions in Restore and Recover of VSAM Clusters	312
Special Processing Options	312
Secondary Extent Consolidation (Defragmentation of VSAM Clusters)	313
Default Catalog Processing	313
Speed Option	313
Changing Data Sets Attributes	314
Empty Clusters	314
Deleting Unexpired Data Sets	314
Last Use Date Maintenance	315
Overriding Erase Option When Deleting	316
Using NOSCRATCH Option When Deleting	317
Processing Clusters in Use by Other Tasks	317
Restoring to Predefined and Reusable Clusters	318
Optional Cluster Data Report at Restore Time	318
Determination of Data and Index Component New Names	318
Determination of Alternate Index and Path New Names	319
Restoring Alternate Indexes	319
Volume Selection for Alternate Indexes	320

Restoring Alternate Indexes which were Archived Separately	321
Volume Selection for Multi-volume Clusters	322
ICF Catalog Recovery	323
FILES Updates During RESTORE/RECOVER	324
Use of Primary or Copy Archive Volumes	324
Restrictions	325
RESTORE JCL	325
RESTORE Password Command	326
RESTORE Command Syntax	327
Parameters Honored Without the Defer Parameter	330
Parameters for Non-VSAM Data Sets	331
Parameters for Non-VSAM Without DEFER	332
Parameter for VSAM Clusters	332
Parameters for VSAM Without DEFER	333
Parameters for VSAM AIXS Without DEFER	334
Parameters for VSAM Clusters Without DEFER	335
Parameters for VSAM Data Components Without DEFER	339
Parameters for VSAM Index Components Without DEFER	342
JCL for Processing Queued Restore Requests	344
Tape Pull List for Queued Restore Requests	344
Listing of Deferred Restore Requests	345
Implicit Recovery and Volume Defragmentation	346
Features Unique to RECOVER Command	347
RECOVER Command	348
RECOVER Command General Information	348
RECOVER Command Syntax	350
RECOVER Command Examples	357
Volume Defragmentation	358
Sample Reports	358
RESTORE Command Report	359
RECOVER Command Report	359
SMS and RESTORE/RECOVER	360
RESIXRPD Concern	361

Chapter 8: VBACKUP/VRECOVER 363

About VBACKUP and VRECOVER	363
General Information	363
Volume-Level Backup	363
Volume-Level Recovery	364
VBACKUP Function	365
Offline VBACKUP	366

Track Overflow Data Sets	366
VBACKUP Security Considerations	367
Data Integrity Considerations	367
ARCHIVE Management Considerations.....	367
VBACKUP Command and Parameters	368
VRECOVER Function	370
Security Considerations	370
VRECOVER Command Syntax.....	371
VRECOVER Condition Codes.....	375
SMS and VBACKUP	375

Chapter 9: XCOPY 377

About XCOPY	378
XCOPY Limitations	378
Basic Operation	380
Automatic Selection	380
Secondary FILES	380
Expiration Date Support.....	381
XCOPY Condition Codes	381
Create a Secondary FILES	381
Identify Data Sets for Disaster Recovery	382
Determine Expiration Criteria	382
Running XCOPY.....	382
System Parameter Considerations.....	382
Simulate Mode Considerations.....	383
XCOPY Commands and Parameters	385
SET MODE= Command	385
SELECT Command.....	386
XCOPY Command	387
Ending an XCOPY Job.....	392
Sample JCL.....	392
Sample OUTPUT	393
User Exits.....	393
MERGUSEX--Merge Sort Field User Exit.....	393
XCPARCEXmmmmmmmm	394
XCPDSNEXmmmmmmmm	394
Record Formats	394

Chapter 10: FILES Maintenance 395

About Files.....	395
General Information.....	396

Maintenance Functions.....	397
LISTD Condition Codes	398
JCL - Index Listings.....	399
LISTV Command	399
LISTV Command Syntax.....	399
LISTV Flags.....	400
LISTV Sample Report	401
LISTD Command	402
LISTD Command Syntax.....	403
LISTD Flags.....	409
LISTD Sample Report	411
LISTD Report Field Description.....	412
LISTF Command	412
LISTF Command Syntax	413
LISTF Report Field Descriptions	413
LISTF Sample Report.....	414
LISTS Command	415
SMS Information Within The Archives	415
LISTS Command Syntax	415
LISTS Flags	416
LISTS Report Field Descriptions.....	416
LISTS Sample Report.....	416
IXMAINT Utility General Information	418
ENQUEUE RESERVE	418
Tape Management Support	419
Graceful Shutdown	419
Condition Code Settings.....	420
SMS Considerations	420
JCL - IXMAINT	420
Deleting Expired Data Sets.....	421
DSNDELETE Command	422
Deleting Expired Volumes	425
VOLDELETE Command.....	427
Purging Archive Data Sets/Volumes.....	428
PURGE Command.....	429
SMS and IXMAINT	430
SMS fields CA Disk Supports.....	430
Determining How Long To Keep DSNINDEX Records	432
Expiration Dates Inspected	432
Command Format	433
SMSPRINT and WHYFLAGS Diagnostics.....	434
Archive Index Rebuild Utility	436

Use of TRecover and Rebuild	436
Full Rebuild.....	436
ARONLY Rebuild	436
Selective Rebuild.....	437
Special Considerations	438
JCL - Rebuild Index	439
REBUILD Command and Parameters.....	440
IXUPDATE Utility.....	441
JCL - Index Updates	441
IXUPDATE Utility Condition Codes	442
DELETE Data Set Index Command and Parameters	443
CHANGE Data Set Index Command and Parameters	444
RESET Command	446
BUILDARC Command.....	447
Making Additional Copies of Archive Tapes	449
Tapes Originally Allocated with DYN1/2/n.....	450

Chapter 11: MERGE 451

About Merge	451
Overview	452
Specifying What to Merge.....	453
Specifying Merge Output Options.....	454
Implementation Concerns.....	455
Operation Concerns	456
TAPEPULL Processing	457
Fast Merge Processing	458
Abend/Restart Considerations.....	459
Simulate Mode Considerations	460
Processing Logic	461
JCL - MERGE Command	462
MERGE Condition Codes	463
MERGE Command	463
MERGE Command Syntax	464
Sample MERGE Reports	469
SMS and MERGE.....	470
RESIXRPD Concern	470

Chapter 12: MOVE/COPY 471

About Move and Copy.....	471
General Information.....	472
Error Recovery	472

Graceful Shutdown	473
ENQ/DEQ.....	473
Scratch/Catalog Defaults.....	473
Target Allocation	474
Excluding Data Set Names from Migration	474
Renaming Data Sets	474
Messages.....	475
Restrictions	475
Moving Data to Different Device Types	476
Special Considerations	476
GDG Data Sets	477
PSE Data Sets.....	477
Model DSCB Processing	478
BDAM Data Sets	478
SYSCTLG Processing.....	480
Reblocking	481
Specifications	482
User Exits.....	483
DSCLSCExmmmmmmmm	483
MOVE Command.....	483
MOVE Command Syntax	484
COPY Command	487
COPY Command Syntax.....	488
VSAM Cluster Considerations.....	491
SMS and MOVE/COPY	493
Guaranteed Space Attribute	494
Sample MOVE/COPY Commands	494
Sample Move/Copy Report	495

Chapter 13: Idle Space Release 497

About Idle Space Release	497
Graceful Shutdown	497
Restrictions	498
Messages.....	499
System Parameters	499
VSAM Considerations.....	500
PSE Considerations.....	500
RELEASE Command	501
RELEASE Command Examples	502
RELEASE Command Sample Report	503
SMS and RELEASE	504

Chapter 14: PDS Compression 505

About Compression	505
General Information	506
Restrictions:	506
Condition Codes	507
JCL — PDS Compression	507
SCAN Command	508
COMPRESS Command	514
RELOAD Command	516
JCL — RELOAD	516
Sample PDS Compression Report.....	517
SMS and PDS COMPRESS	517

Chapter 15: Sequential Migration to Tape 519

About Migration	519
General Information	519
Installation Options	520
Expiration Dates Assigned to Data Sets on Tape.....	521
Sequential Migration Versus Archival	522
Multivolume Tape Processing	522
Compatibility with Tape Management Systems	522
Creation of Sequential Migrate Tape First File.....	523
Assignment of Creation Dates to Output Data Sets	524
Migrating Protected Data Sets	524
Reports	524
Restrictions	525
Condition Codes	525
JCL - Sequential Migrate	525
Sequential Migrate Commands	525
MODESET Command	526
MIGRATE Command.....	527
SCAN Command	528
Migration Concerns	530
Catalog Concerns	530
MIGRECAT Function Description	530
RECATALOG Command	531
PURGE Command.....	531
LIST Command	532

Chapter 16: DASD Billing 533

About Billing	533
General Information	534
Elapsed Mode.....	534
Immediate Mode	535
Billing Rates	535
Computing the Billing Charge.....	535
Assign Volumes to Billing Categories	536
Define Billing Rates for Each Category	536
Example of Assigning Billing Rates	537
Billing Units System Parameters	537
Using Other Currencies	537
User Code/Account Codes	538
Assigning User Codes	538
User Code Description Table	540
Total Charges	540
Exempting Data Sets	540
User Exits.....	541
DSBUSREX	541
Condition Codes	541
Accumulate JCL for Non-VSAM Data Sets	542
SELECT Command.....	542
ACCUMULATE Command	543
Accumulate JCL for VSAM Clusters	545
DASD Billing Total JCL	546
Implementation - Suggested Run Procedures.....	547
Billing Processing Considerations.....	548
DASD Billing and Archival Functions	548
Increased Billing Buffers.....	549
Definitions of Fields in the DASD Billing Report	550
DASD Billing Samples.....	552

Chapter 17: ISPF 555

About ISPF Support	556
Using the ISPF Support	557
List of Available ISPF Functions	558
ISPF Return Function Restriction.....	560
ISPF Panel Examples	561
Condensed Selection Menu	561
Selection Menus (Reports)	562
Selection Menu (List)	563

Selection Menu (Archive/Restore)	564
Selection Menu (Release/Compress/Migrate)	565
Archive Management and Miscellaneous Functions	566
DSCL Functions	566
Enter Request to Archive Data Set	567
CA Disk Catalog Data Set List	568
Enter Request to List/Process Archive Index	569
Archived Data Set Index Selection	570
Data Set Index Record Display	571
Options Menu Pop-Up Panel	572
DSNINDEX Flag Field Status Indicators — Panel 1	573
DSNINDEX Flag Field Status Indicators — Panel 2	574
DSNINDEX Flag Field Status Indicators — Panel 3	575
DSNINDEX Flag Field Status Indicators — Panel 4	576
DSNINDEX Flag Field Status Indicators — Panel 5	576
Compress PDS Data Set	577
Generate JCL — Panel 2	578
Generate JCL - Panel 3	579
Overview of ISPF Online Reporting Facility	580
Hierarchy of ISPF Online Reporting Panels	582
ISPF Online Reports	588
Allocate Library for Report Definition	589
Select Desired Report	590
Display Report Information	590
Specify New Report Information	591
Select Report Field	592
Specify Field Position on Report	593
Select Sort Fields	595
Specify Sort Field Information	597
Select Any Sort Fields Where Subtotaling is Desired	598
Select Subtotal Fields	600
Specify Subtotal Information	601
Select Grand Total Fields	603
Specify Grand Total Information	603
Online Reporting System (Option 5)	603
Enter Data Set Name	604
Specify Subset of Real DASD Volumes to Process	605
Specify Selection Criteria You Desire	606
Specify DSNAMES and/or Table	606
Select Desired Reports	606
Confirm DSCL Report Requests	607
JCL for Batch Reports	608

Browse Report.....	609
Report Disposition.....	610
Build Batch Commands	612
Online Reporting - Expert Mode	614
DSCL Command Edit.....	615

Chapter 18: TSO Command Processor 617

Commands Available	617
General Information.....	618
DARCHIVE — Deferred Archive	618
Restrictions	618
DARCHIVE Command	619
DRESTORE — Deferred Restore	621
DRESTORE Command.....	622
RESTORE — Immediate (Dynamic) Restore	623
RESTORE Command	624
LISTDMS — Listing the Archive Index.....	625
LISTDMS — Sample Output.....	626
LISTREQ — List Status of Deferred Requests.....	628
LISTREQ command	629
LISTREQ — Sample Output.....	630
DERASE — Erase a Deferred Request.....	631
Restrictions.....	631
User Exits.....	632

Chapter 19: Testing CA Disk after Installing 633

Test the CA Disk System	633
Test Various Volume and Data Set Level Reports (IVP01RPT)	634
Test Move and Copy Functions (IVP02MOV)	634
Test Idle Space Release Function (IVP03ISR)	635
Test Backup and Batch Restore Functions (IVP03ISR).....	635
Test Expire Function (IVP05EXP)	636
Test Various PDS compress Functions (IVP06CMP)	636
Test XCOPY Function (IVP07XCP)	636
Test Merge Functions (IVP08MRG)	637
Test Recover Function (IVP09DRC)	637
Test Delete Function (IVP10DEL)	637
Test Files Data Set Management Functions (IVP11IXM)	638
Testing the CA Disk Auto-Restore Feature	638
Forcing an Auto-Restore to be Initiated.....	638
Special Testing Environment	639

Appendix A: Health Checks 641

DISK_SYSPARM_CARTCALC_N	642
DISK_SYSPARM_DSNDUPVL_N	643
DISK_SYSPARM_IOCHNBLK_1	644
DISK_SYSPARM_IOTRACKS_1	645
DISK_SYSPARM_MERGSORT_S	646
DISK_SYSPARM_IXMQTIME_0	647

Appendix B: Examples 649

REPORT Examples	649
Example 1—Multiple Volume Data Set	649
Example 2—Use of DSNNAME Parameter	650
Example 3—Find Selected Proclib Data Sets	650
Example 4—Compare Member Names	651
VREPORT Examples	651
Example—Produce a Volume Allocation Summary	651
ARCHIVE Examples	651
Example 1—Archive Selected Data Sets	652
Example 2—Defer the Archive of Data Sets	652
Example 3—Create Backup Copies	652
Example 4—Dispose of Queued Requests	653
Example 5—List the ARCHIVE Request Queue	653
Example 6—Delete Entry from ARCHIVE Queue	653
RESTORE Examples	653
Example 1—Restore a Data Set	654
Example 2—Restore a Data Set from a Previous Copy	654
Example 3—Restore Two VSAM Clusters	655
Example 4—Restore to a Different Catalog	655
Example 5—Restore a VSAM Cluster and Alternate Indexes	655
Example 6—Restore a VSAM Cluster and Change Control Interval	656
Example 7—Restore a Base Cluster to a New Name and Password	656
Example 8—Process Queued Requests	656
Example 9—List the RESTORE Request Queue	657
Example 10—Delete Entry from RESTORE Queue	657
Volume—Level BACKUP Examples	657
Example 1—Volume-Level Backup with shared ENQs	657
Example 2—Volume-Level Backup with no ENQ Processing	658
Volume—Level RECOVER Examples	658
Example 1—Volume-Level Recovery, No Enqueues or Reserve	658
Example 2—Volume-Level Recovery from Specified Tracks	658
Volume Defragmentation Examples	659

Why Space Allocations Fail.....	659
Solution—Release Overallocated and Unused Space	659
Solution—Move and Recatalog.....	659
Solution—Defragmentation with ARCHIVE and RECOVER.....	660
RECOVER Examples—FMS PROC.....	661
Example 1—Recover Between Specified Dates	662
Example 2—Recover Between Specified Times	662
Example 3—Recover Data Sets from Pseudo Volume	662
Example 4—Recover Data Sets Specifying Location and Portion of the Name	663
Example 5—Recover Data Sets Specifying Volume Patterns and Locations.....	663
Example 6—Recover Current Backups by Name	663
Example 7—Recover Data Sets Specifying Volume Patterns	664
Example 8—Recover Selected Data Sets to a Specified Volume	664
Example 9—Recover Selected Data Sets Archived Yesterday.....	664
Example 10—Recover Data Sets with Specified Criteria.....	665
Example 11—Recover Data Sets with Specified Criteria.....	665
Example 12—Recover Archived Data Sets and Catalogs	666
Example 13—Recover with Tape I/O Error Abend.....	666
Example 14—Restarting a Recovery	667
Example 15—Plan Disaster Recovery.....	668
Example 16—Recovery with GDG and GDGLIMIT.....	669
BACKUP Examples	671
Example 1—Typical Backup Examples	671
Example 2—Backup Specifying Beginning Letters of Data Set Names.....	672
Incremental BACKUP Examples	672
Daily Incremental Backup Example.....	672
LISTD/LISTV Examples	673
Example 1—List All Archived Data Sets on a Tape	673
Example 2—List Selected Archived Data Sets.....	673
Example 3—List Archived Data Sets From from Archive-to-Disk and Tape	673
Example 4—List Selected Archive-to-Disk and Archive Tape Data Sets.....	674
IXMAINT Examples	674
Example 1—Delete Expired Data Sets.....	674
Example 2—Delete Specified Archive Tape and Data Sets	674
MERGE Examples	674
Example 1—Merge Selected Archive Tapes.....	675
Example 3—Merge Selected Disk Archives to Tape.....	676
REBUILD Examples.....	676
Example 1—Rebuild an Archive Index for a Tape	676
Example 2—Rebuild an Archive Index for a Specified Archive Data Set	676
IXUPDATE Examples	676
Example 1—Delete Archive Index Entries.....	677

Example 2—Update the Expiration Date	677
Example 3—Disable an Archive tape Index.....	677
Example 4—Create and Update an Entry for an Archive Tape	677
Example 5—Delete a Copy Volume from its Primary Volume	678
Idle Space Release Examples.....	678
Example 1—Scan for Overallocated PO Data Sets and Release Part of the Space	678
Example 2—Scan for Overallocated PO Data Sets and Release All Extra Space	678
Example 3—Release Space with Specified Criteria	679
Sequential Migrate Examples.....	679
Example—Migrate Specifying Part of Name and Generations	679
PDS Compression Examples	679
Example 1—Compress Specified Data Sets by Percent Full and Number of Extents.....	680
Example 2—Compress ISPF Profile PDSs and Reallocate	680
Example 3—Select Empty PDSs and Reallocate	680
Move/Copy Examples.....	681
Example 1 — Compress by Moving.....	681
Example 2 — Convert a Disk Volume to a New Device Type	681
DASD BILLING Examples	682
PARMLIB(DSBCATEG) Contents.....	682
XCOPY Example	684

Appendix C: Sample Reports 685

Report Example	685
Sample DSCL REPORTS	685
DFSMS and Reports.....	697

Appendix D: Glossary 699

Chapter 1: General Information

This section contains the following topics:

[About CA Disk](#) (see page 25)

[Learning to Use CA Disk](#) (see page 25)

[Basic Conventions and Requirements](#) (see page 26)

[Special Considerations](#) (see page 35)

[General Restrictions](#) (see page 49)

[CA Disk and DFSMS — Overview](#) (see page 50)

[DFSMS Hierarchy](#) (see page 52)

About CA Disk

The *CA Disk User Guide* provides instructions for using CA Disk. CA Disk is an automated data storage management system designed to assist the storage administrator in controlling and using data storage. Extensive reporting capabilities, sufficient free space, backup and recovery, installation standards and device conversions are some of the features available with CA Disk. ISPF menus and help text are available to invoke all functions and to display the status or results of prior requests.

CA Disk gives you the flexibility to establish data management standards based on your installation needs. It allows you to manage space according to the kind of data you have and how it is used. It lets you design and implement controls for DASD and tape usage throughout your storage hierarchy, and gives your data center support staff and the end user an effective and convenient means to manage their data.

Learning to Use CA Disk

The *CA Disk User Guide* contains information necessary to run CA Disk after it has been installed. It is organized into sections according to the functions of the system. Each section describes the purpose of the function, how to use it, and considerations for implementing it. Because this book is organized like a reference manual, it is not easy to learn how to use the system by reading it from beginning to end. If you are new to CA Disk, use the following list of items as a guide for getting to know the system.

1. Review the Table of Contents of the manual to get an idea of the functions the system provides and what information is supplied about them.
2. Continue reading this entire section of the manual. This helps you learn how the system operates, what its restrictions are, and how to code CA Disk command statements.
3. For more information, see the chapter "Basic System \— DMS PROC". This helps you learn how to write the CA Disk command set used by most CA Disk functions.

4. See the section CA Disk Libraries in the *Systems Guide*. It describes the CA Disk PARMLIB, which is used by all CA Disk functions. It is very important to understand its purpose and how it is used.
5. At this point you are prepared to learn about specific functions of the system. To do this, select a function that you are interested in and review the section for it. We suggest that you select the REPORT function as the first to learn because it is the easiest to use and provides a base of knowledge on how CA Disk generally operates.
6. After you have reviewed the section and feel you generally understand what it does and how to use it, see the chapter "Examples" and find the examples pertaining to the function. Review each example.
7. Select one of the examples and tailor it for your installation. Key it into your system and execute it. Verify that it does what you expected. If not, consult the section on the function again.
8. Be sure that when running these test cases you specify the SET MODE=SIMULATE command or the SIMULATE command parameter for all functions that provide it. After you have achieved the desired results with SIMULATE, remove the parameter (or comment it out), and run the function again and CA Disk actually performs the requested task.

Using these techniques, review each function. Begin with simple cases, and then add to them. They help you to better understand the different options of each function and what they do. Keep track of your test output for future reference.

Basic Conventions and Requirements

Throughout the operation of CA Disk, certain basic conventions and requirements are observed that help make processing consistent. These are summarized in the following pages.

Terminology

Several data storage management terms are commonly used from location to location, but the meanings given to the terms can vary widely. See the "Glossary" for descriptions that clarify both the common terms as well as those unique to CA Disk.

Control Statements

All COMMANDS consist of a character string of ten characters or fewer. While a command does not have to begin in any particular position, nothing can precede it in a statement.

Following a command (and preceded by one or more spaces) is a series of PARAMETERS that can consist of a combination of keyword and simple entries. Parameters can be specified in any order for a given command.

Certain KEYWORD PARAMETERS can supply either a single value or a list of values contained within parentheses and separated by commas. The limit on the number of values in a list varies with each particular command. Common examples are lists of data set names and volumes.

COMMAND STATEMENTS can be continued to additional lines in two different ways.

- One technique permits termination of one statement line with a complete parameter and a trailing comma, with the remaining parameters starting in any position on the next line.
- An alternate continuation technique requires coding of the statement being continued through column 71 and placement of an X in column 72 to designate that the statement is being continued. The first non-blank character in the following statement is appended to the character in column 71 of the previous statement.

Whenever many parameters are specified for a given command, the first parameter must begin in the statement containing the command. A statement with only a command is processed as having no parameters.

Parameter names can be abbreviated down to their first three characters, unless more characters would be needed to uniquely determine between similarly named parameters.

COMMENTS are permitted within the control statements. An ASTERISK in column 1 designates them. They are printed out in the CA Disk CMDPRINT exactly as they appear within the control stream.

Data Set Names and Patterns

The DSN= parameter is probably the most frequently used of all parameters, and the one where the greatest degree of flexibility is usually required. CA Disk allows either full data set names or data set name patterns to be entered in the DSN= parameter or in any CA Disk table that contains a data set name field. A pattern name consists of the usual alphanumeric and national characters allowed in a data set name, but also the following:

Character	Hex Value
An asterisk (*)	x'5C'
A question mark (?)	x'6F'
A slash (/)	x'61'
An exclamation point (!)	x'5A'

The asterisk (*) can be used to represent any variable index level or simple name. It must be the last character in the code.

Example	Result
DSN=*	Selects all single-level data set names.
DSN=*. *	Selects all two-level data set names.
DSN=A.*.SOURCE	Selects all three-level names that have <i>A</i> as their first index and <i>SOURCE</i> as the simple name, but any second-level index.
DSN=A*.SOURCE	Selects all data sets having data sets having a first-level index beginning with the letter <i>A</i> followed by any other characters, and a second-level index equal to the value <i>SOURCE</i> .

The question mark (?) can be used to represent any variable character within an index level or simple name. Multiple occurrences can be used within each level or simple name.

Example	Result
DSN=?	Selects all single-character data set names.
DSN=A.TEST??	Selects all two-level data set names with a first-level index of <i>A</i> and a simple name six characters in length, the first four of which must be <i>TEST</i> ; for example, <i>TEST01</i> , <i>TEST02</i> , <i>TEST99</i> , <i>TEST1A</i> , <i>TESTXX</i> , and so on.

The slash (/) can be used to represent any variable character from that position to the end of the name. The portion of the name that precedes the slash is referred to as a PREFIX name. Examples of the Slash (/) Pattern in DSN= are listed in the following table.

Example	Result
DSN=A/	Selects all data sets that begin with the letter <i>A</i> .
DSN=A.TEST/	Selects all data sets that begin with the character string <i>A.TEST</i> , with any following characters.
DSN=A.*.C?./	Selects all data sets that begin with an index of <i>A</i> , followed by any second-level index, a two-character third-level index starting with <i>C</i> , and any following string.

The exclamation point (!) can be used to represent any variable character up to the character string following the exclamation point; for example, the exclamation point defines the beginning of a character string (terminated by the next pattern character or the end of the data) that can be found anywhere within the name. The exclamation point (!) within an index level indicates that zero or more characters can occupy that position.

Example	Result
DSN=!TEST	TEST selects all data sets that contain <i>TEST</i> somewhere in the name.
DSN=A?.!DEPT21	Selects all data sets that have a two-character first-level index that starts with an <i>A</i> and contain <i>DEPT21</i> somewhere in the remainder of the name.
DSN=!TEST!LOAD	Selects all data sets that contain <i>TEST</i> somewhere in the name and <i>LOAD</i> somewhere following <i>TEST</i> .
DSN=TEST.L!AD	Selects all data sets that start with <i>TEST.L</i> and have <i>AD</i> somewhere following.

The exclamation point (!) character can also be used to find a pattern occurring only at the end of a string. This can be accomplished by placing an exclamation point at the end of the desired pattern as well as at the beginning, as in the following example:

Example	Result
DSN=!TEST!.LOAD!	Selects all data sets that contain <i>TEST</i> somewhere in the name and which terminates with <i>.LOAD</i> . Therefore, the data set name <i>LABS.TJP.TEST.LOAD</i> would match the pattern, but <i>LABS.TEST.LOADOLD</i> would not.

Data Set Name Validation

CA Disk commands that name a single specific data set to be processed are referred to as *explicit* commands. Those that cause VTOC and/or catalogs to be scanned are called *implicit* commands. By definition, the data set name entered on an explicit command cannot contain any pattern characters. The specific name entered (original and perhaps new name) is validated according to the requirements as specified in the IBM JCL reference manual—rules for coding the data set name field. Invalid characters will cause the command to be rejected. Implicit commands do not have the data set name/pattern entries validated in this manner, since they process names that already exist in the VTOC or catalog.

Volume Prefixes

The VOL= parameter can specify either complete volume serial names or volume patterns; for example, a partial volume serial name with an appended slash (/) to signify any variable character from that position to the end of the field. CA Disk tables that contain the volume field also allow volume prefixes unless specifically noted otherwise.

Example	Result
VOL=(PACK01,WORK/)	Scans volume PACK01 and all volumes that begin with <i>WORK</i> .
VOL=(/)	Scans all volumes.

Tailoring to User Needs

Many system parameters exist that cause slight variations from the default program execution. If a particular variation is desired, the appropriate sysparm is placed in PARMLIB member SYSPARMS. Sysparms applicable to each CA Disk function are cross-referenced and explained in the *Systems Guide*. A description is also given of how sysparms can be dynamically overridden in those special cases that warrant it.

For more information, see Activating CA Disk Security Features and Interfaces in the *Installation Guide*, to review how to protect these system parameters.

For larger scale variations, user exits are available. For more information, see User Exits in the chapter "XCOPY."

Date Formats

You can enter dates as input parameters in any of the following formats (unless otherwise noted in the documentation of particular CA Disk functions). For more information, see Year 2000 Considerations.

Gregorian Date Formats

Gregorian date formats include the following: ddmonyy, ddmonyyyy, mm/dd/yy, or mm/dd/yyyy. The format is:

dd

Calendar day of the month from 01to 31.

mon

Calendar month such as JAN, FEB, or MAR.

mm

Calendar month number from 01 to 12.

yy

Calendar year from 00 to 99 this year is stored as 1900 to 1999.

yyyy

Calendar year from 1900 to 2155.

Julian Date Formats

Julian date formats include the following: yyddd, yyyyddd, yy.ddd, yyyy.ddd, yy/ddd, or yyyy/ddd. The format is:

ddd

Day of a year from 001 to 366.

yy

Calendar year from 00 to 99 this year is stored as 1900 to 1999.

yyyy

Calendar year from 1900 to 2155.

Julian dates 98000, 99000, 99365, 99366 and 99999 are also accepted in any of the previous formats. For example, you can enter the date July 19, 1997 as:

97200	1997200
97.200	1997.200
97/200	1997/200
19Jul97	19Jul1997
07/19/97	07/19/1997

Also, as a DSCL Select or Exclude criteria, you can specify the date in the form TODAY, TODAY-n or TODAY+n, where *n* is a number of days in the past or future. For example:

- To select data sets created today you can enter:

```
SELECT CRITERIA=(CREDIT,EQ,TODAY)
```

- To select data sets modified within the last 5 days you can enter:

```
SELECT CRITERIA=(MODDT,GE,TODAY-5)
```

- To select data sets not used within the last 30 days you can enter:

```
SELECT CRITERIA=(USEDLT,LT,TODAY-30)
```

- To select data sets expiring within the next week you can enter the following statement. Note that this will automatically bypass *special expiration dates* as described in EXPDT — Expiration date (date format) in the chapter "Basic System – DMS PROC."

```
SELECT CRITERIA=(EXPDT,LT,TODAY+7)
```

Year 2000 Considerations

CA Disk is Year 2000 compliant. Year 2000 date testing was accomplished on a P/390 with CA Disk 9.0 (PD9839) installed. All functions were tested while the system was IPL'd with the following dates:

- 1999.365 — end of century
- 2000.001 — beginning of century
- 2000.060 — February 29

CA Disk follows all IBM conventions to identify dates within specific centuries in the following areas:

- Non-VSAM data sets use discontinuous binary, an xxyyy format where xx is from 0 255 and is the year when added to a base of 1900.
- VSAM and tapes use a separate century field.
- The FILES uses the discontinuous binary for all dates.
- CA Disk REPORTS use the mm/dd/yyyy.

Special Dates

CA Disk treats Julian dates 99365, 99366 and 99999 as *never scratch* dates. That is, data sets whose ARCHVOLS or DSNINDEX records use these dates as expiration dates will not be deleted automatically as expired by the CA Disk function IXMAINT. In addition, if sysparm UNCATDEL defaults to a value of N, CA Disk treats DSNINDEX records with an expiration date of 99000 as *never scratch*.

If you specify sysparm UNCATDEL with a value of Y, CA Disk treats Julian date 99000 as a *catalog control date*. That is, data sets whose DSNINDEX record has an expiration date of 99000 will be deleted automatically by the CA Disk function IXMAINT when the data set becomes uncataloged.

JCL, SYSPARMS, or Command parameters

Even after January 1, 2000, CA Disk will continue to honor these special *never expire* dates. If JCL specifies LABEL=EXPDT=99365, the data set(s) processed will be kept until explicitly deleted even after December 31, 1999.

For example:

- If you explicitly specify EXPDT=99365 for an ARCHVOL or DSNINDEX record, then the data set is kept until explicitly deleted even after December 31, 1999.

```
SCAN REALVOLS
SELECT DSNAMES=PROD./
BACKUP EXPDT=99365
//SYSPARMS DD *
DYNEXPDT=99365
```

The ARCHVOLS and DSNINDEX records created by this sample BACKUP job will have their respective EXPDT values set to 99365. These records will reside in the FILES even after December 31, 1999.

- If you specify RETPD=nnnnn and the system calculates an expiration date of the last day 99365, the actual expiration date will be *bumped* one day to January 1, 2000, to prevent inadvertently assigning a *never expire* date.

```
//jobcard
//BACKUP EXEC DMS
//ARCHIVE0 DD LABEL=RETPD=nnnn
SELECT DSNAMES=PROD./
BACKUP RETPD=nnnn
```

The ARCHVOLS and DSNINDEX records created by this sample BACKUP job will have their respective RETPD values set to *TODAY+nnnn*. If *TODAY+nnnn* calculates to 99365, the actual expiration date is *bumped* to 2000.001.

- If you specify RETPD=nnnn and the calculation extends beyond 2155.365 then the EXPDT is set to 2155.365.

Important! RETPD values calculating to 98nnn and 99nnn can cause problems in certain situations. For example, TMS considers tapes valued at 99060 *scratch* after 60 days of non-use.

Separator Pages

The number of CA Disk separator or header pages can be controlled by sysparm FLYCOUNT. From zero to nine pages can be produced. This sysparm can be overridden in your PARMLIB member SYSPARMS. For details, see *Overriding Sysparms Instream* in the *Systems Guide*.

Messages

CA Disk MESSAGES all begin with a common format of MODULENM MSGN, where MODULENM indicates the name of the module issuing the message, and MSGN is the message number itself. This identifying information is then followed by explanatory text. Further explanations for each message are found in the Message guide. A more detailed discussion of CA Disk message handling can be found in *Message Formatting* in the *Systems Guide*.

User Abend Codes

CA Disk issues USER ABENDS when it is impossible to issue a diagnostic message, when additional information to diagnose a problem is needed, or when due to data integrity it is unsafe to continue.

24

MSGPRINT DD statement missing.

100

Message will explain; dump attempted.

200

Message will explain; no dump attempted.

Storage Memory Requirements

Buffers for CA Disk I/O areas as well as memory tables for such items as member, alias, and note list entries are dynamically built. In some cases, CA Disk does not create variable length memory tables above the 16MB line. Since these items vary for each data set and device type being processed, exact storage requirements cannot be given.

Regions sizes on supplied JCL procedures range from 1024 to 5120K to accommodate running CA Disk with the maximum buffer space possible. The amount of region required below the 16MB line will vary depending upon the size of your catalogs, the number of your DASD volumes, and so on.

JCL Requirements

JCL procedures are provided for most CA Disk functions, and DD statements for any volume or data set that CA Disk has been instructed to process are dynamically allocated as needed, freeing the user from any need to modify JCL. For a list of these procedures and DD statements, see Procedures and Associated DD Statements in the *Installation Guide*.

Data Set Allocation

All CA Disk function allocations use SVC 99 for dynamic allocation of non-VSAM data sets. Using SVC 99 allows allocation control products such as CA Allocate or IBM DFSMS to redirect allocations.

CA Disk uses SVC26 to define the VSAM definitions. These definitions are candidates for redirection to other volumes by a pooling package.

Due to SVC 99 restrictions, CA Disk does not set the DFSMS ACS &ACSENVIR variable. To keep data set allocations consistent, ACS &ACSENVIR is not set for VSAM allocations.

Special Considerations

Types of Data Sets Processed

CA Disk typically processes data by the movement and management of physical data blocks. In some cases, depending on actions that are requested by the user, CA Disk can process at the logical record level. CA Disk has its own *access method* (written at the channel program level) that does *full track reads and writes*, processing as much as one cylinder per I/O. In most CA Disk functions, however, CA Disk is not a *track-image* processing system.

Data is processed at the physical block level to allow CA Disk to manage data without concern for source and target device dependencies, and to allow maintenance functions to be performed during data movement (for example, compress PDS data sets). (An exception to this type of processing is the CA Disk volume-level processing for VBACKUP and VRECOVER, discussed in the following sections.)

For this reason, CA Disk requires that all data sets to be processed be in a format that is standard to the MVS operating system environments. Processing is done for each data set based on the attributes that are carried for it in the format-1 DSCB of the volume VTOC. Attributes that are relied upon are DSORG and RECFM.

In general, if a data set internal record format is not compatible with the standard meanings of the DCB DSORG attribute specified in the VTOC, CA Disk cannot process it successfully. In cases where DCB attributes are not present, CA Disk processes the data set with default attributes.

CA Disk users should be aware of these kinds of data sets and investigate their compatibility before allowing processing against them. Data sets that are in question are generally special-purpose data sets. They can be part of an in-house generated system or used by a purchased product. In most cases, they will be documented as being of a special format and maintainable only through utilities that are supplied with their associated system. These kinds of data sets usually constitute only a small fraction of the total DASD data sets in the user installation.

Track Image Processing for Volume-Level Functions

An exception to standard CA Disk processing at the data set level explained previous is the volume-level processing for the VBACKUP and VRECOVER functions. These functions back up, restore data at the volume level, and use *track image* processing. CA Disk does not examine separate data sets in these functions, and the data is restored exactly as it was backed up, but for an entire volume, not a single data set. Since no catalog action is taken during track image processing, you must verify that the catalog and recovered volume are in sync.

Direct Access and Physical Sequential Data Sets

As explained previous, CA Disk processing of data sets is based on the data set attributes. Sometimes these attributes conflict with the actual data or the attributes are incomplete. The following explains how CA Disk will treat a variety of possible attributes during Archive, Restore, Move, and Copy processing of physical sequential and direct access data sets.

The attributes CA Disk takes into consideration during processing include the DSORG, RECFM, LSTAR, BLKSIZE and KEYLEN. During processing, CA Disk can also compress the data set, which will have an effect on whether or not TTR addressing is maintained.

The data set organization attribute (DSORG) and its types will be referred to as:

PS

Physical Sequential

DA

Direct Access

UNK

Unknown

The record format attributes (RECFM) and its types will be referred to as:

F

Fixed or Fixed Blocked

UV

Undefined or Variable

FBS

Fixed Block Standard

UNK

Unknown

The LSTAR is a pointer to the last record in the data set and is in TTR format. TTR format is a three-byte value. The first two bytes are the track number relative to the start of the data set, and the last byte is the record number relative to the start of the track. TTR addressing refers to a pointer for a data block.

Data set compression can occur when a data set is restored or moved to a DASD device with a different track length, if there is empty space on a track, or if there is an empty track in the middle of a data set. Each record is written on the next available space after the previous record. This can change the TTR address of the record and *compress* the data set. Under some conditions this is not desirable.

Data sets with a DSORG of DA and RECFM of UV require special handling. When a data set with DSORG of DA and RECFM of UV is archived by CA Disk, an *ASDA* flag is set, indicating the data set is archived as a direct access data set. This flag is also set if an empty track is detected anywhere in the data set prior to the end-of-file (EOF). At restore time, if this flag is set, the TTR value for each record is maintained. That is, a record is written back in the same relative position from which it was read.

This also occurs when sysparm HONORTTR is specified with a value of Y, regardless of the RECFM of the PS, DA, or UNK data set.

Note: ESF Spool data sets must use HONORTTRY if the Move, Copy, Backup and Restore commands are to function properly.

Data sets are considered *empty* if the LSTAR value is zero. However, a data set with DA in DSORG and RECFM is UV can have all of these attributes and not be empty. In this case, sysparm IGNLSTAR can be specified with a value of Y to cause CA Disk to check bit 0 of the DS1ND field in the format-1. For more information, see the sysparm description for IGNLSTARy in the *Systems Guide* or data sets with a RECFM of FBS, the BLKSIZE is important. On any other PS or DA data sets, when the EOF is written, it is written on the next track after the last data track. An FBS data set must have the EOF on the same track if another record of BLKSIZE length can be written to that track.

CVOL data sets (a type of IBM catalog whose data set name is SYSCTLG) have a BLKSIZE of 256 bytes and a KEYLEN of eight bytes. Their DSORG is usually either UNK or CTLG. However, they are really PS data sets. IBM requires CVOLs to have a KEYLEN of zero, and their tracks formatted to 25 byte blocks with an eight-byte key. The LSTAR record value must also be X'FF'. After CA Disk has processed a CVOL data set, the DSORG is defined as PS instead of UNK. This does not affect CVOL processing.

CA Disk handles any data set with a DSORG of UNK as a PS data set. Data sets with a RECFM of UNK are treated as though they have a RECFM of U. This corresponds to IBM's handling of the same data sets. Data sets with a DSORG of UNK will have a DSORG of PS after CA Disk processing.

Note: The Idle Space Release function will release space from UNK dsorg's by default. You can instruct CA Disk to bypass this type of DSORG by executing:

```
EXCLUDE CRI=(DSORG,EQ,UNKNOWN)
```

PSE Data Sets

Extended sequential data sets (PSE) are simply multi-volume sequential data sets that are managed by DFSMS. For obvious performance reasons, DFSMS typically allocates these data sets on different channel paths and even different control units if available.

Limitation

IBM has limited the number of volumes a PSE data set can span across to 16, and a limit of 123 extents on any given volume.

How CA Disk Supports PSE

To support PSE, use the PDSORG criteria parameter described in PDSORG – Partitioned data set organization in the chapter "Basic System - DMS PROC." This parameter gives a Data Center a convenient way to manage PSE data sets. CA Disk processes this data by the movement and management of physical data blocks. Data is processed at the physical block level allowing CA Disk to manage data without concern for source and target device dependencies.

IMS and DB2 Databases

CA Disk can process VSAM data sets created by IMS and DB2 database management software and maintain their integrity.

Note: If you use, or plan to use, the DFSMS/MVS Concurrent Copy function, see Processing CICS, DB2, and IMS for more information.

DBRC Support

CA Disk also has the ability to automatically detect that a data set is either part of a database under DBRC control, or that it is an image-copy data set that DBRC knows about. NOTIFY control statements can be produced such that DBRC can be informed of the CA Disk action that has taken place.

Note: DBRC Recon data sets cannot be migrated to tape using the Sequential Migrate function.

IDMS Support

Data sets within IDMS databases are marked as PS (physical sequential) with fixed format records in which the logical record length and block size are equal, and that they can be copied/moved like any other sequential data set. Users must, however, be aware of the interdependencies among the *dictionary*, *load*, and *data* data sets within IDMS databases, and keep them synchronized.

GDG Data Sets and Generation Wrapping

Catalog processing for GDGs turns on the Wrap Bit to create generations from 10,001 to 10,999. The Wrap Bit is then turned off to create generations 1000 and above. This is true for all supported releases of DFSMS.

The Archive, Restore, Delete, Move, Copy, and Sequential Migrate functions have the potential to catalog a GDG by its absolute name. Nearly all of the catalog actions for GDG data sets complete with no alteration of the sequence of generations. The original catalog status is maintained when using RECATMIG set to Y. When RECATMIG is not set to Y, only the active GDGs can be maintained (see sysparm GDGRECAT for other options). The GDG sphere sequence could also be altered incorrectly due to a Wrap Bit when a GDG is cataloged as a new entry.

The Wrap Bits rules Catalog processing uses are as follows:

- Only the generations in the range 0xxx can have the Wrap Bit on.
- The generations in the range 1001–8999 can never have the Wrap Bit on and do not cause the Wrap Bit to be turned on for new generations.
- If there are any generations in the range 9xxx, new generations in the range 0xxx can have the Wrap Bit turned on at creation time.
- If there are generations with the Wrap Bit on and other generations with the Wrap Bit off, creating generation 1000 fails .

- If all of the generations cataloged have the Wrap Bit on, uncataloging the last generation in the range 9xxx causes the Wrap Bits to be turned off, .
- Catalog processing strictly maintains the Wrap Bit. The Wrap Bit cannot be explicitly set, reset, or read.

Example:

- GDS List 1
 - LABS.DSN.LIST.G0002V00
 - LABS.DSN.LIST.G0003V00
 - LABS.DSN.LIST.G0005V00
 - LABS.DSN.LIST.G0099V00
 - LABS.DSN.LIST.G9005V00

All of these can be archived and recataloged without any problem.

- GDS List 2 SMS
 - LABS.DSN.LIST.G0002V00, Rolled off
 - LABS.DSN.LIST.G0003V00, Rolled off
 - LABS.DSN.LIST.G0004V00, Active
 - LABS.DSN.LIST.G0005V00, Active
 - LABS.DSN.LIST.G0006V00, Active
 - LABS.DSN.LIST.G0007V00, Active
 - LABS.DSN.LIST.G0008V00, Active
 - LABS.DSN.LIST.G0009V00, Deferred

All of these can be archived and recataloged without a problem while using RECATMIG Y. For RECATMIG N, the rolled off and deferred generations are handled as specified by the sysparm GDGRECAT.

- GDS List 3 Non-SMS
 - LABS.DSN.LIST.G0001V00
 - LABS.DSN.LIST.G0002V00
 - LABS.DSN.LIST.G0003V00
 - LABS.DSN.LIST.G0004V00
 - LABS.DSN.LIST.G0005V00
 - With LABS.DSN.LIST.G9999V00 uncataloged

If LABS.DSN.LIST.G9999V00 is archived and cataloged, it would become the most current generation regardless of the true sequence. See the GDGRECAT sysparm.

- GDS List 4
 - LABS.DSN.LIST.G0002V00 on volume TSO001
 - LABS.DSN.LIST.G0003V00 on volume TSO001
 - LABS.DSN.LIST.G0005V00 cataloged to ARCIVE
 - LABS.DSN.LIST.G0006V00 on volume TSO001
 - LABS.DSN.LIST.G0007V00 on volume TSO001
- LABS.DSN.LIST.G0005V00 can be safely restored.
- GDS List 5 where the GDG base is defined with a limit of 5
 - LABS.DSN.LIST.G0003V00 cataloged to MIGRAT
 - LABS.DSN.LIST.G0004V00 cataloged to MIGRAT
 - LABS.DSN.LIST.G0005V00 cataloged to MIGRAT
 - LABS.DSN.LIST.G0006V00 cataloged to MIGRAT
 - LABS.DSN.LIST.G0007V00 cataloged to MIGRAT
- List of DSNINDEX records
 - LABS.DSN.LIST.G0001V00, create date 01/01/12
 - LABS.DSN.LIST.G0002V00, create date 02/01/12
 - LABS.DSN.LIST.G0003V00, create date 03/01/12
 - LABS.DSN.LIST.G0004V00, create date 04/01/12
 - LABS.DSN.LIST.G0005V00, create date 05/01/12
 - LABS.DSN.LIST.G0006V00, create date 06/01/12
 - LABS.DSN.LIST.G0007V00, create date 07/01/12

Generations 0003 -0007 would be selected for recovery as the sphere is full.

Important! Review the RECATMIG, GDGRECAT, and ENQGDGBY sysparms. Because CA Disk uses absolute names to allocate and catalog GDGs, set the ENQGDGBY sysparm to N unless you are using JES3. To avoid potential data set loss, set the GDGRECAT sysparm to N. Review the use of CREATE for Restore. Create date is another factor that is used at restore time to determine generation sequence. If an older generation is restored with a new creation date and later re-archived, it can be selected for recovery out of sequence later.

RESTORE/RECOVER and Compression Products

If your system has a data compression product installed that compresses VSAM data sets it can be necessary to exempt CA Disk Restore and Recover processing from the data compression product's processing.

CA Disk reads VSAM data sets at the CI level when writing to the archive medium, for example, the data is read in a compressed format. However, during Restore processing, it writes to disk at the logical record level. This can cause certain data compression products to attempt to compress the data before writing it to disk. To prevent the data from being compressed twice, it is necessary to exclude CA Disk from processing by the data compression package.

Abnormalities Detected

The CA Disk functional capability detects and processes data sets with *external problems*, such as obsolete or unused data sets. The processing technique as described previous also allows detection of *internal problems* or *unusual* conditions within the data sets, disk packs or catalogs themselves. Historically, this second type of detection has not been available at all (or only after it was far too late — an invalid data set shuts down a monthly job six hours into the run, and all the track-image backups contain the same invalid track images). Consequently, users can be surprised by some of the diagnostic messages issued when CA Disk is first implemented.

If continued CA Disk processing does not hinder the integrity or recovery of the resource a message is issued and execution proceeds. If the condition invalidates the previous, messages are issued and the resource is bypassed if possible. Both of these cases provide the user with quick identification of problems, and hence the opportunity to make corrections in a more timely manner.

For example, if the directory of a PDS has been written over, default processing will issue messages to this effect and refuse to copy it to tape. Recovery of the valid backup copy of the data set can now be accomplished prior to its actual need, an ability that is usually greatly appreciated and much preferred over discovering too late that the backup copies are just as bad as the on-line data.

For more information, see Processing PDS that Contain Anomalies in the *Systems Guide*.

Following are some of the types of diagnostics that can be expected:

- Invalid PDS directories
- Wrong length records
- Multivolume data set CATALOG and VTOC disagreements
- The existence of user labels on disk data sets
- Direct access data sets that are marked as sequential

Password-Protected Data Sets

CA Disk supports the processing of password-protected data sets in all functions.

If the installation desires to allow CA Disk to process protected data sets, specify `sysparm PASSWORD` with a value of Y.

CA Disk cannot rename password-protected data sets. To override this feature, use the `sysparm PASSNEWN` as documented in `PASSNEWNn` in the *Systems Guide*.

When dealing with password-protected data sets, CA Disk does not take any action with the system password data set. It is the user's responsibility to take all necessary action to keep passwords correct and up to date.

Multivolume Data Sets

CA Disk backup and archive facilities for VSAM, sequential (PS), and direct (DA) data sets can process datasets that span up to 59 DASD volumes. CA Disk reports handle datasets on any number of volumes. All other functions that encounter multivolume datasets print an error that indicates that the dataset has been bypassed, and then go on to the next dataset.

Processing of multivolume datasets requires that the dataset be cataloged. The volume serial numbers found in the catalog determine the location of all the parts of the dataset. An improperly cataloged dataset can cause loss of data.

In implicit mode, CA Disk selects multivolume VSAM for processing whenever any component is found, even if the data and index components reside on different volumes. For multivolume non-VSAM, selection is made only when the first volume of the dataset is selected for processing.

For non-VSAM datasets, specify if multivolume datasets are processed by including `sysparm ARCMULTI` for explicit archives, or `RETMULTI` for implicit archive. `ARCMULTI` is referenced by the archive proc and `RETMULTI` is referenced by the retain proc. The `dms` proc does not need a `sysparm` to process archive multivolume. The default specification is N, which causes CA Disk to bypass archive or backup processing for all multivolume non-VSAM data sets.

CA Disk restores VSAM datasets to multiple volumes as needed, and non-VSAM datasets to either the first volume or to the original volume allocation depending on the value of sysparm PRIALLOC. If there are problems during restore, preallocating the dataset usually solves the problem.

Note: For information about preallocation, see the sysparm description PREALLOCn in the *Systems Guide* (non-VSAM), or VSPREDEFn (VSAM) in the *Systems Guide*.

If the restore does not provide the desired results, use the EXTRACT utility, documented in Archive Extract Utility in the *Systems Guide*, to restore the dataset to one or multiple volumes.

StorageTek 4400 Automated Cartridge System

CA Disk functions that use tape or cartridge devices are compatible with the ACS 4400 (Automated Cartridge System) from StorageTek, also referred to as *Nearline* storage. Devices under its control appear to CA Disk as any other 3480/3490-type devices. No changes to CA Disk procedures are necessary to use them.

Installations that have the 4400 or similar device installed will find the benefits of an automated tape cartridge handling system particularly valuable for such CA Disk functions as Archive and Backup, auto-restore, Sequential Migrate to Tape, and archive management functions.

StorageTek Redwood Device

The Redwood tape 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 distinct differences:

- The actual tape exits the cartridge from a different corner.
- 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 unit name. After 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, or MERGE processing.

For detailed instructions on how to activate this support, see Implementing Support for StorageTek Redwood in the *Systems Guide*.

User Data Set Integrity

To protect the integrity of your user data sets, before CA Disk processes any data set, CA Disk verifies whether data sets are in use by other tasks. This is done by the system enqueue service. This service works well in both a single CPU environment and a multiple CPU environment. However, the multi-CPU performs better if you have a cross-system enqueue package installed.

In multiple CPU environments with cross-system enqueue packages, you must ensure that all enqueues are propagated across all CPUs. For auto-restores in particular, enqueues for DMSAUT0, DMSAUT1, DMSAUT2, DMSAUT3, and DMSAUT4 must be propagated accurately. These enqueues ensure that data integrity is maintained.

If you have multiple CPUs but do not have a cross-system enqueue package installed, you must run CA Disk at times when the other CPU does not need access to those data sets, or provide integrity through other means.

Enqueues using the normal system enqueue service are not always propagated across all systems in a JES3 environment. CA Disk solves this exposure by enqueueing on data sets using SVC 99. Enqueues resulting from a SVC 99 request are propagated in a JES3 environment and provide a more reliable means of protecting data set integrity.

Note:JES3 customers must specify a Y for sysparm ENQGDGBY because the CA Disk JES3 enqueue allocation technique automatically causes an enqueue of the GDG base. For details, see the sysparm description for ENQGDGBYn in the *Systems Guide*.

For any function (such as Archive, Restore, Release and Move) that requires writing to, deleting or a catalog change to a data set, CA Disk will attempt to get an exclusive enqueue on the data set. If any other task has a shared (DISP=SHR) or exclusive (DISP=NEW, OLD or MOD), CA Disk assumes that someone else is using the data set, and CA Disk will bypass its use of the data set.

For any function (such as Backup and Copy) that requires reading a data set, CA Disk will attempt to get a shared enqueue on the data set. If other tasks have shared (DISP=SHR) enqueues, CA Disk will process the data set. If any other task has an exclusive (DISP=NEW, OLD or MOD) enqueue, CA Disk assumes that someone else is updating the data set, and CA Disk will bypass its use of the data set.

CA Disk Backup and Copy enqueueing techniques follow the standard linkage editor enqueue technique for load libraries and the ISPF enqueue technique for most other partitioned data sets.

The PDS Compress function provides an additional facility to protect user data sets in the shared DASD environment. It allows a command parameter (RESERVE) to be specified that causes CA Disk to lock out all users from access of the volume on which the object data set resides. Lockout occurs during processing for each data set. When this occurs, all tasks in all other machines that need to access the volume must wait for the compress to complete. Check with your installation's systems programming department before using this parameter, as it can severely degrade your machine throughput.

The VBACKUP and VRECOVER functions optionally issue a RESERVE and an ENQUEUE on the VTOC. This locks out all users on other machines from accessing the volume and it prevents other users on the same machine from acquiring space on the volume. Because of reduced opportunity for a data integrity exposure, CA Disk simulate mode and Report processing generally do not issue enqueues, and reports on data sets even if they are in use by other tasks.

For more information, see Processing Clusters in Use by Other Tasks in the chapter "RESTORE/RECOVER."

Using VBACKUP to Process Point-in-time DASD Volume Images

VBACKUP allows a backup of an offline DASD device containing a previously created point-in-time image of an online DASD volume. While VBACKUP is processing the offline DASD volume, the online volume from which the image was taken is available for normal use and updates. Since the offline image is actually read, changes to the online volume will have no effect on VBACKUP. VBACKUP supports point-in-time images created by, among others, IBM RVA with SNAPSHOT COPY feature, IBM 2105 with FLASHCOPY feature, StorageTek ICEBERG/SVA with SNAPSHOT COPY feature, EMC Timefinder BCV Volumes, and HDS ShadowImage feature. The ARCHVOLs created by VBACKUP while processing offline DASD volumes are supported by the VRECOVER command.

Using the MVS STOP command to gracefully terminate Restore type functions

The use of the MVS CANCEL command on Restore, Auto-restore, Recover and FMS Recover jobs has occasionally resulted in partial data sets being left. To prevent this, Restore type jobs will set the job noncancelable during the restore process. Instead of using the CANCEL command to terminate the job, it will be necessary to use the MVS STOP command. When the STOP command is issued to the Restore type job the restore function will be terminated and the data set in the process of being restored will be cleaned up according to normal CA Disk disposition rules for restore failures. Data sets that were cataloged to the pseudo-volume will be deleted and recataloged to the pseudo-volume. Data sets that were uncataloged at the start of the restore will be deleted and uncataloged. Predefined data set disposition will depend on the data set organization. Partitioned data sets (PDS) and partitioned data sets/extended (PDSE) that were being restored with the ERASE parameter specified will be deleted and uncataloged. PDS and PDSE data sets that were being restored without the ERASE parameter will be kept. Generation data sets (GDG) will be kept. All other predefined data sets will be deleted and uncataloged.

The format of the MVS STOP command, which can be abbreviated as P, is:

```
STOP jobname [.taskid][,A=asid]
```

Or:

```
STOP taskid[,A=asid]
```

If batch job DMSREST1 was running in ASID x'105' it can be stopped by:

```
P DMSREST1
```

Or:

```
P DMSREST1,A=105
```

Note: The ASID is in undelimited hexadecimal notation.

If a DMSAR started task was started with the command S DMSAR.DMSREST2 and was running in ASID x'107' it can be stopped by:

```
P DMSAR.DMSREST2
```

Or:

```
P DMSAR,A=107
```

Or:

P DMSREST2,A=107

Or if there were only one DMSAR active in the system:

P DMSAR

The ASID can be determined from a MVS display active of the job, which can be done by issuing the command:

D A,jobname

Concurrent Copy Support

Concurrent Copy is a function of DFSMS/MVS that eliminates prolonged outages of critical on-line data. The function lets you back up data while it is in use. Your data must be allocated on a DASD device attached to a 3990 Model 3 controller equipped with the Extended Platform feature, or a 3990 Model 6 controller. For specific information regarding the Concurrent Copy capabilities and how it operates, see the IBM DFSMSdss Program Manual.

CA Disk BACKUPCC Command

CA Disk provides support for Concurrent Copy using the DSCL BACKUPCC command. All CA Disk customers who want to use the CA Disk Concurrent Copy support, must have a license for DFSMSdss.

Even though CA Disk is using DFSMSdss data mover interfaces, the ARCHVOLS created through the Concurrent Copy support are in the same format as the normal CA Disk BACKUP command. However, CA Disk is limited to the capabilities provided by these data movers. We recommend that you use Concurrent Copy support only when concurrent backup is required. Use the normal DSCL BACKUP command whenever possible.

For more information about limitations, see "[Backup/Archive](#) (see page 233)."

Processing CICS, DB2, and IMS

Depending on the release you have installed, these products can already interface with Concurrent Copy directly. This interface establishes a synchronization point and automatically coordinates backup processes with Concurrent Copy. In summary, if any of these products automatically interface to an executing DFSMSdss Concurrent Copy job, they will work in the same manner as with CA Disk.

However, if the releases you have installed dynamically invoke DFSMSdss for Concurrent Copy support, the implementation method is not consistent with CA Disk. If this is the case, you cannot use the CA Disk Concurrent Copy support.

General Restrictions

Consider the following list of restrictions when planning for CA Disk. This list applies to the CA Disk product in general. Any specific function restrictions are documented with each function. This list helps you avoid problems caused when plans to process data do not include exception conditions.

- CA Disk never scratches data sets with the *SYS1* first-level identifier unless *SCRASYS1* was specified to allow it. This protects system data sets from accidental loss.

Note: CA Disk does not do any checking before taking catalog update actions for *SYS1* data sets.

- CA Disk does not back up all parts of SAS multi-volume data sets unless they were last processed by SAS release 9 or SAS release 8 at the current maintenance level.

Newer releases of SAS follow IBM standards for multi-volume data sets. *BACKUP*, *BACKUPCC*, *ARCHIVE* and *RESTORE* can process these data sets.
- Multivolume VSAM, PS, and DA datasets in the MVS environment can span up to 59 volumes for the archive, backup and restore functions.
- The following data set names are reserved for CA Disk internal use and are not be processed by *ARCHIVE* and *RESTORE* functions:

VTOC .xxxxxx

DMSOS .Vxxxxxx
- In all functions except Archive, Backup and Restore, data sets controlled by the UCC6 program product are automatically bypassed by CA Disk with an identifying message. Since CA Disk does not support the format unique to the UCC6 PDS, copying one with CA Disk converts it to a conventional PDS (for example, restoring one from a backup copy).
- The sysparm *UCC6CPDS* allows other CA Disk functions to process them.
- The CA Disk functions of Archive, PDS Compress, Move, and Copy do not process data sets that have been selected implicitly if they are included in the APF list or linklist. These data sets must be selected explicitly for processing.
- User labels on disk or tape volumes are not supported. Do not confuse user labels with *standard labels* that are present on most tape and DASD volumes. CA Disk issues messages to identify any data set containing user labels, and bypasses processing for the data set.
- Data sets marked as unmovable are normally skipped. CA Disk does not do absolute track allocation or use track-image processing. Both of these are needed to ensure the integrity of data sets marked as unmovable. Sysparms are available that permit CA Disk to copy (and relocate) the data sets.

- CA Disk does not support the PANEXEC program product data set load libraries. These data sets are formatted like a PDS but have no valid directory. CA Disk cannot determine their internal format.
- Exclude DBRC Recon data sets from the Sequential Migrate function.
- Back up CA VIEW databases as a unit and do not update during the backup process. If a database spans multiple volumes, perform full volume backups. If the database needs to be restored, restore all data sets comprising the database from the same backup.

CA Disk and DFSMS — Overview

Since its inception, CA Disk has given storage administrators versatile and flexible tools to manage the storage of their company's data. The fundamental tasks of data storage management include backup and recovery, archive, space release, and the movement of data at the volume and data set level. CA Disk has also provided extensive reporting capability, both on-line and batch, for determining the status of a data center's storage environment. These tools have kept pace over the years with new developments in hardware devices and with software systems, such as database management products.

In performing these storage management tasks, storage administrators have managed the storage of data *after the fact*. That is, decisions regarding the backup, archive, or movement of data were made after the data sets had been allocated, and were based on such criteria as the data set name, organization type, the age of the data set, its last reference date, the device type on which it is stored, and so on.

With the introduction of IBM's DFSMS, capabilities were made available for a storage administrator to set forth and administer a storage plan for each data set prior to data set allocation. This has moved the task of managing data storage into the realm of system-managed storage, a long-term objective of most data centers.

IBM's storage management subsystem (DFSMS) is designed to work in conjunction with software systems that perform the storage management tasks. The subsystem itself allows storage administrators to predefine a storage plan for data sets, based on such criteria as:

- Physical Characteristics
 - Organization type
 - Record length and format
 - Allocation attributes

- Storage Management Requirements
 - Retention or expiration periods
 - Space release constraints
 - Eligibility for archive or migration
 - Frequency of backup
 - Backup copy requirement
- Logical Storage Conditions
 - Multivolume restrictions
 - Availability constraints
 - Response requirements
 - Access requirements

In addition, storage administrators can define the physical storage conditions for volumes or groups of volumes in a system. These conditions include:

- Limits on data set size
- Migration limitations
- Allocation limitations
- Automatic backup requirements (data set level)
- Automatic backup requirements (volume level)

Users who are implementing part of their storage management plan through the facilities of SMS have found that converting their data center to system managed storage is a time-consuming task that requires careful planning and a commitment by management to the concept. Additionally, many types of data sets cannot be managed through the facilities of SMS and need traditional management tools for cost-effective results.

CA Disk has been developed to provide system managed storage for all data within an enterprise, whether under the control of SMS or not. Using a combination of the traditional rules for managing data, along with the SMS constructs where appropriate, storage administrators can use CA Disk to design sophisticated storage management schemes that will meet the needs of multiple end users, expedite routine application production processing, and slow the ever-increasing demand for additional costly hardware storage devices.

The Relationship Between CA Disk and DFSMS

The following paragraphs briefly describe DFSMS concepts and constructs, and how DFSMS and CA Disk work together. For more information about DFSMS concepts and constructs, see the IBM manual *MVS/ESA Storage Administration Reference*.

The DFSMS concept as developed by IBM uses ISMF, DFP 3.x, or DFSMS x.x as a base. The items that can attach to the base includes: a security package, a data manager, a data mover, and a sort program. A main component of IBM's DFP 3.x or DFSMS x.x is the Storage Management Subsystem (SMS). All attached items interact with DFP 3.x through the SMS. CA Disk, in the role of data manager and data mover, uses a System Program Interface (SPI) to interact with SMS.

SMS invokes Automatic Class Selection (ACS) routines during the creation of new data sets. SMS uses ACS routines to assign SMS constructs to a data set at allocation time. SMS constructs work as criteria for managing the data set during its life-cycle from initial creation to final disposition. The CA Disk SPI provides a way to request SMS construct information and apply ACS rules to allow CA Disk to manage and move data sets.

The four SMS constructs and their definitions are:

Data Class

A list of allocation attributes that the system uses for allocation of data sets.

Storage Class

A list of storage performance and availability attributes.

Management Class

A list of data set archive, backup, and retention attributes that can be used to manage storage at a data set level.

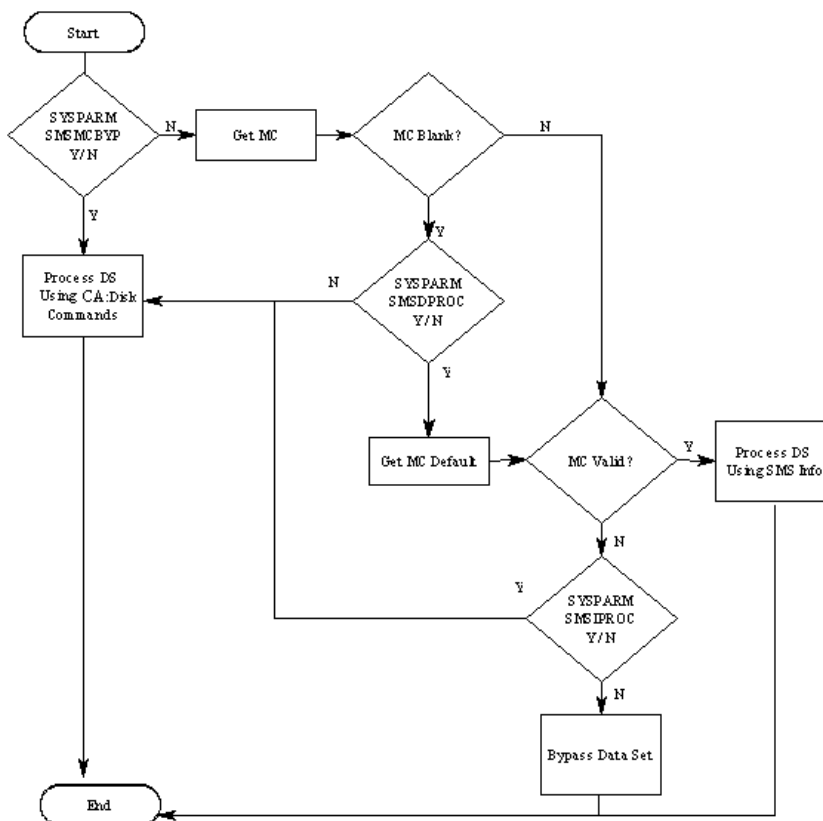
Storage Group

A list of real DASD volumes (pool), or a list of volumes that no longer reside on a system but end-users continue to reference (dummy).

DFSMS Hierarchy

The following flowchart reveals the logic flow used when CA Disk processes an SMS managed data set. It should assist you in understanding how the CA Disk SMS sysparms process an SMS managed data set.

The following illustration is an SMS Hierarchy Flow chart:



The storage group attributes are used only for the ARCHIVE, BACKUP and VBACKUP functions. The logic flowchart is based on a TRUE and FALSE condition with the TRUE condition being the first exception.

When an SMS managed data set is processed using management class or storage group attributes, the expiration date value in the DSNINDEX record is always set to 99365.

When an SMS managed data set is processed through DSCL bypassing management class or storage group attributes; the expiration date value in the DSNINDEX record is established in 1 of 3 ways:

- Specifying RETPD on the DSCL command
- Specifying EXPDT on the DSCL command
- Supplying a retention period to sysparm RETRETPD

This is also true for SMS managed data sets with no assigned management class.

When a data set is processed using SMS management class, CA Disk only uses the management class attributes. By default, the expiration date for all SMS managed backup and archive DSNINDEX records is set to 1999.365, but this does not mean they will never expire. Their actual expiration occurs as follows:

- For a backup DSNINDEX record — this type of record is deleted when other backups are made after the physical data set on DASD is modified, or after the data set is physically deleted from DASD and no archive copies exist. All of this is based upon the management class backup attributes.
- For an archive DSNINDEX record — this type of record is kept until the expiration attributes in the management class are met. Therefore, the 1999.365 is used as a place holder. Expiration of the most current cataloged Archived copy is based on two management class fields:
 - Expire days/date
 - Expire after days non-usage

For more information about this deletion process, see SMS Information Within The Archives in the chapter "FILES Maintenance." When an SMS data set is processed using CA Disk commands, any management class attributes assigned to the data set are ignored for the execution.

Finally, the IXMAINT & EXPIRE functions honor Management Class attributes as described in the *DFSMS Storage Administration Guide*. If you are processing GDG data sets, the NOLIMIT values should be fully understood because the action taken depends on the actual definition of the GDG Sphere, especially while you're using the SCRATCH or NOSCRATCH parameters.

Exploitation of DFSMS

The functional areas of CA Disk that support the SMS of DFP 3.x are shown below. Each function area contains an overview of the function, and where appropriate: the SMS fields supported, a construct indicator, the valid entries for that field, and a description of how CA Disk processes the entries. The construct indicators are as follows: MC, Management Class and SG, Storage Group. The storage class and data class need not be considered for this discussion because these classes have no data set management criteria. CA Disk understands 3 different media upon which data can reside (IBM storage states are shown in parentheses): primary (level 0), disk archive (level 1), and tape or other media (level 2).

SMS Field	CA Disk Functional Area
Auto-migrate (SG)	Archive (Implicit at volume level)
Auto-backup (SG)	Backup (Implicit at volume level)
Guaranteed Backup Frequency (SG)	Backup

SMS Field	CA Disk Functional Area
Auto-dump (SG)	Vbackup (Implicit)
Days Non-usage (MC)	Expire, IXMAINT
Expire After Date/Days (MC)	Expire, IXMAINT
Partial Release (MC)	Idle Space Release
Primary Days Non-usage (MC)	Archive
Level 1 Days Non-usage (MC)	Archive, Merge
Command or Auto Migrate (MC)	Archive (Implicit)
#GDG Elements On Primary (MC)	Archive
Rolled-off GDS Action (MC)	Archive, Expire
Backup Frequency (MC) Backup	Backup
Number of Backup Versions (MC) -data set Exists	IXMAINT
Number of Backup Versions (MC) -data set Deleted	IXMAINT
Retain Days Only Backup Version (MC) - data set Deleted	IXMAINT
Retain Days Extra Backup Versions (MC)	IXMAINT
Admin or User Command Backup (MC)	Backup
Auto-backup (MC)	Backup (Implicit)

Miscellaneous

The following system parameters were either introduced in Release 9.0 or have changed recently, and need to be reviewed to determine their relevance to various system environments. See the *Systems Guide* for the sysparm description.

ARBACKUP	DSCLRPLY	RECCHCAT	SMSFORCE
ARCNOSTD	DSSEIOTR	RESCRCATb	SMSGBRST
ARCLENQ	DSSMSGDD	RESCRNEX	SMSIPROC
ARDDNEXC	DSUCATLG	RESIXRPD	SMSLOWRC
ARDIAGNM	FMSSAFEX	SELECTAM	SMSMCBYP
ARESCRDT	IDRCRATE	SELECTDA	SMSGBYP
ARJOBEXC	IDRCSIZE	SELECTPO	SMSSTGAD

CARTCALC	IXMQTIME	SELECTPS	TABLESDD
CARTPRCT	IXMREVEX	SMSACSDF	USEDSSIO
CATBKSPC	MVDCATLG	SMSALLOC	VSUNOPEN
CCDRESET	OFFLDASD	SMSCACHE	XCPARCEX
CCINITEX	ONCCFAIL	SMSCPROC	XCPDSNEX
DATECUTO	POINTABS	SMSDPROC	

Diagnostics

To aid in the determination of why CA Disk acted in a particular way with SMS controlled data sets, decision making diagnostic messages are written to a DD statement called *SMSPRINT*. The descriptions of these messages can be found in *SMSPRINT* and *WHYFLAGS* Diagnostics in the chapter "FILES Maintenance." See this section when researching the validity of a CA Disk action. The *SMSPRINT* DD statement has already been placed in the following JCL Procedures.

- DMS
- IXMAINT
- MERGE

These diagnostic messages are also available while in simulate mode, and when included with the DATE= parameter, can provide examples of future processing if conditions remain unchanged.

Providing Class Names During Recovery

At RESTORE, RECOVER, FMS, MOVE and COPY time, CA Disk provides a way to supply SMS class parameters to the allocation process. This is accomplished using the following parameters:

STORCLAS=

Provide a Storage Class name which is defined to SMS.

MGMTCLAS=

Provide a Management Class name which is defined to SMS.

DATACLAS=

Provide a Data Class name which is defined to SMS.

Using these parameters, an SMS-managed data set can change its class names or a nonSMS-managed data set can be made to be SMS-managed by assigning it, at least, a Storage Class name. The following command, for example, will make a non-SMS data set SMS-managed:

```
RESTORE DSN=your.data.set.name,  
        STORCLAS=stornam
```

The following command will change the Management Class and blank out the Data Class of an SMS-managed data set:

```
RESTORE DSN=your.data.set.name,  
        MGMTCLAS=newname,  
        DATACLAS=
```

These parameters are only valid when the system parameter SMSALLOC is specified with a value of Y. For SMS-managed data sets, the class name(s) stored in the DSNINDEX record is passed to the allocation process if the STORCLAS=, MGMTCLASS=, or DATACLAS= parameters are not specified.

A complete description of these parameters can be found in the section Parameters Honored Without the Defer Parameter in the chapter "RESTORE/RECOVER."

Chapter 2: Basic System DMS PROC

This section contains the following topics:

[Commands Executed by the DMS JCL Procedure](#) (see page 59)

[Data Storage Command Language](#) (see page 60)

[SET Command](#) (see page 74)

[PASSWORD Command \(For VSAM Only\)](#) (see page 75)

[SCAN Commands](#) (see page 77)

[FIND Command](#) (see page 79)

[SELECT Command](#) (see page 82)

[EXCLUDE Command](#) (see page 119)

[ACTION Commands](#) (see page 123)

[DFSMS and DSCL](#) (see page 125)

[SMS Data Class Fields](#) (see page 126)

Commands Executed by the DMS JCL Procedure

The following are the commands that are executed by the DMS JCL procedure:

- ARCHIVE
- BACKUP
- BACKUPCC
- COPY
- DELETE
- EXPIRE
- MOVE
- REPORT
- RELEASE
- VBACKUP
- VREPORT

In this release, Data Storage Control Language (DSCL) is used by most functions of CA Disk.

Data Storage Command Language

The objective of the CA Disk Data Storage Command Language (DSCL) is to provide a rich, comprehensive palette of commands for efficient, common processing of mixtures of non-VSAM data sets and VSAM clusters.

Within a single execution you can report on, back up, archive, delete, expire, move, copy, and/or release idle space in non-VSAM data sets and VSAM clusters. You can do this based on scans of VTOCs of disk volumes, scans of ICF, VSAM or explicitly named OS CVOL catalogs, or the results of a catalog superlocate.

You also have a rich set of criteria to select data sets for and exclude data sets from processing as desired. In addition, a rich set of action parameters lets you control exactly what is done to these data sets.

CA Disk preprocesses your command stream to find any *overlapping requests*, that is, those that request processing of the same volumes or catalogs. Such requests are grouped together to allow all needed actions to be performed during a single pass of volume VTOCs, catalogs, or catalog superlocates.

The functions Sequential Migrate to Tape, Merge, Xcopy, Restore, and PDS Compression continue to use their own unique command sets. Functions using DSCL are invoked by executing the JCL procedure named DMS. In its simplest form, its use would look like:

```
//jobname JOB (acct,info),..
```

```
//STEP EXEC DMS  
          DSCL commands
```

Note: The FMS proc, used for data set recovery, is very similar to the DMS proc and its DSCL command structure. For more information, see Basic System ?FMS PROC the chapter "Basic System – FMS PROC."

To avoid unexpected execution of partially bad command strings, the DMS proc takes no action until all commands have been successfully understood, and all sysparm overrides have been validated.

To allow for sorted reports, the DMS proc has separate process, sort and print steps.

Coding Multiple Action Commands

Multiple action commands can be specified for a single candidate command, SCAN or FIND, to limit system overhead and do additional processing. However, care should be taken since all commands stacked together will use the information gathered at the time the candidate command is executed. Any modifications to the data set done by one command may or may not be seen by the other commands.

For instance, if a RELEASE command and a REPORT command are coded together, the report produced will contain the original or unreleased space information. Coding the RELEASE command before or after the REPORT command does not make a difference. If the data set was allocated with 100 tracks when the SCAN/FIND command processed it, and that same data set was using only 50 tracks after the RELEASE processing completed, the information that will appear on the report will be the original dataset allocation of 100 tracks.

DSCL Graceful Shutdown

The DSCLRPLY sysparm can be coded to gracefully shutdown, stop prematurely, any job using DSCL processing. This eliminates the dangers associated with forcing termination by canceling a job.

The DSCLRPLY sysparm causes a message to be issued to the operator at the start of the job that remains on the console for the duration of the job. If the operator replies END to this message, it indicates that processing is to be stopped and the job will terminate when the current object has finished.

Object

An object can be data sets, volumes, and/or catalogs, depending on the structure of the DSCL command.

After the operator replies to the messages, processing completes and all reports and statistics will be generated as normal.

```

16.58.54 JOB02303 $HASP373 TRAINERB STARTED - INIT 3 - CLASS B - SYS XE61
16.58.54 JOB02303 ACF9CCCD USERID TRAINER IS ASSIGNED TO THIS JOB - TRAINERB
16.58.54 JOB02303 IEF403I TRAINERB - STARTED - TIME=16.58.54
16.58.56 JOB03303 *122 4193 REPLY 'END' TO TERMINATE TRAINERB PROCESSING
16.58.58 JOB02303 *IEC501A M 353A,PRIVAT,SL,COMP,TRAINERB,DMS.BACKUP01,TRAINER.DMS C.32004338.T165858
16.58.58 JOB02303 *CT3001 IEC501A M
353A,PRIVAT,SL,COMP,TRAINERB,DMS.BACKUP01,TRAINER.DMS C.32004338.T165858
16.59.02 JOB02303 IEC705I TAPE ON
353A,104137,SL,COMP,TRAINERB,DMS.BACKUP01,TRAINER.DMS C.32004338.T165858,MEDIA2
17.00.53 JOB02303 IEC205I ARCHIVE0,TRAINERB,DMS,FILESEQ=0001, COMPLETE VOLUME LIST, 345
345
345
DIN=TRAINER.DMS C.32004338.T165858,VOLS=104137,
TOTALBLOCKS=22142
17.00.54 JOB02303 IEC502E K 353A,104137,SL,TRAINERB,DMS
17.00.54 JOB02303 CT3014 IEC502E K 353A,104137,SL,TRAINERB,DMS
17.00.54 JOB02303 *IEC501A M 353A,PRIVAT,SL,COMP,TRAINERB,DMS.BACKUP01,TRAINER.DMS C.32004338.T170054
17.00.54 JOB02303 *CT3001 IEC501A M
353A,PRIVAT,SL,COMP,TRAINERB,DMS.BACKUP01,TRAINER.DMS C.32004338.T170054
17.00.55 JOB03303 R 122,END
17.00.58 JOB02303 IEC705I TAPE ON
353A,104138,SL,COMP,TRAINERB,DMS.BACKUP01,TRAINER.DMS C.32004338.T170054,MEDIA2
17.01.15 JOB02303 IEC205I ARCHIVE0,TRAINERB,DMS,FILESEQ=0001, COMPLETE VOLUME LIST, 333
333
DIN=TRAINER.DMS C.32004338.T170054,VOLS=104138,TOTALBLOCKS=9804
17.01.15 JOB02303 IEF234E K 353A,104138,PVT,TRAINERB,DMS
17.01.15 JOB02303 CT3014 IEF234E K 353A,104138,PVT,TRAINERB,DMS
17.01.15 JOB02303 CAJR250I STEPFNAME STEP PROCSTEP C CODE EST-COST EXCP3 ELAPSED TOT-CPU
17.01.15 JOB02303 CAJR251I BACKUP01 1 DMS 0000 $205.62 25325 00:02:20.78 00:00:11.42
17.01.15 JOB02303 CAJR251I BACKUP01 2 AMS30RT FLUSH $ .00 0 00:00:00.00 00:00:00.00
17.01.15 JOB02303 CAJR251I BACKUP01 3 AMS30RT FLUSH $ .00 0 00:00:00.00 00:00:00.00
17.01.16 JOB02303 CAJR251I BACKUP01 4 SORTL 0000 $ .06 3 00:00:00.46 00:00:00.05
17.01.17 JOB02303 CAJR251I BACKUP01 5 PRINT 0000 $ .78 135 00:00:01.11 00:00:00.20
17.01.17 JOB02303 IEF404I TRAINERB - ENDED - TIME=17.01.17
17.01.17 JOB02303 CAJR252I JOB ENDED. TOTAL EST-COST $206.47 TOTAL CPU TIME 00:00:11.67
17.01.17 JOB02303 $HASP395 TRAINERB ENDED

```

The following illustration is an example of a DSCL Graceful Shutdown.

While reviewing this example, make note of the following time stamps:

- **16.58.56**—This is the outstanding message generated by using DSCLRPLY. This message remains on the console for the duration of the DSCL job.
- **17.00.55**—The operator replied, *END*. This instructs DSCL to begin termination. The elapsed time DSCL takes to complete the termination is variable, depending on the object being processed. If the objects are data sets, termination can be immediate. However, if the objects are volumes or catalogs, the elapsed time can be much longer.
- **17.01.15**—The graceful shutdown starts. Each job step completes as it normally would, including the PRINT step. The PRINT step produced all the normal reports for the data sets that were backed up.

For more information, see the sysparm description for DSCLRPLYn in the *Systems Guide*.

Condition Codes

The DMS proc returns the following JCL condition codes:

0

All actions were processed successfully and informational messages issued. Check to see if any messages are issued and if further action is required. Review the generated reports to make sure that the desired data sets were processed as stated.

4

Some minor problems were encountered in processing, but it was possible to continue. The results cannot be exactly what you wanted, but there seemed to be no harm in continuing. Warning messages were issued. Review these messages and the generated reports to make sure that the desired data sets were processed as stated.

Causes of this return code include selected data set in use by another task, insufficient access authority, I/O error, disagreements between the VTOC, VVDS and/or catalog for one or more individual user data sets, data sets found for processing but no processing is done because they did not meet the stated selection criteria, data set not processed because it is cataloged to the CA Disk pseudo volser, or the data set was not selected because it is restricted for the specified action command. The last case can occur when trying to select a user catalog with action commands other than *BACKUP* and *REPORT*.

8

A major problem was encountered in processing, resulting in CA Disk not even attempting some processing. Error messages were issued.

Causes of this return code include inability to mount a volume, inability to get the Format-4 DSCB for a volume, volume-oriented backup error, catalog scan errors, or no data sets can be found for processing.

12

The entire function, or a large portion of it, cannot be processed. Error messages were issued.

Causes of this return code include task abends, or data sets selected for processing but bypassed by the action being attempted.

16

Commands were in error. Error messages were issued and no action was taken.

If these condition codes do not suit you, you can install a user exit to modify these condition codes based on message numbers that are issued or other criteria. For details, see User-Specified Condition Codes, CCDRESETn, DSCLOWRCn, DSCLRPLYn, and SMSLOWRCn in the *Systems Guide*.

Messages

During DSCL SCAN REALVOLS and SCAN CATALOGS processing, CA Disk can issue error messages when *broken* data sets are encountered. If you do not want these messages to be issued, set the sysparm DSCLMSGs to N. This will suppress *broken* data set messages without affecting the job step return code. For more information, see the sysparm description for DSCLMSGs in the *Systems Guide*.

Commands

The DMS proc is controlled by commands provided to the SYSIN dd statement. The commands are of four main types:

Environment

Those commands that set up the environment in which CA Disk will run: SET and PASSWORD.

Candidate

Those commands that find candidate data sets: SCAN and FIND.

Selection

Those commands that select or exclude data sets to be processed based on your own specific criteria: SELECT and EXCLUDE.

Action

Commands that take action upon the selected resources: REPORT, VREPORT, BACKUP, BACKUPCC, ARCHIVE, DELETE, VBACKUP, RELEASE, MOVE, COPY, EXPIRE, and BILLING.

The minimum information required for you to execute DSCL is a SCAN or FIND command to name the volumes, catalogs or data set names to be processed, and an action command.

You can add other commands to gain additional selection or functionality.

DSCL Command Sequence

DSCL commands are listed below in their proper sequence. Each command is optional, except as described in the following table:

Type	Name	Parameters
Environment	SET	MODE=, SELDSORG=
Environment	PASSWORD (VSAM only)	CATALOGS=, MASTER=

Type	Name	Parameters
Candidate	SCAN	REALVOLS, SHOWVOL
Candidate	FIND	DSNAMES=, CATALOGS=, PASSWORD=, TABLE=
Candidate	SCAN	CATALOGS=, EXCCATS=, BACKUP
Selection	SELECT	DSNAMES=, COMPONENTS, TABLE=, VOLUMES=, VAMPOOL=, SGNAME=, VTOCS, CATALOGS=, CRITERIA=
Selection	EXCLUDE	DSNAMES=, COMPONENTS, TABLE=, VOLUMES=, VAMPOOL=, SGNAME=, VTOCS, CATALOGS=, CRITERIA=
Action	REPORT	CLDATA, CLDOPT=, CIARPT, CIAOPT=, DSINDX, DSUTIL, ISAMDS, MAPPER, MEMBER, MEMBERS=, MVDICT, PDSTAT, POCCHH, PODUMP, POTTTRD, POXREF, PO1REF, PO2REF, SVDICT, COMPONENTS, CMDGENDSN=, SPFDSNAME=, SPFRPTS=
Action	VREPORT	ALLOCS, ATTRB, DISTR, FREESP, LAYOUT
Action	BACKUP	RETPD=, EXPDT=, ACCESS=, RPT=CIA, CIAOPT=, NORESET, DEFER, NEWNAME=, NEWHLQ=, MEMBER=, SELTYPE=, MODE=
Action	BACKUPCC	RETPD=, EXPDT=
Action	ARCHIVE	RETPD=, EXPDT=, ACCESS=, RPT=CIA, CIAOPT=, DISP=, DEFER, NEWNAME=, NEWHLQ=, MEMBER=
Action	DELETE	DISP=
Action	EXPIRE	DATE=
Action	VBACKUP	RETPD=, EXPDT=, DSNENQ=, RESERVE, RPT=, LSTAR
Action	RELEASE	PCTUSED=, PCTSECONDARY=, CONVERT
Action	BILLING (VSAM only)	Action BILLING (VSAM only)
Action	COPY	ABSOLUTE, BLKSIZE=, CATALOGALL, CONTIG, CREATE, CYL, NEWNAME=, NEWVOLUMES, PREALLOC, REDUCE, TOVOL=, TRACK, VATHRESH=

Type	Name	Parameters
Action	MOVE	ABSOLUTE, BLKSIZE=, CATALOGALL, CONTIG, CREATE, CYL, FILL, NEWNAME=, NEWVOLUMES, PREALLOC, REDUCE, TOVOL=, TRACK, VATHRESH=,

SET and PASSWORD commands are optional. If you do specify a PASSWORD command, it applies to processing of VSAM password-protected clusters only.

Although the SCAN and FIND commands are each optional individually, you must specify at least one SCAN or FIND command before any action command.

SELECT commands are mandatory, unless you have sysparm SELECTRE specified with a non-default value of N. EXCLUDE commands are optional.

Although action commands are each optional individually, you must specify at least one action command for each SCAN/FIND group (defined later).

Since the purpose of each command is largely self-evident, general rules and guidelines for using these commands are described immediately following. Detailed explanations for most DSCL commands and parameters are presented later in this section. Detailed explanations for each of the action commands and parameters are presented in later sections.

SCAN and FIND Rules

You can specify any number of SCAN or FIND commands in a command stream. Multiple commands can also be intermixed for any given action command. CA Disk observes the following rules in processing SCAN and FIND commands. The rules are complex enough that you can try them out with SET MODE=SIMULATE until you get the feel of them.

Note: SET MODE=SIMULATE is not necessary (and will be ignored) while executing the REPORT or VREPORT commands, because no *action* is taking place.

SCAN and FIND Rule #1

DSCL processes the specified requests in terms of logical blocks termed *SCAN/FIND groups*. A SCAN/FIND group begins with a SCAN or FIND command and ends when the next non-consecutive SCAN or FIND command is encountered, or at the end of the input command stream.

At least one ACTION command must be specified in every SCAN/FIND group, with the one exception that SCAN CATALOGS=, BACKUP does not require an action command.

In its simplest form, a valid SCAN/FIND group looks like:

```
SCAN REALVOLS
  SELECT VOL=/,DSN=/
  REPORT ALLOC
```

This would cause CA Disk to scan all online volumes and produce a volume allocation summary report.

Another simple SCAN/FIND group:

```
FIND DSNAMES=LABS.TJP./
  REPORT MVDICT
```

would report on all data sets and clusters beginning with the prefix *LABS.TJP*.

While DSCL does not care about spacing, the indentation shown can help us in the examples that follow.

SCAN and FIND Rule #2

Consecutive SCAN and/or FIND commands apply to all subsequent selection and action commands, until another SCAN or FIND command is encountered.

For example, if you have multiple naming conventions, you can report on each of your data sets with:

```
FIND DSNAMES=SBTJP./
FIND DSNAMES=LABS.TJP./
  REPORT MVDICT
```

The report command would apply to each of the FIND commands.

To backup the data sets and clusters cataloged in three ICF, VSAM or OS CVOL catalogs, you can use:

```
SCAN CATALOGS=VOL05.USERCAT
SCAN CATALOGS=VOL00.USERCAT
SCAN CATALOGS=VOL05.TESTCAT
  SELECT DSN=/
  BACKUP
```

You can also combine FIND and SCAN commands, although this would result in extra overhead if the data set name on the FIND command is defined in the object of the SCAN:

```
SCAN CATALOGS=ICF.TESTCAT
FIND DSNAMES=LABS.*.TESTVSAM/
  SELECT DSN=/
  REPORT CLDATA
```

SCAN and FIND Rule #3

SELECT and/or EXCLUDE commands apply to all action commands that follow, until either a non-consecutive SELECT or EXCLUDE command is encountered, or until a SCAN or FIND command is encountered.

You can apply the preceding rule to one or more selection tests for multiple action commands. In the following example, the two catalogs named will be scanned and all data sets that meet either SELECT test will be both backed up and reported on:

```
SCAN CATALOGS=VOL05.USERCAT

SCAN CATALOGS=VOL00.USERCAT
  SELECT CRITERIA=(TRKS,GT,30),DSN=/
  SELECT CRITERIA=(ARCDT,LE,TODAY-30),DSN=/
    BACKUP
    REPORT MAPPER,CLDATA,SVDICT
```

In the following example, both catalogs will be scanned and those data sets that have not been modified within the last year will be archived. In addition, those data sets that use less than 25 percent of their allocated space will be reported on. Note that because the two SELECT commands are not consecutive, the first SELECT command has no effect on the ARCHIVE command. However, a data set can be selected by both action commands if it met both selection criteria:

```
SCAN CATALOGS=VOL05.USERCAT
SCAN CATALOGS=VOL00.USERCAT
  SELECT CRITERIA=(PCTUSED,LT,25),DSN=/
    REPORT MVDICT
  SELECT CRITERIA=(MODDT,LT,TODAY-36),DSN=/
    ARCHIVE
```

You can also combine FIND commands with SELECT and/or EXCLUDE commands:

```
FIND DSNAME=LABS.TJP.VSAMFIL
FIND DSNAME=LABS.DMS.VSAM/
  SELECT CRITERIA=(TRKS,GT,20),DSN=/
  EXCLUDE DSNAME=LABS.DMS.VSAMP/
    ARCHIVE
```

SCAN and FIND Rule #4

Any ACTION command applies to all SCAN and/or FIND commands in the current SCAN/FIND group.

This rule has already been demonstrated in previous examples. Whenever an action command is entered, the easiest way to determine which volume(s), catalog(s) and/or data set name(s) it will be processed against is to look up in the command stream and find the first SCAN or FIND command that precedes it. The real volumes, catalog, or data set name on that command, plus any others named on preceding consecutive SCAN and/or FIND command(s), will be affected by that ACTION command.

The examples have shown command input with only one SCAN/FIND group. Now let's look at an example with four SCAN/FIND groups. Remember that a SCAN/FIND group is detected by the presence of a non-consecutive SCAN or FIND command. In this example we scan all real volumes and all ICF and VSAM (but not OS CVOL) catalogs and report on all data sets created before January 16, 1996. In addition, another catalog is scanned to archive VSAM data sets that have not been modified in the last 30 days, but no report is needed for that catalog.

The final catalog is scanned and all data sets are reported on.

```
SCAN REALVOLS
SELECT CRITERIA=(CREDT,LT,1/16/96),DSN=/
      REPORT MVDICT
SCAN CATALOGS=/
      SELECT CRITERIA=(CREDT,LT,1/16/96),DSN=/
      REPORT MVDICT
SCAN CATALOGS=VOL00.TESTCAT
      SELECT CRITERIA=(MODDT,LT,TODAY-30),DSN=/
      ARCHIVE
SCAN CATALOGS=DMS.PRODUCT.CATALOG
      SELECT DSN=/
      REPORT CLDATA,CLDOPT=DIBEBBAC
```

SCAN and FIND Rule #5

Some identical actions such as Move can only be processed once in the same step with the same data set because of the internal structure of control blocks. Specified actions are processed in a logical order.

An incorrect example is:

```
SCAN REALVOLS SELECT CRI=(USED,LT,TODAY-15) ARCHIVE RELEASE REPORT MVD
```

Because CA Disk processes the commands in the order they are specified and the data set previous is archived before attempting an idle space release or a report. The correct syntax for the previous example is:

```
SCAN REALVOLS SELECT CRI=(USED,LT,TODAY-15) RELEASE ARCHIVE REPORT MVD  
(12/06/1996:GLH)
```

SELECT and EXCLUDE Rules

You can specify any number of SELECT and EXCLUDE commands in a command stream. Multiple commands can also be intermixed for any given action command. CA Disk observes the following rules in processing SELECT and EXCLUDE commands. Use SET MODE= SIMULATE until you understand the complexities of the rules.

SELECT and EXCLUDE Rule #1

For each SELECT or EXCLUDE command, all parameters specified must be true for the statement to be true. This is equivalent to AND selection based on the parameters on any individual SELECT or EXCLUDE commands.

For example,

```
SELECT VOLUMES=TSO/,CRITERIA=(DSORG,EQ,PS),DSN=/
```

would hold true only for sequential data sets on your TSO volumes, not for any other data set on your TSO volumes, nor for any sequential data sets on any other volume.

Note: SCAN CATALOGS or FIND commands compare the select volume list to the primary volume only for non-VSAM multivolume data sets.

SELECT and EXCLUDE Rule #2

If only SELECT commands apply to an action command, at least one SELECT statement must be true for the data set to be selected for CA Disk processing. This is equivalent to ORing selection based on combining SELECT or EXCLUDE commands.

For example,

```
SCAN REALVOLS  
  SELECT VOLUMES=TSO/,DSN=/  
  SELECT VOLUMES=WORK/,DSN=/  
  REPORT MVDICT
```

would report only on data sets on TSO and WORK volumes, not on any other volumes.

SELECT and EXCLUDE Rule #3

SELECT and/or EXCLUDE commands apply to all action commands that follow, until either a non-consecutive SELECT or EXCLUDE is encountered, or until a SCAN or FIND command is encountered. You may apply the preceding rule to one or more selection tests for multiple action commands. In the following example, the two catalogs named will be scanned and all data sets that meet either SELECT test will be both backed up and reported on:

```
SCAN CATALOG=VOL05.USERCAT
SCAN CATALOG=VOL00.USERCAT
  SELECT CRITERIA=(TRKS,GT,30),DSN=/
  SELECT CRITERIA=(ARCDT,LE,TODAY-30),DSN=/
    BACKUP
    REPORT MAPPER, CLDDATA, SVDICT
```

In the following example, both catalogs will be scanned and those data sets that have not been modified within the last year will be archived. In addition, those data sets that use less than 25 percent of their allocated space will be reported on. Note that because the two SELECT commands are not consecutive, the first SELECT command has no effect on the ARCHIVE command. However, a data set can be selected by both action commands if it met both selection criteria:

```
SCAN CATALOG=VOL05.USERCAT
SCAN CATALOG=VOL00.USERCAT
  SELECT CRITERIA=(PCTUSED,LT,25),DSN=/
    REPORT MVDICT
  SELECT CRITERIA=(MODDT,LT,TODAY-365),DSN=/
    ARCHIVE
```

You may also combine FIND commands with SELECT and/or EXCLUDE commands:

```
FIND DSNAMES=LABS.TJP.VSAMFIL
FIND DSNAMES=LABS.DMS.VSAM/
  SELECT CRITERIA=(TRKS,GT,20),DSN=/
  EXCLUDE DSNAMES=LABS.DMS.VSAMP/
    ARCHIVE
```

Additionally, you can use multiple SELECT statements with the FIND statements and the SELECT statements will be processed in a logical OR capacity. As an example:

```
FIND DSNAMES=LABS.TJP.VSAMFIL
  SELECT CRITERIA=(ARCDT,LE,TODAY-20),DSN=/
FIND DSNAMES=LABS.DMS.VSAM/
  SELECT CRITERIA=(TRKS,GT,20),DSN=/
  EXCLUDE DSNAMES=LABS.DMS.VSAMP/
    ARCHIVE
```

The previous example will process any data sets starting with LABS.DMS.VSAM and the data set LABS.TJP.VSDAMFIL if it is larger than 20 tracks OR if it has not been backed up in the last 20 days. Note that the order of the FIND and SELECT statements makes no difference as to how the criteria will be applied. The only thing that ends the group is the ACTION command.

SELECT and EXCLUDE Rule #4

VTOCs, and catalogs are not considered user data sets, and so are never processed unless specifically selected by SELECT VTOCS and SELECT CATALOGS=.

Consistent with rule 2 previous, if you specify SELECT VTOCS or SELECT CATALOGS= and you want to also process regular data sets as well, you must also supply a SELECT DSNAME=/ command.

For example,

```
SCAN REALVOLS
  SELECT CATALOGS=/
  BACKUP
```

will back up only your catalogs, but,

```
SCAN REALVOLS
  SELECT CATALOGS=/
  SELECT VTOCS
  SELECT DSNAME=/
  BACKUP
```

will back up your catalogs, VTOCs and data sets.

SELECT and EXCLUDE Rule #5

Consistent with rule 4 previous, if you set sysparm SELECTRE to N and specify neither SELECT commands nor EXCLUDE commands, CA Disk will process all user data sets, but neither VTOCS nor VTOC indexes nor catalogs nor VVDSs.

For example,

```
SCAN REALVOLS  
    BACKUP
```

will back up your user data sets, but not your VTOCS nor VTOC indexes nor catalogs nor VVDSs. But,

```
SCAN REALVOLS  
    SELECT CATALOGS=  
    SELECT VTOCS  
    SELECT DSNAMES=  
    BACKUP
```

will back up your catalogs, VTOCs and data sets.

SELECT and EXCLUDE Rule #6

If you use sysparm SELECTRE with its default value of Y, you must enter at least one SELECT command in each SCAN/FIND group.

For example,

```
SCAN REALVOLS  
    DELETE
```

would not be permitted.

SELECT and EXCLUDE Rule #7

If only EXCLUDE commands are specified for a given action command, if any EXCLUDE statement is met, the data set is bypassed. If no EXCLUDE commands test true, the data set will be processed.

For example,

```
SCAN REALVOLS  
    EXCLUDE VOLUMES=PROD/  
    EXCLUDE VOLUMES=SYS/  
    SELECT DSN=  
    REPORT MVDICT
```

would not report on your PROD or SYS volumes, only on your other volumes.

SELECT and EXCLUDE Rule #8

If both SELECT and EXCLUDE commands are specified for a given action command, the following rules apply:

- If an EXCLUDE command tests true, the data set is bypassed whether a SELECT statement tests true or not.
- If no EXCLUDE tests are met, at least one SELECT command must still test true for the data set to be processed.

For example,

```
SCAN REALVOLS
  EXCLUDE VOLUMES=PROD/
  EXCLUDE VOLUMES=SYS/
  SELECT CRITERIA=(DSORG,EQ,VSAM),DSN=/
  REPORT CLDATA
```

would report on VSAM clusters on other than your PROD or SYS volumes. It would not report on non-VSAM data sets, nor would it report on any VSAM clusters on your PROD or SYS volumes.

SET Command

Use the SET command to define environment values that will be in effect for the duration of this execution only. If used, you can put the SET command anywhere in the command stream before the first action command. For clarity, we recommend that you specify this command before all other commands.

As with all CA Disk commands, they can be commented out (nullified) by placing an asterisk * in column 1 of the line containing the command.

The syntax of the SET command is as follows:

```
SET MODE=,PRIMARY2,OFFLINE=,NOADMIN
```

MODE=

Specify SIMULATE or SIM to have CA Disk simulate the execution.

To run in LIVE mode, either comment out the SET MODE= command from the input stream, or delete the line entirely.

In general, in simulate mode CA Disk produces the normal messages and reports as if processing had taken place, but does not alter any data sets. For example, reports do not alter any data sets, so reports run equally well under simulate mode or live mode.

OFFLINE=

Specify INCLUDE or EXCLUDE. A value of INCLUDE will cause offline units specified in the UNIT= parameter of the SELECT command to be considered for VBACKUP. A value of EXCLUDE will prevent offline volumes from being processed by VBACKUP.

PRIMARY2

Specify PRIMARY2 or PRI to have CA Disk initiate the Dual Primary Volume process.

NOADMIN

Specify NOADMIN to bypass the use of Storage Administration FACILITY Class Profiles for this job step. All functions performed in this step will make use of the standard security access for the user submitting the job.

SET Command Example

The following SET command executes Dual Primary Volume processing in SIMULATE mode:

```
SET PRIMARY2,MODE=SIM
  SCAN REALVOLS`
  SELECT VOL=vvvvvv
  BACKUP RETPD=dd
```

The following SET command executes Dual Primary Volume processing in LIVE mode (comment out or delete MODE=SIM):

```
SET PRIMARY2
  SCAN REALVOLS
  SELECT VOL=vvvvvv
  BACKUP RETPD=dd
```

PASSWORD Command (For VSAM Only)

The PASSWORD command is used to provide CA Disk with any password necessary to carry out the functions requested in the command stream. Note that a catalog's master password is not required to scan a VSAM-password-protected catalog. Therefore, you need not specify any password to access a catalog. The proper password is required, however, to display passwords on the VSAM cluster data report, and also to archive or backup a cluster if it is password-protected. The command should precede the SCAN command that it is intended to cover. For clarity, we recommend that you specify this command just after the SET command, if any, and before all other commands.

The syntax of the PASSWORD command is as follows:

PASSWORD MASTER=,CATALOGS=

MASTER=

This REQUIRED parameter is used to supply CA Disk with the proper level of password it will need to carry out its requested functions. You can supply any of the following passwords:

- Catalog master password
- Cluster master password
- CA Disk Master password

The following rules govern when a password is required and the level of authority that is required:

If the catalog is not password-protected, no password is required (even if clusters within the catalog were defined with passwords).

If the REPORT function is being executed, no password is required, regardless of any catalog password protection. However, to display passwords on the cluster data report, the proper password must be specified. If either the catalog's master password or the CA Disk master password is specified, all passwords will be displayed for all clusters that are protected in the catalog. If a cluster password is entered, all clusters will be listed on the report, but the passwords will be suppressed for those clusters that have passwords, which do not match. If no password is supplied, no passwords will be displayed on the report.

If a BACKUP command is executed, for those clusters that are selected and are password-protected, either the catalog's master password, the CA Disk master password, or the cluster's read password must be specified. This same requirement applies for clusters that are being reported on in the cluster internal analysis report (CIA). In fact, you must supply the control password for the data set (if one exists) because CA Disk must read the cluster with control interval access to produce the report.

If ICF catalogs are to be backed up (exported), the highest level password for each catalog should be specified, or the CA Disk master password. If not properly specified, and unless *attempts=0* is in effect, the task will wait while the operator is prompted for the password.

If an ARCHIVE command is executed, for those clusters that are selected and are password-protected, either the catalog's master password, the CA Disk master password, or the cluster's master password must be specified.

CATALOGS=

Specify the catalog name for which the password value is to be used. If this parameter is omitted, CA Disk will use the password specified on the MASTER= parameter for all catalogs that are specified on SCAN commands for which no explicit password is specified via the PASSWORD command. If more than one PASSWORD command is specified that does not name a catalog explicitly, the most recent PASSWORD specified without a catalog name will take precedence for any succeeding catalogs named on a SCAN command that had no PASSWORD explicitly named for them. (See the example on Complex DSCL Command Sequences for further clarification of PASSWORD substitution rules.)

Because the password is specified on a separate command, it can be included from a different source than the remaining CA Disk commands. Creating a protected data set containing the PASSWORD command and concatenating this data set ahead of the typical CA Disk input set is one way to implement an easily maintained secured environment. When it becomes necessary to modify the password, simply replace the PASSWORD command also. This allows users to continue to process without knowledge of the required or changed security. CA Disk will suppress the printing of all passwords on the CMDPRINT output.

SCAN Commands

The SCAN command is used to tell CA Disk to scan the VTOCs of volumes, or to scan ICF, VSAM or explicitly named OS CVOL catalogs. A SCAN or FIND command must precede any action commands. If any SELECT and/or EXCLUDE commands are used, they must follow the SCAN command(s) and precede the action command(s). If you request a scan of a protected catalog and you desire to supply a password to CA Disk, a PASSWORD command should precede the SCAN CATALOGS= command.

On any given SCAN command, one and only one of the following parameters must be specified:

- REALVOLS
- CATALOGS=

These parameters tell CA Disk what type of scan is to be performed. With REALVOLS, the volume VTOCs will be scanned, whereas the CATALOGS= parameter will cause ICF, VSAM or explicitly named OS CVOL catalogs to be scanned.

SCAN REALVOLS Command

The syntax of the SCAN REALVOLS command is as follows:

SCAN REALVOLS,SHOWVOL

SCAN REALVOLS=

This parameter tells CA Disk to scan real, online volumes. It will cause CA Disk to process all real volumes in your installation that are currently available. The volumes to be scanned can be restricted by using the SELECT and/or EXCLUDE commands. More information regarding volume selection can be found in SELECT Command and EXCLUDE Command.

SHOWVOL=

Use this parameter to instruct CA Disk to issue an explicit message (2860) to inform you when it starts processing a DASD volume (that is, begins to scan the volume's VTOC). The message helps relate any subsequent errors or warning conditions that CA Disk can encounter to a specific DASD volume. The issuance of this message can be made automatic by specifying sysparm DLVOLMSG with a value of Y (the default value).

SCAN CATALOGS Command

The syntax of the SCAN CATALOGS command is as follows:

SCAN CATALOGS=,EXCCATS=,BACKUP

SCAN CATALOGS=

Specify a catalog name (or catalog pattern) to be scanned. The catalog(s) can be ICF, VSAM or an explicitly named OS CVOL. If an explicit catalog name is specified (for example, PROJ99.USERCAT), only that one catalog will be processed. However, if a catalog pattern is specified (for example, SYS1/), all catalogs that match the pattern and are connected to the master catalog will be processed. All standard CA Disk pattern matching capabilities are available (/ , ? , * and !).

Note: The pattern matching capability is not available in any operating environment that does not connect user catalogs to the master catalog. You can turn off this support by specifying sysparm VSCATPTN with a value of N, thereby forcing you to explicitly name a catalog. It should also be noted that the pattern matching capability does incur some additional overhead in that the master catalog must be searched in order to determine the user catalogs defined to the system. The master catalog will also be selected if it meets the specified pattern.

Normally, if a pattern name is specified that doesn't match any of the catalog names in your installation, CA Disk will terminate all commands in the command stream. This can be overridden by specifying sysparm VSCATPTN with a value of C, such that only the one SCAN command will fail and all other commands will be executed.

EXCCATS=

This parameter should be specified only when a catalog pattern is specified on the CATALOGS= parameter. Its function is to allow the user to exclude up to ten catalog names and/or patterns that would otherwise be selected based on the pattern supplied on the CATALOGS= parameter. See the example in Complex DSCL Command Sequences for a further explanation.

BACKUP

This parameter causes a backup copy to be taken of each ICF catalog selected for processing by the SCAN command, with the retention period provided by the RETRETPD sysparm. CA Disk uses the IDCAMS EXPORT utility to create the catalog backup, but then copies its ALIAS entries from the master catalog as part of the backup as well.

Do not confuse this BACKUP parameter (on the SCAN command) with the BACKUP command itself. This BACKUP parameter causes ICF catalogs to be backed up. The BACKUP command causes VSAM or non-VSAM data sets or ICF catalogs to be backed up. If you would like to specify a particular retention period or expiration date for the backup of an ICF catalog, specify SELECT CATALOGS= and BACKUP RETPD= or EXPDT= under any SCAN command.

For more information, see the section Year 2000 Considerations in the chapter "General Information."

FIND Command

The FIND command is used to specify a data set name or pattern to be processed. The function of the FIND command is to improve the performance of CA Disk when only one or a small subset of data sets is to be processed. Rather than scanning an entire catalog (as happens with the SCAN command), CA Disk will issue one superlocate for the data set name or pattern specified. Catalog management returns all data set name(s) matching the data set name specified in one inquiry, thereby reducing catalog management overhead for the job.

SELECT and EXCLUDE commands can be used with the FIND command, although this would be useful only if a pattern is specified for the data set name. If any SELECT and/or EXCLUDE commands are used, they must follow the FIND command and precede the action command(s) for the specified data set name/pattern. You can use this command in the same job stream as the SCAN command, although it would be duplicating effort if the data set name specified on the FIND command would otherwise be processed as a result of the SCAN command.

When should you use the FIND command? Any time that you are interested in processing one cataloged data set or cluster, a few cataloged data sets or clusters, or a group of cataloged data sets or clusters that share the same high level qualifier or qualifiers.

As an example of how the FIND and SCAN commands can accomplish the same objective, look at the following:

```
SCAN REALVOLS
  SELECT DSNAMES=LABS.TJP./
  ARCHIVE
SCAN CATALOGS=ICF.TESTCAT
  SELECT DSNAMES=LABS.TJP./
  ARCHIVE
FIND DSNAMES=LABS.TJP./
  ARCHIVE
```

In this instance, the FIND command would probably be a better choice because much of the overhead associated with scanning VTOCs or a VSAM catalog is eliminated. However, a SCAN command with the SELECT command will accomplish the same objective.

The FIND command searches ICF, VSAM and OS CVOL catalogs for entries. One disadvantage of using the FIND command is that catalog management will search only one catalog to find the data set name or pattern that you specify. In some situations, the catalog name should be specified on the FIND command so that catalog management can locate the data set. Two of these situations are: when the first-level node of the data set does not have an alias pointer in the master catalog, and when the data set is in the wrong catalog. Note that if the data sets you are attempting to process are uncataloged, you must change your command and use SCAN REALVOLS.

Processing Considerations

While the SCAN or FIND is processing, there is an exposure window between the time the data sets have been selected for processing and when they actually are backed-up or archived. It's possible that during this window, the data set can either *roll-off* the catalog (particularly in the case of a GDG) or be deleted by some other event. If this happens, CA Disk issues message 2664 which can be misleading in such cases.

Syntax

The syntax of the FIND command is as follows:

```
FIND DSN=, CATALOG=, PASSWORD=, TABLE=
```

DSNAME=

This parameter, REQUIRED only when the TABLE= parameter is not specified, is used to specify one explicit data set name or CA Disk pattern. This parameter is mutually exclusive with the TABLE= parameter. All standard CA Disk pattern matching capabilities are available (/ , ? , * and !).

If a pattern is specified, the first node of the data set name must be specified without pattern characters. This is a requirement of catalog management, since the high-level node will be used to determine what catalog to search. For example, *AB/* is not the same as *AB./*, even if *AB* is a complete node. The former will require much more memory because the *end of the node* is not indicated, which limits the catalog locate function.

Note: Specifying the first node of the data set name without pattern characters extends to multiple nodes as well, assuming MLA is in use. For example, in a nodes=2 MLA environment, the following is a valid command:

```
FIND DSN=node1.node2./
```

However, in the previous example, specifying the second dsn value as *node2/* will result in messages 2461 and 3262 being issued.

Users wanting to locate data sets with only a partial HLQ must use the SELECT TABLE= command on either a SCAN REALVOLS, or SCAN CATALOG.

CATALOG=

Use this OPTIONAL parameter to specify a catalog that catalog management should search to find the specified data set name or pattern. This parameter is required only if the data set name does not have an alias connector in the master catalog, or if the data set is in a catalog other than the one to which the alias points.

PASSWORD=

Use this OPTIONAL parameter when processing password-protected VSAM clusters to supply CA Disk with the proper level password it will need to carry out its requested functions. You can supply any of the following passwords:

Catalog master password

Cluster master password

CA Disk master password

See PASSWORD Command (For VSAM Only) to determine the level of password required for each function (such as REPORT and ARCHIVE). Note that this parameter must be specified even if the PASSWORD command is specified separately — the PASSWORD command is not used by the FIND command.

TABLE=

This parameter, REQUIRED only when the DSNAME= parameter is not specified, is used to specify the name of a member in the parmlib data set containing the list (table) of fully qualified data set names or patterns. If a pattern is specified, the first node of the data set name must be specified without pattern characters. Finally, this parameter is mutually exclusive with the DSNAME= parameter.

Note: Since SELECT and/or EXCLUDE commands can be combined with FIND TABLE=, data sets are not just automatically selected because they appear in this list. In the following example, the data set must both be listed in the parmlib member named *IMSDSNS* and must have been backed up within the last five days:

```
FIND TABLE=IMSDSNS
```

```
SELECT CRITERIA=(ARCDT,GE,TODAY-5)
```

Values specified in the TABLE= member have the same limitations as those for the DSNAME= parameter, described in DSNAME=.

There is not an exact limit to the number of entries that can be entered into a TABLE= member. It all depends on the environment. The information for the table is compiled and stored in a Variable Length table in memory above the 16M line. So, the maximum number of table entries is directly related to the amount of memory available to DSCL.

The format of the TABLE= member can be found under the topic Selection Lists —Generalized in the *Systems Guide*.

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 in the *Systems Guide*.

SELECT Command

The SELECT command allows a user to process only those data sets that satisfy the specified selection test(s). This command is required unless you have CA Disk sysparm SELECTRE specified with a value of N.

If a SELECT statement is specified, in order for a data set to be selected, ALL tests specified on a single SELECT command must be met. If only one parameter is specified, only that single test must be met for selection to take place.

For example, if the DSNAME= parameter or the VOLUMES= parameter, but not both, are specified on a single command, conceptually think *process only these data sets* or *process only these volumes*. If both the DSNAME= and VOLUMES= parameters are specified, conceptually think *process only these data sets if defined on one of these volumes*.

If multiple SELECT commands are specified for any given action command, this has the effect of a logical *OR* between SELECT commands. That is, if any SELECT command is true, the data set is processed, regardless of whether other SELECT commands test true or not (see EXCLUDE exception below).

Note: If an EXCLUDE command immediately precedes or follows any SELECT command, special rules apply. See SELECT and EXCLUDE Rules for a further explanation.

SELECT DSNAMES=, COMPONENTS, TABLE=, VOLUMES=, UNIT=, VAMPOOL=, SGNAME=, VTOCS, CATALOGS=, CRITERIA=

DSNAMES=

You can specify up to ten fully qualified names or CA Disk patterns.

The reason for this option is that the SELECT and EXCLUDE commands control the data sets that are selected. If any SELECT command is specified without the DSNAMES= parameter, all data sets are selected. If multiple SELECT commands are entered and you want them to all process the same data sets, the data set list must be copied to each SELECT, as in the following example:

```
SCAN REALVOLS
  SELECT DSN=(SYS1/,SYS2/),CRITERIA=(TRKS,GT,200)
  SELECT DSN=(SYS1/,SYS2/),CRITERIA=(EXTENTS,GT,3)
  REPORT SVDICT,MVDICT
```

Because a SELECT DSNAMES= statement is already very restrictive, you cannot specify the VTOCS, nor CATALOGS= parameters on the same SELECT statement. You can, however, add the COMPONENTS, TABLE=, VOL=, VAMPOOL=, SGNAME= or CRI= parameters.

For improved performance for VSAM while using the REPORT command, see the section COMPONENTS in the chapter "Reports."

COMPONENTS

Specifying this parameter can eliminate the need for CA Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet the SELECT DSNAMES= or TABLE= list. This saves significant processing time and also prevents many *broken* VSAM cluster messages (those that see components outside of the range of the DSNAMES= or TABLE= parameters).

This COMPONENTS parameter is only allowed on SELECT DSNAMES= or SELECT TABLE= commands.

This COMPONENTS parameter is ineffective under SCAN CATALOGS= or FIND commands and when checking fields that are not available within the FMT1 DSCB. This parameter should only be used with those SELECT CRITERIA fields that search the FMT1 information, NOT those parameters that review fields contained in the catalog or FILES, such as MODIFIED or ARCDT. The COMPONENT parm is intended for use mainly with the FMT1 DSCB date fields, (such as F1CREDIT, F1MODDT) or if the SELECT CRITERIA= parameter is not used.

Use this parameter if you have installed DFP release 3.1 or above, have installed a VSAM allocation manager product or have taken administrative action to ensure that VSAM components have names that relate to their associated cluster names, or when not processing VSAM data sets.

For example, to report on format-1 DSCBs starting with VSAM created today on your VSAM volumes:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SCAN REALVOLS  
SELECT DSNAMES=VSAM/,VOL=VSAM/,COMPONENTS,CRITERIA=(F1CREDIT,EQ,TODAY)  
REPORT MVD,COMPONENTS
```

TABLE=

Specify the name of a member in the parmlib data set containing the list (table) of fully qualified data set names or patterns.

Note: Data sets are not automatically selected because of their appearance in this list. If any other parameters are specified on the same SELECT command, all parameters must be true for the data set to be selected. For example, in the following command, the data set must both be listed in the member named *IMSDSNS* and must have been backed up within the last five days:

```
SELECT TABLE=IMSDSNS,CRITERIA=(ARCDT,GE,TODAY-5),DSN=
```

Because a SELECT TABLE= statement is already very restrictive, you cannot specify the VTOCS, nor CATALOGS= parameters on the same SELECT statement. You can, however, add the DSNAMES=, COMPONENTS, VOLUMES=, VAMPOOL=, SGNAME= or CRITERIA= parameters.

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 in the *Systems Guide*.

VOLUMES=

If you are running under the FIND command or you let CA Disk sysparm VOLSELRE default, you can optionally specify a list of volumes to be selected. If you are running under the SCAN command and have CA Disk sysparm VOLSELRE specified with a value of Y, you must specify one of the parameters SGNAME=, VAMPOOL=, UNIT=, or VOLUMES=.

You can specify up to 30 volumes or volume patterns.

The reason for this option is that the SELECT and EXCLUDE commands control the volumes that are scanned. If any SELECT command is specified without the VOLUMES= parameter or the UNIT= parameter, all volumes are scanned, regardless of any other SELECT commands. If multiple SELECT commands are entered and you want them to all process the same volumes, the volume list must be copied to each SELECT, as in the example below:

```
SCAN REALVOLS
SELECT VOL=(LABS/,WORK/),CRITERIA=(TRKS,GT,200),DSN=/
SELECT VOL=(LABS/,WORK/),CRITERIA=(EXTENTS,GT,3),DSN=/
REPORT SVDICT,MVDICT
```

You can add the DSNAMES=, COMPONENTS, TABLE=, VTOCS, CATALOGS= or CRITERIA= parameters on the same SELECT statement. VOLUMES= is mutually exclusive with the other volume-oriented SELECT command parameters of UNIT=, VAMPOOL= and SGNAME=. This means that VOLUMES= cannot be specified with UNIT=, VAMPOOL= or SGNAME= on the same SELECT statement.

UNIT=

If you are running under the FIND command or you let CA Disk sysparm VOLSELRE default, you can optionally specify a list of units to be selected. If you are running under the SCAN command and have CA Disk sysparm VOLSELRE specified with a value of Y, you must specify one of the parameters SGNAME=, VAMPOOL=, UNIT= or VOLUMES=. This parameter must also be used to specify the unit addresses for any offline volumes to be backed up by VBACKUP.

You can specify up to 16 unit addresses or patterns. The valid pattern characters for this parameter are '*', '?' and '/'. Leading zeroes in the unit number must be specified if pattern characters that can match multiple characters are used, for example: UNIT=50? will select units 0500 and 0501 where UNIT=50/ will select units 5000 and 50FF but not 0500 or 0501.

The reason for this option is that the SELECT and EXCLUDE commands control the volumes that are scanned. If any SELECT command is specified without the VOLUMES= or UNIT= parameter, all volumes are scanned, regardless of any other SELECT commands. If multiple SELECT commands are entered and you want them to all process the same units, the unit list must be copied to each SELECT, as in the example below:

```
SCAN REALVOLS
SELECT UNIT=(500,501,240?),CRITERIA=(TRKS,GT,200),DSN=/
SELECT UNIT=(500,501,240?),CRITERIA=(EXTENTS,GT,3),DSN=/
REPORT SVDICT,MVDICT
```

You can add the DSNAMES=, COMPONENTS, TABLE=, VTOCS, CATALOGS= or CRITERIA= parameters on the same SELECT statement. UNIT= is mutually exclusive with the other volume-oriented SELECT command parameters of VOLUMES=, VAMPOOL= and SGNAME=. This means that UNIT= cannot be specified with VOLUMES=, VAMPOOL= or SGNAME= on the same SELECT statement.

VAMPOOL=

This parameter is only valid when CA Allocate (Volume Allocation Manager) is active on the system. It allows selection at the CA Allocate pool level.

For example, `SELECT VAMPOOL=(SYSTEM,DMS*)` would cause CA Disk to check if the volume being processed or the cataloged volume of a data set being processed for selection is in the CA Allocate pool named SYSTEM or in pools whose names start with DMS.

Note: If you choose to use pattern matching for this parameter, you must use the CA Allocate pattern conventions. SMS managed volumes are not included for VAMPOOL. If you have an SMS volume that matches patterns in your VAMPOOL, they will be bypassed, as they are not part of the CA Allocate pool.

To accomplish this selection CA Disk searches the in-memory copy of the CA Allocate pools using the volume currently being processed by a `SCAN REALVOLS` or the volume to which a data set being processed by a `SCAN CATALOGS` command is cataloged. A table of pool names containing this volume is generated. The generated table of pool names is then compared to the pool names or patterns coded in the VAMPOOL parameter to determine if a match exists.

There is a limit of 30 on the number of pool names that can be placed in the generated table. If the volume used for the search is contained in more than 30 CA Allocate pools, the 31st and higher pool names will not be entered in the table and, therefore, will not be used in the compare for a match with the VAMPOOL parameter.

On each SELECT command statement, you can specify up to 22 CA Allocate pool names or CA Allocate pool name patterns.

There are four DSCL SELECT command volume-oriented parameters. They are `VOLUMES=`, `UNIT=`, `VAMPOOL=`, and `SGNAME=`, which are each mutually exclusive. This means that VAMPOOL= cannot be specified with VOLUMES=, UNIT= or SGNAME= on the same SELECT command statement.

SGNAME=

This parameter is only valid when IBM's Storage Management Subsystem (SMS) is active on the system. It allows selection at the DFSMS storage group level. Volumes and/or data sets will be selected for processing based on whether or not their volser belongs to a particular storage group name specified in the SGNAME= parameter.

For example, `SELECT SGNAME=(SGLARGE,SG1/)` would select all volumes and/or data sets that reside on the volumes, in the storage group of SGLARGE and in storage group names matching the pattern name of SG1/.

On each SELECT statement, you can specify up to 22 DFSMS storage group names or DFSMS storage group name patterns.

There are four DSCL SELECT command volume-oriented parameters. They are VOLUMES=, UNIT=, VAMPOOL=, and SGNAME=, which are each mutually exclusive. This means that SGNAME= cannot be specified with VOLUMES=, UNIT= or VAMPOOL= on the same SELECT command statement.

You can add the DSNAMES=, COMPONENTS, TABLE=, VTOCS, CATALOGS= or CRITERIA= parameters on the same SELECT statement.

VTOCS

Normally, DSCL processes user data sets only, and bypasses VTOCs, VTOC indexes, catalogs and VVDSs. Specify this parameter to have CA Disk process VTOCs and VTOC indexes. Because a SELECT VTOCS statement is already very restrictive, you cannot specify the DSNAMES=, COMPONENTS, TABLE=, CATALOGS= or CRITERIA= parameters on the same SELECT statement. You can, however, add a VOLUMES=, VAMPOOL= or SGNAME= parameter.

Note: The command *SCAN REALVOLS* is required for a SELECT VOL=/,VTOCS to backup vtocs.

CATALOGS=

Normally, DSCL processes user data sets only, and bypasses VTOCs, VTOC indexes, catalogs and VVDSs. Specify this parameter to have CA Disk process catalogs and VVDSs as specified by a fully qualified data set name or CA Disk pattern. Because a SELECT CATALOGS= statement is already very restrictive, you cannot specify the DSNAMES=, COMPONENTS, TABLE=, VTOCS nor CRITERIA= parameters on the same SELECT statement. You can, however, add a VOLUMES=, VAMPOOL= or SGNAME= parameter.

CRITERIA=

Specify the selection criteria to be met before any CA Disk action will occur. The criteria parameter is specified by a minimum of three subparameters as follows:

operand1,operator,operand2)

where the operands can consist of an expression as defined below. The maximum number of subparameters allowed is controlled by sysparm VSCRIMAX. Because a SELECT CRITERIA= statement is already very restrictive, you cannot specify the VTOCS nor CATALOGS= parameters on the same SELECT statement. You can, however, add the DSNAMES=, COMPONENTS, TABLE=, VOLUMES=, VAMPOOL= or SGNAME= parameters.

For logical operators, you can specify any of the following:

EQ

Equal to

NE

Not equal to

LT

Less than

LE

Less than or equal to

GT

Greater than

GE

Greater than or equal to

For operands, you can specify any of the following:

- Decimal number constants, such as 10 or 1000.
- Date constants in any acceptable CA Disk format, such as 96200, 96.200, 96/200, 19JUL96, 07/19/96, 1996200, 1996.200, 19JUL1996, 1996/200 or 07/19/1996. If CA Disk has trouble differentiating between a digit Julian date constant, such as 96200, and a decimal number constant, it will issue an appropriate message and ask you to use another format.
- Constant values TRUE, FALSE, YES, NO, ON or OFF. TRUE, YES, and ON are all-equivalent. FALSE, NO, and OFF are all-equivalent.
- Any of the DSCL select criteria field names listed below.
- Expressions that use addition (+) or subtraction (-) to combine a constant value with a field name or other constant value. For example, the expression TODAY-30 represents the date 30 days ago.

Note: SELECT CRITERIA only applies to data set level processing so it is not honored for volume processing as when command VREPORT is being used.

Criteria Operators and their Meanings

The following list shows the DSCL select criteria field name, listed alphabetically, and definition.

ALLOBLKS

Allocated in blocks

ALLOCYLS

Allocated in cylinders

ALLOPRI

Primary allocation amount

ALLOSEC

Secondary allocation amount

ALLOTRKS

Allocated in tracks

ARCDT

Last backup/archive date

BLKSZ

Block size

CASPLITS

Control Area splits

CATSTAT

Catalog status of the data set

CISPLITS

Control Interval splits

CREDT

Creation date

DC*

DFSMS Data Class field names

DSNTYPE

Data set name type

DSORG

Data set organization

EXPDT

Expiration date

F1CREDT

Format 1 DSCB creation date

F1EXPDT

Format 1 DSCB expiration date

F1MODBIT

F1 modification (change) bit

F1MODDT

Format 1 DSCB last mod

F1USEDT

Format 1 DSCB last use date

EXTENTS

Extents in the data set

GDG

Generation data group generation

GDGGEN

Relative generation number

GDGSTAT

Generation data group status

HWCOMPR

Hardware compressed data sets

IDLETRKS

Idle tracks in the data set

KBYTES

Kilobytes allocated to data set

LRECL

Logical record length

MC*

DFSMS Management Class field names

MODBIT

Modification (change) bit

MODDT

Last modification date

MODELDC

Model DSCB

MODIFIED

Modified since last backup

PASSWORD

Password-protected

PASSTYPE

Password protection type

PCTUSED

Percent used

PDSORG

Partitioned data set organization

PSORG

Physical Sequential data set organizations

RACFIND

RACF indicated

RECFM

Record format

SC*

DFSMS Storage Class field names

SG*

DFSMS Storage Group field names

SMSF1IND

F1DSCB SMS bit set

SMSIND

An SMS controlled data set

SVCMODE

SVC mode field for VTOC statistics

TEMP

System temporary non-VSAM data set

TODAY

Today's date

TRKS

Tracks allocated

USEDT

Last use date

UNMOVE

Unmovable attribute

VL*

DFSMS Volume field names

VOLCNT

Data set allocated volume count

VOLSEQNO

Volume sequence number for multivolume data sets

VSAMORG

VSAM organization type

ZFS

zSeries file system

Midnight "Rapid Aging" Problems

When CA Disk jobs that examine date fields are run very shortly after midnight, you must take into account the *rapid aging* that occurs at the stroke of midnight. Data sets that can have been created or used merely minutes ago suddenly appear as if the action took place a day ago!

When batch jobs create data sets, the create date used by the operating system is taken from the date the job started, not the date that each specific step of the job started. Therefore, data sets actually created in steps running after midnight, but part of a multistep job that began before midnight, will have create dates of the previous day — and immediately appear to be one day old!

Any selection tests based upon date values should take these two items into account, or many more data sets can be selected than were intended.

DSCL SELECT CRITERIA Field Name Descriptions

ALLOBLKS — Allocated in blocks (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of ALLOBLKS is set to YES/TRUE/ON if the DS1AVR bit is set and the DS1TRK bit is not set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOBLKS is set to YES/TRUE/ON if the DS1AVR bit is set and the DS1TRK bit is not set in the space options field SPACOPTN in the catalog entry for each cluster.

ALLOCYLS — Allocated in cylinders (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of ALLOCYLS is set to YES/TRUE/ON if the DS1CYL bits are set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOCYLS is set to YES/TRUE/ON if the DS1CYL bits are set in the space options field SPACOPTN in the catalog entry for each cluster.

ALLOPRI — Primary allocation amount (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of ALLOPRI is set to the primary allocation amount (like the 2 in *SPACE=(TRKS,2,1)*) as determined by the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOPRI is set to the primary space allocation (like the 2 in *TRACKS (2,1)*) field PRIMSPAC in the catalog entry for each cluster.

ALLOSEC — Secondary allocation amount (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of ALLOSEC is set to the secondary allocation amount (like the 1 in *SPACE=(TRKS,2,1)*) as determined by the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOSEC is set to the secondary space allocation (like the 1 in *TRACKS (2,1)*) field SCONSPAC in the catalog entry for each cluster.

ALLOTRKS — Allocated in tracks (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of ALLOTRKS is set to YES/TRUE/ON if the DS1TRK bit is set and the DS1CYL bits are not set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOTRKS is set to YES/TRUE/ON if the DS1TRK bit is set and the DS1CYL bits are not set in the space options field SPACOPTN in the catalog entry for each cluster.

ARCDT — Last BACKUP/ARCHIVE date (date format)

Comparison dates can be entered in any acceptable CA Disk format.

Inspect the CA Disk FILES for the most recent backup or archive for the data set or cluster to set the ARCDT field. If the CA Disk FILES does not have a record of a backup or archive, this field contains 00/00/0000, or 0.

For example, to select data sets never backed up by CA Disk, use:

```
SELECT CRITERIA=(ARCDT,EQ,0)
```

To select data sets and clusters not backed up in the last 28 days, use

```
SELECT CRITERIA=(ARCDT,LE,TODAY-28)
```

Note: This selects data sets and clusters never backed up as well! Also, because of the need to inspect the CA Disk FILES, using this parameter can cause CA Disk processing time to increase substantially.

Also see sysparm DSNDUPVL, by default, 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.

BLKSZ — Block size (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of BLKSZ is set to the value of the field DS1BLKL in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of BLKSZ is set to the physical block size field PHYBLKSZ or the CI size field AMDCINV in the catalog record for each VSAM cluster, depending on the value of sysparm VSUSEBLK. For more information, see the sysparm description for VSUSEBLK in the *Systems Guide*.

CASPLITS — Control Area splits (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of CASPLITS is set to 0. For all other processing of VSAM clusters, the value of CASPLITS is set to the Number of Control Area Splits field AMDNCAS in the catalog entry for each cluster.

CATSTAT — Catalog status of the data set

This field should be used only when running SCAN REALVOLS. The second operand in a CATSTAT comparison can be any of the following:

YES/TRUE/ON

Data set is properly cataloged.

NO/FALSE/OFF

Data set is not cataloged (except CATOTHER or WRONGVOL).

WRONGVOL

Data set is cataloged, but to a different volume.

CATOTHER

Catalog status cannot be determined for the data set due to an error, such as the volume with the catalog is not online.

For example, to select non-cataloged and miss-cataloged data sets, use:

```
SELECT CRITERIA=(CATSTAT,NE,YES)
```

Data sets with a CATOTHER status are never selected unless (CATSTAT,EQ,CATOTHER) is specified. This ensures that if a catalog is damaged or offline, data sets are not selected by any other combinations of the CATSTAT values. Sysres data sets that are cataloged to alternate sysres packs and show a value of ***** or &vvvvv in the volume field of an IDCAMS List Catalog are considered incorrectly cataloged and will be selected when using the NE operand in the criteria statement unless the WRONGVOL parameter is also coded.

For example, if the following statement is used to process data sets on a Sysres volume which contains data sets cataloged to an alternate sysres pack, all data sets with the ***** or &vvvvv value are selected as well as all of the un-cataloged data sets.

```
SELECT CRITERIA=(CATSTAT,NE,YES)
```

However, if the WRONGVOL parameter is also used, as shown in the next example, only the un-cataloged data sets are selected.

```
SELECT CRITERIA=(CATSTAT,NE,YES,AND,CATSTAT,NE,WRONGVOL)
```

Note: If you are not running a report that requires the catalog status to be checked, specifying this criterion can cause a significant increase in both elapsed and CPU times (since the catalog is checked for every data set).

CREDIT — Creation date (date format)

Comparison dates can be entered in any acceptable CA Disk format. For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of CREDIT is set to the value of the creation date field DS1CREDIT in the Format-1 DSCB for each data set or component. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For all other processing of VSAM clusters, the value of CREDIT is set to the creation date field DSETCRDT in the catalog entry for each cluster. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For example, to select data sets without creation dates, use:

```
SELECT CRITERIA=(CREDIT,EQ,0)
```

To select data sets and clusters with creation dates in the last 3 days, use `SELECT CRITERIA=(CREDIT,GE,TODAY-3)`. Note that this will NOT select data sets and clusters without creation dates!

EXPDT — Expiration date (date format)

Comparison dates can be entered in any acceptable CA Disk format.

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of EXPDT is set to the value of the expiration date field DS1EXPDT in the Format-1 DSCB for each data set or component. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For all other processing of VSAM clusters, the value of EXPDT is set to the expiration date field DSETEXDT in the catalog entry for each cluster. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For example, to select data sets with expiration dates, use:

```
SELECT CRITERIA=(EXPDT,NE,0)
```

To select data sets and clusters set to expire in the next week, use:

```
SELECT CRITERIA=(EXPDT,LT,TODAY+7)
```


F1CREDT — Format-1 DSCB create date (date format)

Comparison dates can be entered in any acceptable CA Disk format. For all data sets, the value of F1CREDT is set to the value of the creation date field DS1CREDT in the format-1 DSCB of each data set or VSAM component. Missing or invalid values are assigned a value of 00/00/0000 or 0.

Under *SCAN REALVOLS*, the F1CREDT selection criteria will give increased performance over the CREDT selection criteria due to eliminating the need for CA Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet this CREDT test.

The creation date in the format-1 DSCBs of components of VSAM clusters can provide different results compared to selection based on the creation date maintained in the cluster's catalog entry. Non-VSAM data sets are processed identically between CREDT and F1CREDT.

If you want to select VSAM clusters by the creation date in the cluster's catalog entry, while maintaining non-VSAM compatibility, use CREDT instead of F1CREDT.

This selection criteria is only valid when *SCAN REALVOLS* is used. It is ignored for *SCAN CAT*, unlike the CREDT selection criteria, which is valid for both. Another difference between F1CREDT and CREDT selection criteria is that F1CREDT can allow your CA Disk job to process more quickly. This faster processing occurs because F1CREDT is set up to allow CA Disk to bypass some catalog processing.

For more information, see Midnight "Rapid Aging" Problems.

F1MODBIT — Format-1 DSCB modification (change) bit (on/off)

For non-VSAM data sets and VSAM components, the value of F1MODBIT is set to YES/TRUE/ON if the change bit DS1IND02 is set in DS1DSIND of the Format 1 DSCB for each data set or component. This differs from MODBIT for multi-volume data sets in that MODBIT is a composite setting that is set ON if any volume has the modbit set for that data set. F1MODBIT is not a composite setting and varies for each volume of a multi-volume data set.

The MODBIT selection criteria is valid for both *SCAN REALVOLS* and *SCAN CAT* but the F1MODBIT is valid only when using *SCAN REALVOLS*. The use of F1MODBIT may allow your CA Disk job to process more quickly, since it allows CA Disk to bypass some catalog processing.

For example, for HSM conversions to select only those multi-volume data sets having the modbit set on for volume sequence number 01 use:

```
SCAN REALVOLS
SELECT VOL=(SMD0/),
      DSN=MULTIVOL.DATASET/,
      CRI=(VOLSEQNO,EQ,1,AND,F1MODBIT,EQ,ON)
```

Note: Certain applications bypass normal OPEN processing. Data sets updated by these applications will not have their change bits set. Thus, these data sets will never be selected by `SEL CRI=(F1MODBIT,EQ,YES)`. To select all potentially modified data sets, you must specifically select these data sets.

F1MODDT — Format-1 DSCB last modified date (date format)

Comparison dates can be entered in any acceptable CA Disk format.

For all data set processing, the value F1MODDT is set to the location indicated by the CA Disk system parameter DSCBLMOD in the format-1 DSCB of each data set or VSAM component. For more details, see the system parameter description for DSCBLMODnnnk in the *Systems Guide*. Missing or invalid values are assigned a value of 00/00/0000 or 0. CA Disk or IBM does not set the last modified date in the format-1 DSCB of VSAM index components.

Under *SCAN REALVOLS*, the F1MODDT selection criteria will give increased performance over the MODDT selection criteria due to eliminating the need for CA Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet this MODDT test.

The modification date in the format-1 DSCBs of components of VSAM clusters can provide different results compared to selection based on the modification date maintained in the cluster's catalog entry. Non-VSAM data sets are processed identically between MODDT and F1MODDT.

If you want to select VSAM clusters by the modification date in the cluster's catalog entry, while maintaining non-VSAM compatibility, use MODDT instead of F1MODDT. This selection criteria is only valid when using *SCAN REALVOLS*, it is ignored for *SCAN CAT*, unlike the MODDT selection criteria, which is valid for both. Another difference between F1MODDT and MODDT selection criteria is that F1MODDT can allow your CA Disk job to process more quickly. This is because F1MODDT is set up to allow CA Disk to bypass some catalog processing.

For more information, see Midnight "Rapid Aging" Problems.

F1USEDT — Format-1 DSCB last used date (date format)

Comparison dates can be entered in any acceptable CA Disk format. For all data set processing, the value F1USEDT is set to the location indicated by the CA Disk system parameter DSCBLUSD in the format-1 DSCB of each data set or VSAM data component. For more details, see the system parameter description for DSCBLUSDnnk in the *Systems Guide*. Missing or invalid values are assigned a value of 2155366. For more information, see Midnight "Rapid Aging" Problems. The last used date is not set in the format-1 DSCB of VSAM index components.

Note: There are restrictions involving special expiration date and Year 2000 processing. For details, see Year 2000.

Under *SCAN REALVOLS*, the F1USEDT selection criteria will give increased performance over the USEDTE selection criteria due to eliminating the need for CA Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet this USEDTE test.

This selection criteria is only valid when using *SCAN REALVOLS*, it is ignored for *SCAN CAT*, unlike the USEDTE selection criteria, which is valid for both. Another difference between F1USEDT and USEDTE selection criteria is that F1USEDT can allow your CA Disk job to process more quickly. This is because F1USEDT is set up to allow CA Disk to bypass some catalog processing.

For more information, see Midnight "Rapid Aging" Problems.

F1EXPDT — Format-1 DSCB expiration date (date format)

Comparison dates can be entered in any acceptable CA Disk format. For all processing of non-VSAM data sets and VSAM components with "SCAN REALVOLS", the value of F1EXPDT is set to the value of the expiration date field DS1EXPDT in the format-1 DSCB for each data set or VSAM component. Missing or invalid values are assigned a value of 00/00/0000 or 0.

This selection criteria is only valid when using *SCAN REALVOLS*, it is ignored for *SCAN CAT*, unlike the EXPDT selection criteria, which is valid for both. Another difference between F1EXPDT and EXPDT selection criteria is that F1EXPDT can allow your CA Disk job to process more quickly. This is because F1EXPDT is set up to allow CA Disk to bypass some catalog processing.

Note: Restrictions exist involving *special* expiration dates. See EXPDT—Expiration date (date format).

For more information, see Midnight "Rapid Aging" Problems.

CISPLITS — Control Interval splits (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of CISPLITS is set to 0. For all other processing of VSAM clusters, the value of CISPLITS is set to the Number of Control Interval Splits field AMDNCIS in the catalog entry for each cluster.

DSNTYPE — Data Set Name Type

The second operand in a DSNTYPE comparison can be any of the following:

LIBRARY

Partitioned data set extended (PDSE)

HFS

HFS data set

PDS

Partitioned data set (PDS)

LARGE

Large-format sequential data set

EXTENDED

Extended data set

DSORG — Data Set Organization

The second operand in a DSORG comparison can be any of the following:

AM

Access Method (VSAM)

DA

Direct access

INVALID

Invalid pattern of DSORG bits in format-1 DSCB

IS

Indexed Sequential (ISAM)

ISAM

Indexed Sequential (ISAM)

PDS

Partitioned

PO

Partitioned

SEQ

Sequential

UNKNOWN

No DSORG bits on in format-1 DSCB

VSAM

VSAM

For example, to select all partitioned and sequential data sets, use:

```
SELECT CRITERIA=(DSORG,EQ,PO,OR,DSORG,EQ,PS)
```

To select unmovable data sets by using the UNMOVE field. For example,

```
SELECT CRITERIA=(DSORG,EQ,PS,AND,UNMOVE,EQ,YES)
```

For similar fields, see VSAMORG—VSAM organization type, PDSORG—Partitioned data set organization, and PSORG—Physical Sequential Extended.

EXTENTS — extents in the data set (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of EXTENTS is set to the value of the DS1NOEPV field in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of EXTENTS is set to the Number of Extents field NOEXTNT in the catalog entry for the first data component of each cluster. It does not include the extents in the index component of any cluster. VSAM cluster reports, however, will display the sum of the two, which thus will be larger than the number on which selection was based.

For a similar field, see MODEL DSC—Is the data set a model DSCB? (yes/no).

GDG — Generation data group generation (yes/no)

For non-VSAM data sets, the value of this field is set to YES/TRUE/ON if the name represents, and is cataloged as, a generation data group generation.

For VSAM data sets, the value of GDG is always set to NO/FALSE/OFF.

GDGGEN — Relative generation number (number)

For non-GDG data sets (see the previous field GDG), the value of this field is set to 0. For GDG data sets, this field is set to the relative generation number as shown by the catalog.

For example, to select GDG generations older than the three most current versions (older than the -2, -1 and 0 generations) use:

```
SELECT CRITERIA=(GDGGEN,LE,-3)
```

Because non-GDGs are given a relative generation number of 0, to select the most current version of GDGs, use:

```
SELECT CRITERIA=(GDG,EQ,YES,AND,GDGGEN,EQ,0)
```

GDGSTAT — Generation data group status

The value of this field represents the status of a generation data group. The second operand in a GDGSTAT comparison can be any of the following:

YES/TRUE/ON

Active GDG

NO/FALSE/OFF

Not a GDG

ROLLOFF

Rolled off GDG

DEFERRED

Deferred GDG

For VSAM data sets, the value of GDGSTAT is always set to NO/FALSE/OFF.

HWCOMPR — Is the Data set Hardware Compressed (yes/no)

For this selection test to be true the format-1 DSCB DS1STRP and DS1COMPR bits must be set, that is, the data set is eligible for hardware compression.

For example, to select POE data sets that are hardware compressed, code:

```
SELECT CRITERIA=(PSORG,EQ,PSE,AND,HWCOMPR,EQ,YES)
```

To select VSAM data sets that are hardware compressed, code:

```
SELECT CRITERIA=(DSORG,EQ,VSAM,AND,HWCOMPR,EQ,YES)
```

Note: Hardware compression is only available for extended sequential and extended KSDS data set organizations.

IDLETRKS — Idle tracks (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with "SCAN REALVOLS, REPORT COMPONENTS", the value of IDLETRKS is set to the total number of tracks occupied by the data set or component as shown by its Format-1 and possible Format-3 DSCBs, less the value of the DS1LSTAR field in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of IDLETRKS is set to the number of tracks represented by the difference between the High Allocated RBA field HARBAS and the High Used RBA HURBAS in each cluster's catalog entry.

Note: Do not use this parameter for PDSEs in ESA 4.3 or higher environments because their DS1LSTAR is always zero.

KBYTES — Kilobytes allocated to the data set (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS, REPORT COMPONENTS*, the value of KBYTES is set to the total number of tracks occupied by each data set or component as shown by its format-1 and possible format-3 DSCBs, multiplied by the maximum block size for the device, and divided by 1000.

For all other processing of VSAM clusters, the value of KBYTES is set to the number of tracks represented by each of the extents listed in each cluster's catalog entry, multiplied by the maximum block size for the device, and divided by 1000.

LRECL — Logical record length (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of LRECL is set to the value of the logical record length field DS1LRECL in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of LRECL is set to the average logical record length field LRECL in the catalog entry for each cluster.

MODBIT — Modification (change) bit (on/off)

For all processing of non-VSAM data sets, ICF VSAM clusters, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of MODBIT is set to YES/TRUE/ON if the change bit DS1IND02 is set in the Format-1 DSCB for each data set or component. This is also true for the MODIFIED field.

Note: Certain applications bypass normal OPEN processing. Data sets updated by these applications will not have their change bits set. These data sets will never be selected by `SEL CRI=(MODBIT,EQ,YES)`. To select all potentially modified data sets, you must specifically select these data sets.

MODDT — Last modification date (date format)

Comparison dates can be entered in any acceptable CA Disk format. For non-VSAM data set processing and reporting on VSAM components with "SCAN REALVOLS, REPORT COMPONENTS", the value of MODDT is set to the location indicated by CA Disk system parameter DSCBLMOD in the Format-1 DSCB for each data set or component. For more details, see the system parameter description for DSCBLMODnnnk in the *Systems Guide*. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For all other processing of VSAM clusters, the value of MODDT is set to the last modified date in field AMDSTSP in the catalog entry for each cluster. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For example, to select data sets without last modification dates, use:

```
SELECT CRITERIA=(MODDT,EQ,0)
```

To select data sets and clusters modified in the last day, use

```
SELECT CRITERIA=(MODDT,GE,TODAY)
```

Note: This will NOT select data sets and clusters without last modification dates.

HFS datasets, DB2 and certain other applications bypass normal open processing and therefore do not maintain a last modified date. Thus, these data sets will never be selected by the previous selection criteria.

For improved performance, see F1MODDT—Format-1 DSCB last modified date.

For more information, see Midnight "Rapid Aging" Problems.

MODELDC — Is the data set a model DSCB? (yes/no)

For non-VSAM data sets, the value of MODELDC is set to YES/TRUE/ON if the value of the DS1NOEPV field is 0 and the value of the DS1SCAL3 field is 0 in the Format-1 DSCB for the data set.

For VSAM data sets, the value of MODELDC is always set to NO/FALSE/OFF.

For a similar field, see EXTENTS—extents in the data set (number).

MODIFIED — Modified since last backup (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of MODIFIED is set to YES/TRUE/ON if the change bit DS1IND02 is set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of MODIFIED is set to YES/TRUE/ON if the CA Disk Files data set shows that this cluster has never been backed up or archived by CA Disk, or if the last modified date and time in field AMDSTSP in the catalog entry for each cluster are more recent than the last CA Disk backup or archive of the clusters, or if the VSAM cluster open flag in catalog field OPENIND is on. The OPENIND flag is set ON whenever a VSAM cluster is open and set OFF when it is closed. If the cluster is not closed normally (that is, when a system crash occurs), then its flag is left on. Even if the comparison of last modified date/time and last backup date/time indicates the data set has not been modified since its last backup, the OPENIND flag test will indicate that MODIFIED is YES/TRUE/ON.

Note: If a VSAM cluster is currently in use, normal system enqueues will ensure that CA Disk does not get access to the data set when the OPENIND is ON.

Because CA Disk must check the FILES for each cluster, using this parameter with VSAM clusters can increase processing time.

Note: Certain applications bypass normal OPEN processing. Non-VSAM data sets updated by these applications will not have their change bits on. The following SELECT statement will never select these data sets:

```
SELECT CRITERIA=(MODIFIED,EQ,YES)
```

To select all potentially modified data sets, you must specifically select these data sets.

VSAM clusters used by these applications (that is, DB2) will have a last modified date of 00/00/0000 or 0. In this case, the last use date for the primary data component will be used in comparison with the date of the last backup. The change bit for the primary data component must also be on.

PASSTYPE — Password protection type

For all processing of non-VSAM data sets, and reporting on VSAM components with "SCAN REALVOLS, REPORT COMPONENTS", the value of PASSWORD is set to READ or WRITE depending on the values of the DS1IND10 and DS1IND04 bits in the Format-1 DSCB for each data set or component.

Note: Passwords are not supported on DFSMS-controlled volumes.

For all other processing of VSAM clusters, the value of PASSWORD is set to READ, WRITE, CINV or MSTR depending on the value of the password field PASSWALL in each cluster's catalog entry.

The second operand in a PASSTYPE comparison can be:

READ

Read password required

WRITE

Write password required

CINV

Control interval password (VSAM only)

MSTR

Master level password required (VSAM only)

For example, to select all write-password-protected data sets, use:

```
SELECT CRITERIA=(PASSTYPE,EQ,WRITE)
```

Greater than and *less than* type comparisons can also be made, based upon read-level password authority being less than write, write less than control interval, and control interval less than master. For example, to select all data sets with write, control interval, or master level password protection, use:

```
SELECT CRITERIA=(PASSTYPE,GE,WRITE)
```

PASSWORD — Password-protected (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of PASSWORD is set to YES/TRUE/ON if the DS1IND10 bit is set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of PASSWORD is set to YES/TRUE/ON if the password field PASSWALL in each cluster's catalog entry shows that a read, write, control or master password exists.

PCTUSED — Percent used (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of PCTUSED is set to 0 if the DS1LSTAR field in the Format-1 DSCB for each data set or component is set to 0. If the DS1LSTAR field is not 0, this PCTUSED is set to the number of tracks in the DS1LSTAR divided by the tracks allocated to the data set, expressed as a percentage with a minimum of 1 and a maximum of 100.

This means that a 1,000 track non-VSAM data set that uses 1 track would have a PCTUSED of 1, not 0. If it were truly empty (a zero DS1LSTAR), it would have a PCTUSED of 0. Beware of data sets used by applications that do not maintain the DS1LSTAR field.

Note: PO data sets that have a member count of zero will not be selected, even if you specify *PCTUSED,EQ,0*, because the directory information makes the data set appear partially used.

VSAM components never appear to be empty.

For all other processing of VSAM clusters, the value of PCTUSED is set to 0 if the value of the High Used RBA field HURBADS in each cluster's catalog entry is 0. If the HURBA is not 0, this PCTUSED is set to the High Used RBA HURBADS divided by the High Allocated RBA HARBADS, expressed as a percentage with a minimum of 1 and a maximum of 100.

This means that a cluster that is nearly empty, or which has had records inserted and subsequently deleted, would have a PCTUSED of 1, not 0. If it were truly empty (a zero HURBADS), it would have a PCTUSED of 0.

PDSORG — Partitioned data set organization

Possible values for the second operand in a PDSORG comparison are:

HFS

HFS data set

PDS

Partitioned

PO

Partitioned

PDSE

Partitioned extended

POE

Partitioned extended

For example, to select all PDSEs, use:

```
SELECT CRITERIA=(PDSORG,EQ,POE)
```

or

```
SELECT CRITERIA=(PDSORG,EQ,PDSE)
```

To select all PDSs, use:

```
SELECT CRITERIA=(PDSORG,EQ,PO)
```

or

```
SELECT CRITERIA=(PDSORG,EQ,PDS)
```

PSORG — Physical Sequential Extended

Possible values for the second operand in a PSORG comparison are:

PS

Sequential

PSE

Sequential Extended (Striped data set)

SEQ

Sequential

SEQE

Sequential Extended (Striped data set)

For example, to select all PSEs, use:

```
SELECT CRITERIA=(PSORG,EQ,PSE)
      or
```

```
SELECT CRITERIA=(PSORG,EQ,SEQE)
```

To select all PS data sets, use:

```
SELECT CRITERIA=(PSORG,EQ,PS)
      or
```

```
SELECT CRITERIA=(PSORG,EQ,SEQ)
```

RACFIND — RACF-indicated (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of RACFIND is set to YES/TRUE/ON if the DS1IND40 bit is set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of RACFIND is set to YES/TRUE/ON if the x'80' bit is set in the RACF security attributes field SECFLAGS in each cluster's catalog entry.

RECFM — Record format

The second operand in a RECFM comparison can be any of the following:

F

Fixed-length records

FB

Fixed blocked records

FBA

Fixed blocked with ASA carriage control

FBM

Fixed blocked with machine carriage control

U

Undefined length records

V

Variable-length records

VB

Variable blocked records

VBA

Variable blocked with ASA carriage control

VBM

Variable blocked with machine control

This field requires an exact match, that is, `SEL CRI=(RECFM,EQ,FB)` selects only FB data sets, not F or FBA, and so on.

If you need to select data sets with a particular RECFM attribute but you don't care what other attributes it is combined with, use the following fields and simply test for the condition being present:

RECFMF

Fixed attribute (yes or no)

RECFMV

Variable attribute (yes or no)

RECFMU

Undefined attribute (yes or no)

RECFMS

Standard or Spanned attribute (yes or no)

RECFMT

Track overflow attribute (yes or no)

RECFMB

Blocked attribute (yes or no)

RECFMA

ASA carriage attribute (yes or no)

RECFMM

Machine carriage attribute (yes or no)

For example, `SELECT CRITERIA=(RECFMB,EQ,YES)` selects all blocked data sets, regardless if fixed or variable, and so on.

`SELECT CRITERIA=(RECFMV,EQ,YES,AND,RECFMS,EQ,YES)` would select all variable-spanned data sets, regardless of whether they are blocked or not.

SMSF1IND — F1DSCB SMS bit (yes/no)

If the SMS bit is set in the format-1 DSCB on the VTOC, this field will be equal to yes. Specifying this criterion parameter will cause fewer overheads than if you were to specify SMSIND, because SMSIND must verify the catalog.

SMSIND — SMS-controlled data set (yes/no)

For this field to be set to yes the format-1 DSCB SMS bit must be set and the data set must be cataloged correctly as an SMS data set (that is, a storage class name must exist).

SVCMODE — SVC mode field for VTOC statistics (number)

The value of SVCMODE is set to the value of DS1SVCMD, at the offset defined by sysparm DSCBSVMD, within the format-1 DSCB for the data set. This field is maintained in the format-1 DSCB by the CA Disk Open SVC as described in The CA Disk SVC in the *Installation Guide*.

TEMP — System temporary non-VSAM data set (yes/no)

For non-VSAM data sets, the value of this field is set to YES/TRUE/ON if the data set name matches CA Disk pattern name SYS?????.T?????.?????. or ****SYSUT** followed by anything.

For VSAM data sets, the value of TEMP is always set to NO/FALSE/OFF.

TODAY — Today's date (date format)

Comparison dates can be entered in any acceptable CA Disk format.

This value is derived from the system clock.

TRKS — Tracks allocated

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of TRKS is set to the total number of tracks occupied by each data set or component as shown by its format-1 and possible format-3 DSCBs.

For all other processing of VSAM clusters, the value of TRKS is set to the number of tracks represented by each of the extents listed in each cluster's catalog entry.

UNMOVE — Unmovable attribute (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with "SCAN REALVOLS, REPORT COMPONENTS", the value of UNMOVE is set to YES/TRUE/ON if the unmovable attribute bit DS1DSGU is set in the Format-1 DSCB for each data set or component. For more information, see DSORG—Data set Organization.

For all other processing of VSAM clusters, the value of UNMOVE is always set to NO/FALSE/OFF.

USEDT — Last Use date (date format)

Comparison dates can be entered in any acceptable CA Disk format. For non-VSAM data set processing, and reporting on VSAM components with *SCAN REALVOLS, REPORT COMPONENTS*, the value of USEDT is set to the location indicated by CA Disk system parameter DSCBLUSD in the Format-1 DSCB for each data set or component. For more details, see the system parameter description for DSCBLUSDnnnk in the *Systems Guide*.

For all other processing of VSAM clusters and its components, the value of USEDT is set to the location pointed to by CA Disk sysparm DSCBLUSD in the Format-1 DSCB for the first data component of each cluster. But this field is filled in only for clusters cataloged in ICF catalogs. Consult the appropriate personnel at your own installation to determine if last use dates are being maintained by either IBM routines or the enhanced routines supplied with CA Disk. If they are not, you should not use this field name for VSAM clusters.

To ensure that data sets without a last use date do not get archived when last use criteria are specified, invalid values are assigned a value of 2155366, a high date. To maintain compatibility, the old invalid date of 31DEC1999 or 99365 will be accepted until the cutoff date of 01JAN1999 or 99.001. The 99365 date will be converted to 2155366 if it is the object of USEDT or F1USEDT comparisons. The NEW invalid USEDT or F1USEDT value is 0, *zero*, or 2155366 or 2155.366. Either of these values can be used now to search for zero and invalid USEDTs.

Note: We recommend conversion begin immediately, so that you can begin treating USEDTs of 99365 as *real* beginning on 99.001. This will also assist you during testing with IPL dates GE 99.001.

For example, to select data sets with zero and invalid last use dates, use:

```
SELECT CRITERIA=(USED, EQ, 31DEC1999)
```

Works until IPL dates GE 99.001.

```
SELECT CRITERIA=(USED, EQ, 0)
```

Works now and in future.

```
SELECT CRITERIA=(USED, EQ, 2155.366)
```

Works now and in future.

To select data sets and clusters cataloged in ICF catalogs with date stamping turned on but not used in the last 30 days, use:

```
SELECT CRITERIA=(USED, LE, TODAY-30)
```

Note: This will NOT select data sets and clusters without last use dates.

For improved performance, see F1USED—Format-1 DSCB last used date (date format).

For more information, see Midnight "Rapid Aging" Problems.

VOLCNT — Data set allocated volume count (number)

The value of VOLCNT is set to the number of allocated volumes for the data set. For non-VSAM cataloged data sets the volume count is retrieved from the catalog. The VOLCNT value is set to 1 for non-VSAM uncataloged data sets. The volume count for the VSAM cluster is retrieved from the catalog for each VSAM data set.

VOLSEQNO — Multivolume data set Volume Sequence Number

For all processing of non-VSAM data sets, and reporting on VSAM components with *SCAN REALVOLS*, *REPORT COMPONENTS*, the value of VOLSEQNO is set to the value of the DS1VOLSQ field in the Format-1 DSCB for each data set or component.

For example, `SELECT CRITERIA=(VOLSEQNO,GT,1)` selects all non-first portions of multivolume data sets.

For all other processing of VSAM clusters cataloged in ICF catalogs, the value of VOLSEQNO is set to the DS1VOLSQ field in the Format-1 DSCB for the first data component of each cluster. For VSAM clusters cataloged in non-ICF catalogs, this field is set to 1.

VSAMORG — VSAM organization type

The second operand in a VSAMORG comparison can be any of the following:

BROKEN

Broken VSAM data sets (component)

ESDS

Entry-sequenced data set (cluster)

KSDS

Key-sequenced data set (cluster)

LDS

Linear data set (cluster)

RRDS

Relative record data set (cluster)

VRRDS

Variable length relative record data set (cluster)

For example, to select only KSDS clusters; use:

```
SELECT CRITERIA=(DSORG,EQ,VSAM,AND,VSAMORG,EQ,KSDS)
```

After CA Disk determines that a VSAM data set is broken, the value of VSAMORG is set to BROKEN. Broken VSAM data sets will only be selected for the REPORT action command. Only the DSINDEX, DSUTIL, MAPPER, MVD and SVD reports will report on broken VSAM data sets. The broken VSAM data sets are reported at the component level as opposed to the cluster level.

When using the SCAN REALVOLS command, uncataloged or incorrectly cataloged VSAM components will be detected as broken if the COMPONENTS parameter is not specified. When using the SCAN CAT or FIND commands, cataloged VSAM components that do not physically reside on their cataloged volumes are detected as broken by CA Disk.

Specifying the CA Disk sysparm DSCLMSGS to Y causes CA Disk to issue messages concerning the broken VSAM data sets.

For more information, see DSORG—Data set Organization, PDSORG—Partitioned Data Set Organization, and PSORG—Physical Sequential Extended.

ZFS - zSeries File System (yes/no)

Set the value of this field to YES/TRUE/ON or NO/FALSE/OFF. For example, SELECT CRITERIA=(ZFS,EQ,YES) selects all zFS data sets.

Special Considerations While Processing PSEs

DSCL Selection Criteria

CRITERIA=(DSORG,EQ,PS)

or

CRITERIA=(DSORG,EQ,SEQ)

Either of these two examples selects both PS and PSE data sets.

DSCL Selection Criteria

CRITERIA=(PSORG,EQ,PSE)

or

CRITERIA=(PSORG,EQ,SEQE)

Either of these two examples selects PSE data sets only.

DSCL Selection Criteria

CRITERIA=(PSORG,EQ,PS)

or

CRITERIA=(PSORG,EQ,SEQ)

Either of these two examples selects PS data sets only

SYSPARMS

To restore PSE data sets, sysparm SMSALLOC should be set to a value of Y, or the DSNNAME must be eligible for PSE allocation.

PS/PSE Conversions

Data set conversions can be accomplished by preallocating the target data set prior to the CA Disk action being performed.

To restore an Archived PSE to a PS, the target PS must be either pre-allocated or its DFSMS DC must be changed. Specifically, the *data set NAME TYPE* field must be changed to NON-EXTENDED.

Note: Later phase will allow conversion of an Archived PS to PSE.

Year 2000

Due to the Year 2000 predicament, some dates are considered *special*. These dates include 0, 99000, 99365, 99366, and 99999, specified in any accepted date format. The concern is treating these dates as *special* dates to avoid inadvertent selection. Statements using relative operands such as TODAY+7 can select these special dates. Since DSCL normally just does an arithmetic compare, if the current date of TODAY were 2000.001 or higher, never expire dates such as 99365 would be selected.

To avoid this problem in the past, you would have to explicitly code exclusions in most cases. Now, DSCL avoids these special dates automatically. If an operand in the selection criteria is EXPDT, the other operand is checked. If this check resolves into a special date, the operator can only be EQ or NE. Using any other operands is not allowed. The operands themselves must also be single term operands, that is, no multiple term combinations with special dates such as EXPDT, EQ, and 99.365+10 are allowed.

Examples. Remember that any valid forms can be used with Select or Exclude to achieve desired results

EXPDT,GT,0	command reject
EXPDT,NE,0	true for all non zero expiration dates including <i>special dates</i>
EXPDT,LT,99365	command reject
EXPDT,EQ,99365,AND,EXPDT,LT,2100.001	valid but false for 99365 or any special date
EXPDT,EQ,99365,OR,EXPDT,LT,2100.001	true for 99365 and any non-special date below 2100
EXPDT,LT,2000.001,AND,EXPDT,EQ,0	True for all non-special expdts below Y2K and all zero expdts
EXPDT,GT,TODAY	True for all non-special expdts from current date to 2155.366
EXPDT+10,EQ,TODAY+20	Valid command but false for any special dates
EXPDT+10,NE,99365	Command reject

For improved performance, see F1EXPDT-Format-1 DSCB expiration date.

DSCL Select Criteria Boolean (AND/OR) Logic

The Boolean logic support of the CRITERIA parameter is invoked by specifying either an AND or an OR between conditional tests. Remember that a conditional test consists of an operand, an operator, and a second operand. You can specify a maximum of 64 conditional tests on a single CRITERIA parameter when each is separated by a compound condition (note that the maximum allowed can be affected by sysparm VSCRIMAX). Therefore, a valid Boolean test would be:

```
SELECT CRITERIA=(TRKS,GT,300,AND,PCTUSED,LT,80,AND,CREDT,GT,TODAY-60)
```

As in the example previous, if all of the compound conditions are AND, all of the conditional tests specified must be true for the statement to be true. Therefore, the data set would have to be larger than 300 tracks AND be less than 80 percent used AND have a creation date within the last 60 days to meet the criteria. Conversely, if all of the compound conditions are OR, if any of the conditional tests are true, the statement is true.

If both AND OR compound conditions are specified on a single command, all AND conditions are logically separated by each OR. In other words, if all AND conditions are true and an OR condition is encountered, the statement is considered true. To better describe the tests involved, let's look at a few examples.

- Suppose you would like to select all data sets that are either larger than 300 tracks and have a percent used of less than 65, or that are less than or equal to 300 tracks but have a percent used less than 50. The required command would be:

```
SELECT CRITERIA=(TRKS,GT,300,AND,PCTUSED,LT,65,  
OR,TRKS,LE,300,AND,PCTUSED,LT,50)
```

- Suppose you want to select all data sets that are larger than 50 tracks and meet one of the following criteria: either the data set is expired or the last modification date was not within the last 90 days. The command that would accomplish this is:

```
SELECT CRITERIA=(TRKS,GT,50,AND,EXPDT,LT,TODAY,AND,  
EXPDT,NE,0,  
OR,  
TRKS,GT,50,AND,MODDT,LT,TODAY-90)
```

Remember that in DSCL, neither indentation nor line breaks are important to the command parser, but they can be used to make the meaning clearer to users.

Note: In this example, the *TRKS,GT,50* test was specified on both sides of the OR condition. This is necessary because the OR condition logically separates the AND conditions. See what happens when the second *TRKS,GT,50* test is removed:

```
SELECT CRITERIA=(TRKS,GT,50,AND,EXPDT,LT,TODAY,AND,  
                EXPDT,NE,0,  
                OR,  
                MODDT,LT,TODAY-90)
```

With this test, data sets will be selected if:

- Tracks are greater than 50 AND the expiration date is less than today's date AND the expiration date is not zero, OR if
- The data set has not been modified in the last 90 days.

Note: The second test now makes no comparisons for track size and will be selected regardless of its size if the modification date test is true.

- To select all data sets spanning more than 400 tracks:

```
SELECT CRITERIA=(TRKS,GT,400)
```

- Select all data sets created since January 9, 1996:

```
SELECT CRITERIA=(CREDT,GT,1/9/1996)
```

- Select all data sets that have not been modified within the last year:

```
SELECT CRITERIA=(MODDT,LT,TODAY-365)
```

- Select all data sets that are greater than 300 tracks and that have not been modified in the last 60 days:

```
SELECT CRITERIA=(TRKS,GT,300,AND,MODDT,LT,TODAY-60)
```

- Select all data sets that either do not have a valid last use date, or that have not been used in the last year:

```
SELECT CRITERIA=(USEDT,EQ,0,OR,  
                USED,LT,TODAY-365)
```

EXCLUDE Command

The EXCLUDE command allows you to exclude data sets from processing. In order for a data set to be excluded, all tests specified on a single EXCLUDE command must be met. If only one parameter is specified, only that single test must be met for the exclusion to be honored. If the EXCLUDE command is specified more than once for any given ACTION command, it has the effect of a logical *OR* between EXCLUDE commands. That is, if any EXCLUDE command is true, then the data set is bypassed, regardless if other EXCLUDE commands test true or not.

For example, if the DSNAMES= parameter or the VOLUMES= parameter, but not both, are specified, conceptually think *don't process these data sets* or *don't process these volumes*. If both the DSNAMES= and VOLUMES= parameters are specified, conceptually think *don't process these data sets if they reside on these volumes*.

Note: If a SELECT command immediately precedes or follows any EXCLUDE command, special rules apply. See SELECT and EXCLUDE Rules for a further explanation.

The syntax of the EXCLUDE command, is as follows:

```
EXCLUDE DSNAMES=,COMPONENTS, TABLE=, VOLUMES=,UNIT=,VAMPOOL=,SGNAME=,VTOCS,  
CATALOGS=,CRITERIA=
```

DSNAMES=

You can specify the data sets to be excluded by NAME or PATTERN. Up to ten names or patterns are allowed.

Because an EXCLUDE DSNAMES= statement is already very restrictive, you cannot specify the VTOCS or CATALOGS= parameters on the same EXCLUDE statement. You can, however, add the COMPONENTS, TABLE=, VOLUMES=, VAMPOOL=, SGNAME= or CRITERIA= parameters.

For improved performance for VSAM, see the section COMPONENTS in the chapter "Reports."

COMPONENTS

Under *SCAN REALVOLS*, the COMPONENTS parameter will modify the action of EXCLUDE DSNAMES= or EXCLUDE TABLE= parameters to cause those parameters to act on VSAM component names (the names of their format-1 DSCBs) instead of their cluster names. Specifying this parameter can eliminate the need for CA Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet the EXCLUDE DSNAMES= or TABLE= list. This saves significant processing time and also prevents many *broken* VSAM cluster messages (those that see components outside of the range of the DSNAMES= or TABLE= parameters).

This COMPONENTS parameter is only allowed on EXCLUDE DSNAMES= or EXCLUDE TABLE= commands.

This COMPONENTS parameter is ineffective under SCAN CATALOGS= or FIND commands.

Use this parameter if you have installed DFP release 3.1 or above, have installed a VSAM allocation manager product or have taken administrative action to ensure that VSAM components have names that relate to their associated cluster names, or are not processing VSAM data sets.

TABLE=

Specify the name of a member in the parmlib data set that contains a list of data set names and patterns that are to be excluded from CA Disk processing.

Note: Data sets are not automatically excluded because of their appearance in this list. If any other parameters are specified on the same EXCLUDE command, all tests must be true for the data set to be excluded. For example, in the following command, the data set must both be listed in the member named *IMSDSNS* and be more than 300 tracks in size to be excluded:

```
EXCLUDE TABLE=IMSDSNS,CRITERIA=(TRKS,GT,300)
```

Because an EXCLUDE TABLE= statement is already very restrictive, you cannot specify the VTOCS, nor CATALOGS= parameters on the same EXCLUDE statement. You can, however, add the DSNAMES=, COMPONENTS, VOLUMES=, VAMPOOL=, SGNAME= or CRITERIA= parameters.

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* in the *Systems Guide*.

VOLUMES=

You can exclude all data sets on a volume or volume group by use of this parameter. Up to 30 volumes or volume patterns are allowed.

When an EXCLUDE command with the VOLUMES= parameter is specified under a SCAN REALVOLS command, special processing rules apply. If the VOLUMES= parameter is specified by itself with no other parameters (such as CRITERIA= or DSNAMES=), the volume will not have its VTOC read, even if other SELECT commands are present that name the same volume or pattern name. Conversely, if any other parameter is specified on the EXCLUDE command, this will have no effect on which volumes get scanned to satisfy the request.

You can add the DSNAMES=, COMPONENTS, TABLE=, VTOCS, CATALOGS= or CRITERIA= parameters on the same EXCLUDE statement. VOLUMES= is mutually exclusive with the other volume-oriented EXCLUDE command parameters of UNIT=, VAMPOOL= and SGNAME=. This means that VOLUMES= cannot be specified with UNIT=, VAMPOOL= or SGNAME= on the same EXCLUDE statement.

UNIT=

You can exclude all data sets on a unit or unit group by use of this parameter. Up to 16 units or unit patterns are allowed. This will also exclude a unit from being processed by VBACKUP.

When an EXCLUDE command with the UNIT= parameter is specified under a SCAN REALVOLS command, special processing rules apply. If the UNIT= parameter is specified by itself with no other parameters (such as CRITERIA= or DSNAMES=), the volume will not have its VTOC read, even if other SELECT commands are present that name the same unit or pattern name. Conversely, if any other parameter is specified on the EXCLUDE command, this will have no effect on which units get scanned to satisfy the request.

You can add the DSNAMES=, COMPONENTS, TABLE=, VTOCS, CATALOGS= or CRITERIA= parameters on the same EXCLUDE statement. UNIT= is mutually exclusive with the other volume-oriented EXCLUDE command parameters of VOLUMES=, VAMPOOL= and SGNAME=. This means that UNIT= cannot be specified with VOLUMES=, VAMPOOL= or SGNAME= on the same EXCLUDE statement.

VAMPOOL=

This parameter is only valid when CA Allocate (Volume Allocation Manager) is active on the system. It allows exclusion at the CA Allocate pool level. CA Disk. For example, EXCLUDE

VAMPOOL=(SYSTEM,DMS*) would cause CA Disk to check if the volume being processed or the cataloged volume of a data set being processed for selection is in the CA Allocate pool named SYSTEM or in pools whose names start with DMS.

Note: If you choose to use pattern matching for this parameter, you must use the CA Allocate pattern conventions.

To accomplish this selection CA Disk searches the in-memory copy of the CA Allocate pools using the volume currently being processed by a SCAN REALVOLS or the volume to which a data set being processed by a SCAN CATALOGS command is cataloged. A table of pool names containing this volume is generated. The generated table of pool names is then compared to the pool names or patterns coded in the VAMPOOL parameter to determine if a match exists.

There is a limit of 30 on the number of pool names that can be placed in the generated table. If the volume used for the search is contained in more than 30 CA Allocate pools, the 31st and higher pool names will not be entered in the table and, therefore, will not be used in the compare for a match with the VAMPOOL parameter.

On each SELECT command statement, you can specify up to 22 CA Allocate pool names or CA Allocate pool name patterns.

There are four DSCL EXCLUDE command volume-oriented parameters. They are VOLUMES=, UNIT=, VAMPOOL=, and SGNAME=, which are each mutually exclusive. This means that VAMPOOL= cannot be specified with UNIT=, VOLUMES=, or SGNAME= on the same EXCLUDE command statement.

SGNAME=

This parameter is only valid when IBM's Storage Management Subsystem (SMS) is active on the system. It allows exclusion at the DFSMS storage group level. Volumes and/or data sets will be excluded for processing based on whether or not their volser belongs to a particular storage group name specified in the SGNAME= parameter.

For example, EXCLUDE SGNAME=(SGLARGE,SG1/) would exclude all volumes and/or data sets that reside on the volumes, in the storage group of SGLARGE and in storage group names matching the pattern name of SG1/.

On each EXCLUDE statement, you can specify up to 22 DFSMS storage group names or DFSMS storage group name patterns.

There are four DSCL EXCLUDE command volume-oriented parameters. They are VOLUMES=, UNIT=, VAMPOOL=, and SGNAME=, which are each mutually exclusive. This means that SGNAME= cannot be specified with VOLUMES=, UNIT= or VAMPOOL= on the same EXCLUDE statement.

You can add the DSNAME=, COMPONENTS, TABLE=, VTOCS, CATALOGS= or CRITERIA= parameters on the same EXCLUDE statement.

VTOCS

Normally, DSCL processes user data sets only, and bypasses VTOCs, VTOC indexes, catalogs and VVDSs. This parameter is optional, you can specify it to have CA Disk bypass VTOCs and VTOC indexes.

Because an EXCLUDE VTOCS statement is already very restrictive, you cannot specify the DSNAME=, COMPONENTS, TABLE=, CATALOGS= nor CRITERIA= parameters on the same EXCLUDE statement. You can, however, add a VOLUMES=, VAMPOOL= or SGNAME= parameter.

CATALOGS=

Normally, DSCL processes user data sets only, and bypasses VTOCs, VTOC indexes, catalogs and VVDSs. This parameter is optional, you can specify it to have CA Disk bypass catalogs and VVDSs as specified by a required fully qualified data set name or CA Disk pattern.

Because an EXCLUDE CATALOGS= statement is already very restrictive, you cannot specify the DSNAME=, COMPONENTS, TABLE=, VTOCS nor CRITERIA= parameters on the same EXCLUDE statement. You can, however, add a VOLUMES=, VAMPOOL= or SGNAME= parameter.

CRITERIA=

Specify the exclusion criteria that, if met, will cause processing by CA Disk for that data set to be bypassed. See CRITERIA= for additional details.

Because an EXCLUDE CRITERIA= statement is already very restrictive, you cannot specify the VTOCS nor CATALOGS= parameters on the same EXCLUDE statement. You can, however, add the DSNAME=, COMPONENTS, TABLE=, VOLUMES=, VAMPOOL= or SGNAME= parameters.

ACTION Commands

The commands REPORT, VREPORT, BACKUP, BACKUPCC, ARCHIVE, DELETE, VBACKUP, RELEASE, MOVE, COPY, EXPIRE, and BILLING are known as action commands because each causes CA Disk to take a specific action against a selected data set or volume. You must provide at least one of these commands for each SCAN/FIND group.

- The REPORT and VREPORT commands and their parameters are described in the chapter "Reports."
- The BACKUP, BACKUPCC, ARCHIVE, DELETE, and EXPIRE commands and their parameters are described in the chapter "Backup/Archive."
- The VBACKUP command and its parameters are described in the chapter "VBACKUP/VRECOVER."
- The RELEASE command and its parameters are described in the chapter "Idle Space Release."
- The MOVE and COPY commands and their parameters are described in "MOVE/COPY."
- The BILLING command is described in "DASD Billing."

Complex DSCL Command Sequences

Even complicated structures are based on the simple rules discussed previously. In time, you will be able to build very sophisticated processing, in one or two scans if you like. An example:

```
//jobname JOB (acct,info),etc.
//*****
//* GET A BEFORE IMAGE REPORT, DELETE TRASH, ARCHIVE UNUSED  *
//* data sets, BACK UP EVERYTHING USEFUL, RELEASE SPACE ON  *
//* OVERALLOCATED data sets. THEN GET AN AFTER IMAGE REPORT *
//*****
//STEP EXEC DMS
SET MODE=SIMULATE
PASSWORD MASTER=the-DMS-master-password
SCAN REALVOLS
*****
* GET A BEFORE IMAGE REPORT:                                *
  VREPORT ALLOCS
*****
* DELETE TRASH:                                              *
*****
  SELECT CRITERIA=(TEMP,EQ,YES,AND,CREDT,LE,TODAY-2)
  SELECT CRITERIA=(CATSTAT,NE,YES,AND,CREDT,LE,TODAY-2)
  DELETE
*****
* ARCHIVE MOST GDGS, LEAVING 2 GENERATIONS ON DISK:        *
*****
  EXCLUDE DSNAMES=(data.set.name.patterns.that.should,
    have.more.than.two.generations,left.on.disk),
  SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-2)
  ARCHIVE DISP=RECAT,EXPDT=99000 catalog control
*****
* ARCHIVE OTHER GDGS, LEAVING 5 GENERATIONS ON DISK:        *
*****
  SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-5)
  ARCHIVE DISP=RECAT,EXPDT=99000 catalog control
*****
* ARCHIVE UNUSED STUFF FOR 7 YEARS, ALLOW AUTO-RESTORE:     *
*****
  SELECT CRI=(USEDT,LE,TODAY-28)          not used in 28 days
  EXCLUDE DSNAMES=SYS/                    leave my SYS/ on disk
  EXCLUDE DSNAMES=*.DSNDBC./             DB2 doesn't maintain,USEDT
  ARCHIVE DISP=RECAT,RETPD=2557
*****
* data set-ORIENTED BACKUP EVERYTHING THAT NEEDS IT:        *
*****
  SELECT CRITERIA=(MODIFIED,EQ,YES)        newly modified
  SELECT CRITERIA=(ARCDT,LE,TODAY-27)     about to roll off
```

```

SELECT DSNAME=*.DSNDBC./      DB2 doesn't maintain MODDT
SELECT CATALOGS=/              my ICF catalogs
SELECT VT0CS                   for RECOVER VOL=,VT0CS
    BACKUP RETPD=28
*****
* RELEASE OVER-ALLOCATED SPACE:                               *
*****
    SELECT CRITERIA=(DSORG,EQ,VSAM,AND,PCTUSED,LT,70)
        RELEASE PCTUSED=75
    SELECT CRITERIA=(DSORG,EQ,P0,AND,PCTUSED,LT,75)
        RELEASE PCTUSED=80
    SELECT CRITERIA=(DSORG,EQ,PS,AND,PCTUSED,LT,100)
        RELEASE PCTUSED=100
//*****
//* GET AN AFTER IMAGE REPORT:                                *
//*****
// EXEC DMS
SCAN REALVOLS
VREPORT ALLOCS

```

DFSMS and DSCL

The following pages contain additional information on the SELECT/EXCLUDE parameters that pertain exclusively to the processing of SMS data.

Syntactical use of these fields is described in DSCL SELECT CRITERIA Field Name Descriptions.

Note: There are several criteria selection fields that are used in the form of a character literal. Character literals are those fields that are of a free form nature and thus must be enclosed between the < and > symbols. CA Disk pattern matching is supported for character literals.

Additional Select Parameter

SGNAME=

This parameter is only valid when IBM's Storage Management Subsystem (SMS) is active on the system. It allows selection at the DFSMS storage group level. Volumes and/or data sets will be selected for processing based on whether or not their volser belongs to a particular storage group name specified in the SGNAME= parameter.

For example, SELECT SGNAME=(SGLARGE,SG1/) would select all volumes and/or data sets that reside on the volumes, in the storage group of SGLARGE and in storage group names matching the pattern name of SG1/.

There are four DSCL SELECT command volume-oriented parameters. They are VOLUMES=, UNIT=, VAMPOOL=, and SGNAME=, which are each mutually exclusive. This means that SGNAME= cannot be specified with VOLUMES=, UNIT=, or VAMPOOL= on the same SELECT statement. On each SELECT statement, you can specify up to 22 DFSMS storage group names or DFSMS storage group name patterns.

You can add the DSNAME=, COMPONENTS, TABLE=, VTOCS, CATALOGS= or CRITERIA= parameters on the same SELECT statement.

An example of how to use this parameter is as follows:

```
SCAN REALVOLS
  SELECT SGNAME=(LARGE,SMALL),CRI=(DSORG,EQ,AM)
  BACKUP
```

The previous example accomplishes an incremental and guaranteed backup of VSAM clusters in an SMS environment. CA Disk will scan the VTOC's of the SMS storage groups called *LARGE* and *SMALL* and will back up the clusters which have the change bit turned on, or which meet the criteria for guaranteed backup. The SMS information for each cluster will be used as criteria when determining whether the cluster will be backed up or not.

SMS Data Class Fields

Following is a list of possible Data Class fields. The table provides the following requirements for each field: Maximum length, Type of parameters, and an example.

Field Name	Definition	Max Length	Type	Example
DCNAME	Data Class Name	30	Char Literal	CRI=(DCNAME,EQ,AME/)
DCDESC	Data Class Description	120	Char Literal	CRI=(DCDESC,EQ,data class descrip/)
DCUSERID	Last USERID to update the Data Class	8	Char Literal	CRI=(DCUSERID,EQ,AME/)
DCDATE	Date of update	N/A	CA Disk formats	CRI=(DCDATE,EQ,08/13/94)
DCTIME	Time of update	4	Must be 4 numeric digits in HHMM format	CRI=(DCTIME,EQ,0830)
DCFRCORG	VSAM REORG specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFRCORG,EQ,YES)

Field Name	Definition	Max Length	Type	Example
DCFLRECL	LRECL specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFLRECL,EQ,YES)
DCFRECFM	RECFM specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFRECFM,EQ,YES)
DCFKEYLN	Key length specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFKEYLN,EQ,YES)
DCFKEYOF	Key offset specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFKEYOF,EQ,YES)
DCFEXPDT	EXPDT specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFEXPDT,EQ,YES)
DCFRETPD	RETPD specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFRETPD,EQ,YES)
DCFPRMSP	Primary space specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFPRMSP,EQ,YES)
DCFSECSP	Secondary space specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFSECSP,EQ,YES)
DCFDIRBL	Directory blocks specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFDIRBL,EQ,YES)
DCFAUNIT	Allocation unit specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFAUNIT,EQ,YES)
DCFAVREC	Average record specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFAVERC,EQ,YES)
DCFVOLCT	Volume count specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFVOLCT,EQ,YES)
DCFCSIZE	CI size specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFCSIZE,EQ,YES)
DCFCIFSP	CI free space specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFCIFSP,EQ,YES)
DCFCASFSP	CA free space specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFCASFSP,EQ,YES)
DCFSOREG	XREGION share option specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFSOREG,EQ,YES)
DCFSOSYS	XSYSTEM share option specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFSOSYS,EQ,YES)
DCFIMBED	VSAM IMBED specified	N/A	Fixed Params - (YES or NO)	CRI=(DCFIMBED,EQ,YES)

Field Name	Definition	Max Length	Type	Example
DCFREPL	VSAM Replicate specified	N/A	Fixed Parm - (YES or NO)	CRI=(DCFREPL,EQ,YES)
DCRECORG	VSAM ORG	4	Fixed Parm - (KSDS,ESDS,RRDS, or LDS)	CRI=(DCRECORG,EQ,LSD)
DCLRECL	Logical record length	5	Numeric	CRI=(DCLRECL,EQ,4060)
DCRECFM	RECFM flags	4	Fixed Parm - (F,FB,FBA,FBM,V,VB,VBA,VB M or U)	CRI=(DCRECFM,EQ,FB)
DCKEYLEN	Key length	3	Numeric	CRI=(DCKEYLEN,EQ,44)
DCKEYOFF	Key offset	5	Numeric	CRI=(DCKEYOFF,EQ,100)
DCEXPDT	Expiration date	N/A	CA Disk date format	CRI=(DCKEYLEN,EQ,12/31/99)
DCRETPD	Retention period	4	Numeric	CRI=(DCRETPD,EQ,30)
DCSPPRIM	Primary space allocation	6	Numeric	CRI=(DCSPPRIM,EQ,100)
DCSPSEC	Secondary space allocation	6	Numeric	CRI=(DCSPSEC,EQ,50)
DCSPDIR	Directory block allocation	6	Numeric	CRI=(DCSPDIR,EQ,35)
DCAUNIT	Average value	5	Numeric	CRI=(DCAUNIT,EQ,133)
DCAVGREC	Type of allocation if allocation unit is specified	1		CRI=(DCAVGREC,EQ,)
DCVOLCNT	Volume count	2	Numeric	CRI=(DCVOLCNT,EQ,15)
DCCISZ	CI size	5	Numeric	CRI=(DCCISZ,EQ,1026)
DCCIFRSP	CI freespace percent	3	Numeric	CRI=(DCCIFRSP,EQ,25)
DCCAFRSP	CA freespace percent	3	Numeric	CRI=(DCCAFRSP,EQ,25)
DCXREGSO	Share option - region	1	Fixed Parm - (1,2,3 or 4)	CRI=(DCXREGSO,EQ,3)
DCXSYSSO	Share option - system	1	Fixed Parm - (1,2,3 or 4)	CRI=(DCXSYSSO,EQ,4)

Field Name	Definition	Max Length	Type	Example
DCINIMBD	VSAM IMBED option	N/A	Fixed Parm - (YES or NO)	CRI=(DCINIMBD,EQ,NO)
DCINREPL	VSAM replicate option	N/A	Fixed Parm - (YES or NO)	CRI=(DCINREPL,EQ,NO)

SMS Storage Class Fields

Following is a list of possible Storage Class fields. The table provides the following requirements for each field: Maximum length, Type of parameters, and an example.

Field Name	Definition	Max Length	Type	Example
SCNAME	Storage Class name	30	Char Literal	CRI=(SCNAME,EQ,<LARGE/>)
SCDESC	Storage Class description	120	Char Literal	CRI=(SCDESC,EQ,<storage class descrip/>)
SCUSERID	Last USERID to update Storage Class	8	Char Literal	CRI=(SCUSERID,EQ,<NAME/>)
SCDATE	Date of update	N/A	CA Disk data formats	CRI=(SCDATE,EQ,07/30/94)
SCTIME	Time of update	4	Must be 4 numeric digits in HHMM format	CRI=(SCTIME,EQ,0530)
SCFGUSPC	Guaranteed space	N/A	Fixed Parm - (YES or NO)	CRI=(SCFGUSPC,EQ,YES)
SCFAVAIL	Availability specified	N/A	Fixed Parm - (YES or NO)	CRI=(SCFAVAIL,EQ,YES)
SCFDIRET	Direct response time	N/A	Fixed Parm - (YES or NO)	CRI=(SCFDIRET,EQ,YES)
SCFDIBIA	Direct bias	N/A	Fixed Parm - (YES or NO)	CRI=(SCFDIBIA,EQ,YES)
SCFSERET	Sequential response time	N/A	Fixed Parm - (YES or NO)	CRI=(SCFSERET,EQ,YES)
SCFSEBIA	Sequential bias	N/A	Fixed Parm - (YES or NO)	CRI=(SCFSEBIA,EQ,YES)

Field Name	Definition	Max Length	Type	Example
SCAVAIL	Availability	N/A	Fixed Parm - (STANDARD or CONTINUS)	CRI=(SCAVAIL,EQ,STANDARD OR CONTINUS)
SCDRESP	Direct Millsec response	3	Numeric	CRI=(SCDRESP,EQ,10)
SCDBIAS	Direct bias	N/A	Fixed Parm - (READ or WRITE)	CRI=(SCDBIAS,EQ,READ or WRITE)
SCSRESP	Sequential millsec response	3	Numeric	CRI=(SCSRESP,EQ,5)
SCSBias	Sequential bias	N/A	Fixed Parm - (READ or WRITE)	CRI=(SCSBias,EQ,READ or WRITE)

SMS Management Class Fields

Following is a list of possible Management Class fields. The table provides the following requirements for each field: Maximum length, Type of parameters, and an example.

Field Name	Definition	Max Length	Type	Example
MCNAME	Management class name	30	Char Literal	CRI=(MCNAME,EQ,<NAME/>)
MCDESC	Management class description	120	Char Literal	CRI=(MCDESC,EQ,<DisasterRecovery/>)
MCUSERID	Last USERID to update management class	8	Char Literal	CRI=(MCUSERID,EQ,userid/)
MCDATE	Date of last update	N/A	CA Disk date format	CRI=(MCDATE,EQ,01/13/94)
MCTIME	Time of last update	4	Must be 4 numeric digits in HHMM format	CRI=(MCTIME,EQ,0710)
MCIDSPRL	Idle Space Release	N/A	Fixed Parm - (YES, NO or COND)	CRI=(MCIDSPRL,EQ,NO)

Field Name	Definition	Max Length	Type	Example
MCFPRDYS	Primary days specified	N/A	Fixed Parm - (YES or NO)	CRI=(MCFPRDYS,EQ,NO)
MCFDADYS	Disk archive days specified	N/A	Fixed Parm - (YES or NO)	CRI=(MCFDADYS,EQ,NO)
MCFGELP	GDG on primary specified	N/A	Fixed Parm - (YES or NO)	CRI=(MCFGELP,EQ,NO)
MCFBKFQ	Backup frequency specified	N/A	Fixed Parm - (YES or NO)	CRI=(MCFBKFQ,EQ,NO)
MCFBVRS	Backup version specified	N/A	YES/NO	CRI=(MCFBVRS,EQ,NO)
MCFBVRD	Backup version data set delete specified	N/A	Fixed Parm - (YES or NO)	CRI=(MCFBVRD,EQ,NO)
MCFBLDY	Days to keep last backup version specified	N/A	Fixed Parm - (YES or NO)	CRI=(MCFBLDY,EQ,NO)
MCFBDAYS	Days to keep backup version specified	N/A	Fixed Parm - (YES or NO)	CRI=(MCFBDAYS,EQ,NO)
MCPRDYNU	Primary days non-usage	4	Numeric	CRI=(MCPRDYNU,EQ,30)
MCADDYNU	Disk archive days non-usage	4	Numeric	CRI=(MCADDYNU,EQ,30)
MCARCEA	Archive	N/A	NONE, COMMAND, or BOTH	CRI=(MCARCEA,EQ,BOTH)
MCGDGELP	GDG data sets left on primary	3	Numeric	CRI=(MCGDGELP,EQ,3)
MCROFGDS	Rolled off GDS action	N/A	ARCHIVE or EXPIRE	CRI=(MCROFGDS,EQ,EXPIRE)
MCEXDNU	Expire days non-usage	4 Numeric digits or fixed parameter	0-9999 or NOLIMIT	CRI=(MCEXDNU,EQ,NOLIMIT) or CRI=(MCEXDNU,EQ,25)
MCEXPDT	Expiration date	N/A	CA Disk data formats	CRI=(MCEXPDT,EQ,01/30/98)

Field Name	Definition	Max Length	Type	Example
MCEXDAYS	Expire after days	4 Numeric digits or fixed parameter	0-9999 or NOLIMIT	CRI=(MCEXDAYS,EQ,NOLIMIT) or CRI=(MCEXDAYS,EQ,60)
MCRETLT	Retention limit	4 Numeric digits or fixed parameter	0-9999 or NOLIMIT	CRI=(MCRETLT,EQ,NOLIMIT) or CRI=(MCRETLT,EQ,60)
MCBKFEQ	Backup frequency	4 Numeric digits	0-9999	CRI=(MCBKFEQ,EQ,60)
MCBKVER	Number of backup versions	2 Numeric digits	1-13	CRI=(MCBKVER,EQ,13)
MCBKVERD	Number of backup versions - data set deleted	2 Numeric digits	1-13	CRI=(MCBKVERD,EQ,2)
MCLSTBK	Days to keep last backup version	4 Numeric digits or fixed parameter	0-9999 or NOLIMIT	CRI=(MCLSTBK,EQ,NOLIMIT) or CRI=(MCLSTBK,EQ,60)
MCBKDAYS	Days to keep backup versions	4 Numeric digits or fixed parameter	0-9999 or NOLIMIT	CRI=(MCBKDAYS,EQ,NOLIMIT) or CRI=(MCBKDAYS,EQ,60)
MCBKCMD	Backup	N/A	Fixed Parm - (NONE, ADMIN or BOTH)	CRI=(MCBKCMD,EQ,BOTH)
MCAUTOBK	Auto BACKUP	N/A	Fixed Parm - (YES, or NO)	CRI=(MCAUTOBK,EQ,YES)

SMS Storage Group Fields

Following is a list of possible Storage Group fields. The table provides the following requirements for each field: Maximum Length, Type of parameters, and an Example.

Field Name	Definition	Max Length	Type	Example
SGNAME	Storage Group name	30	Char Literal	CRI=(SGNAME,EQ,<NAME/>)
SGDESC	Storage Group description	120	Char Literal	CRI=(SGDESC,EQ,<storage group descrip/>)

Field Name	Definition	Max Length	Type	Example
SGUSERID	Last USERID to update Storage Group	8	Char Literal	CRI=(SGUSERID,EQ, <NAME/>)
SGDATE	Date of update SGTIME Time of update	N/A	CA Disk date format	CRI=(SGDATE,EQ, 01/12/04)
SGTIME	Time of Update	4	Must be 4 numeric digits in HHMM format	CRI=(SGTIME,EQ,0800)
SGFAUBUP	Auto BACKUP	N/A	Fixed Parms - (YES, or NO)	CRI=(SGFAUBUP,EQ,YES)
SGFAUARC	Auto ARCHIVE	N/A	Fixed Parms - (YES, or NO)	CRI=(SGFAUARC,EQ,YES)
SGFAVBUP	Auto VBACKUP	N/A	Fixed Parms - (YES, or NO)	CRI=(SGFAVBUP,EQ,YES)
SGFTHRES	Thresholds specified	N/A	Fixed Parms - (YES, or NO)	CRI=(SGFTHRES,EQ,YES)
SGTYPE	Type	N/A	Fixed Parms - (POOL, VIO or DUMMY)	CRI=(SGTYPE,EQ,VIO)
SGVMAX	Maximum size of VIO data sets	7	Numeric	CRI=(SGVMAX,EQ,4080)
SGVUNIT	Generic device type for VIO data sets	4	—	CRI=(SGVUNIT,EQ,3380)
SGHTHR	High threshold percent	2	Numeric	CRI=(SGHTHR,EQ,95)
SGLTHR	Low threshold percent	2	Numeric	CRI=(SGLTHR,EQ,75)
SGVBUPC1	VBACKUP class 1	8	Char Literal	CRI=(SGVBUPC1,EQ,<buclass1>)
SGVBUPC2	VBACKUP class 2	8	Char Literal	CRI=(SGVBUPC2,EQ,<buclass2>)
SGVBUPC3	VBACKUP class 3	8	Char Literal	CRI=(SGVBUPC3,EQ,<buclass3>)
SGVBUPC4	VBACKUP class 4	8	Char Literal	CRI=(SGVBUPC4,EQ,<buclass4>)
SGVBUPC5	VBACKUP class 5	8	Char Literal	CRI=(SGVBUPC5,EQ,<buclass5>)

Field Name	Definition	Max Length	Type	Example
SGSYSST1	Status on system 1	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGSYSST1,EQ,ENABLED)
SGSYSST2	Status on system 2	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGSYSST2,EQ,ENABLED)
SGSYSST3	Status on system 3	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGSYSST3,EQ,ENABLED)
SGSYSST4	Status on system 4	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGSYSST4,EQ,ENABLED)
SGSYSST5	Status on system 5	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGSYSST5,EQ,ENABLED)
SGSYSST6	Status on system 6	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGSYSST6,EQ,ENABLED)

Field Name	Definition	Max Length	Type	Example
SGSYSST7	Status on system 7	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGSYSST7,EQ,ENABLED)
SGSYSST8	Status on system 8	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGSYSST8,EQ,ENABLED)
SGCSMSS1	Status on system 1	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGCSMSS1,EQ,NOSTATUS)
SGCSMSS2	Status on system 2	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGCSMSS2,EQ,NOSTATUS)
SGCSMSS3	Status on system 3	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGCSMSS3,EQ,NOSTATUS)
SGCSMSS4	Status on system 4	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGCSMSS4,EQ,NOSTATUS)

Field Name	Definition	Max Length	Type	Example
SGCSMSS5	Status on system 5	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGCSMSS5,EQ,NOSTATUS)
SGCSMSS6	Status on system 6	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGCSMSS6,EQ,NOSTATUS)
SGCSMSS7	Status on system 7	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGCSMSS7,EQ,NOSTATUS)
SGCSMSS8	Status on system 8	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, DISABALL, DISABNEW OR NOSTATUS)	CRI=(SGCSMSS8,EQ,NOSTATUS)
SGABKSYS	Auto BACKUP system name	8	Char Literal	CRI=(SGABKSYS,EQ,<buname>)
SGAVBSYS	Auto VBACKUP system name	8	Char Literal	CRI=(SGAVBSYS,EQ,<buname>)
SGAARSYS	Auto ARCHIVE system name	8	Char Literal	CRI=(SGAARSYS,EQ,arcname)

Note: ENABLED relates to IBM's ENABLE, QUIESALL relates to IBM's QUIALL, QUIESNEW relates to IBM's QUINEW, DISABALL relates to IBM's DISALL, DISABNEW relates to IBM's DISNEW, and NOSTATUS relates to IBM's NOTCON.

SMS Volume Definition Fields

Following is a list of possible Volume definition fields. Although these fields are volume related, they also apply to individual data sets and as such, do not have as much application as the SMS class or group fields.

Field Name	Definition	Max Length	Type	Example
VLNAME	Volume Definition name	30	Char Literal	CRI=(VLNAME,EQ,<NAME>/)
VLUSERID	Last USERID to update volume definition	8	Char Literal	CRI=(VLUSERID,EQ,<NAME>/)
VLDATE	Date of update	N/A	CA Disk date formats	CRI=(VLDATE,EQ,09/13/94)
VLTIME	Time of update	4	Must be 4 numeric digits in HHMM format	CRI=(VLTIME,EQ,0800)
VLFCNVR	Volume is in conversion	N/A	Fixed Parms - (YES, or NO)	CRI=(VLFCNVR,EQ,NO)
VLSTGP	Storage group of volume	30	Char Literal	CRI=(VLSTGP,EQ,<LARGE>/)
VLSSMS1	SMS status - system 1	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABNEW)	CRI=(VLSSMS1,EQ,DISABALL)
VLMSVS1	MVS status - system 1	N/A	Fixed format - (ONLINE, OFFLINE, BOXED, NOTREADY, POFFLINE)	CRI=(VLMSVS1,EQ,BOXED)
VLSSMS2	SMS status - system 2	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLSSMS2,EQ,DISABALL)

Field Name	Definition	Max Length	Type	Example
VLSMVS2	MVS status - system 2	N/A	Fixed format - (ONLINE, OFFLINE, BOXED, NOTREADY, POFFLINE)	CRI=(VLSMVS2,EQ,BOXED)
VLSSMS3	SMS status – system 3	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLSSMS3,EQ,DISABALL)
VLSMVS3	MVS status – system 3	N/A	Fixed format - (ONLINE, OFFLINE, BOXED, NOTREADY, POFFLINE)	CRI=(VLSMVS3,EQ,BOXED)
VLSSMS4	SMS status – system 4	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLSSMS4,EQ,DISABALL)
VLSMVS4	MVS status – system 4	N/A	Fixed format - (ONLINE, OFFLINE, BOXED, NOTREADY, POFFLINE)	CRI=(VLSMVS4,EQ,BOXED)
VLSSMS5	SMS status – system 5	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS,DI SABALL, or DISABENE)	CRI=(VLSSMS5,EQ,DISABALL)

Field Name	Definition	Max Length	Type	Example
VLSMVS5	MVS status – system 5	N/A	Fixed format - (ONLINE, OFFLINE, BOXED, NOTREADY, POFFLINE)	CRI=(VLSMVS5,EQ,BOXED)
VLSSMS6	SMS status - system 6	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS,DI SABALL, or DISABENE)	CRI=(VLSSMS6,EQ,DISABALL)
VLSMVS6	MVS status - system 6	N/A	Fixed format - (ONLINE, OFFLINE, BOXED, NOTREADY, POFFLINE)	CRI=(VLSMVS6,EQ,BOXED)
VLSSMS7	SMS status - system 7	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLSSMS7,EQ,DISABALL)
VLSMVS7	MVS status - system 7	N/A	Fixed format - (ONLINE, OFFLINE, BOXED, NOTREADY, POFFLINE)	CRI=(VLSMVS7,EQ,BOXED)
VLSSMS8	SMS status - system 8	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLSSMS8,EQ,DISABALL)

Field Name	Definition	Max Length	Type	Example
VLBMVS8	MVS status - system 8	N/A	Fixed format - (ONLINE, OFFLINE, BOXED, NOTREADY, POFFLINE)	CRI=(VLBMVS8,EQ,BOXED)
VLTCPCY	Total capacity	9	Numeric	CRI=(VLTCPCY,EQ,1000)
VLFREES	Free space	7	Numeric	CRI=(VLFREES,EQ,4000)
VLLFEXT	Largest free extent	7	Numeric	CRI=(VLLFEXT,EQ,1)
VLSTSGS	Storage status - this system	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLSTSGS,EQ,NOSTATUS)
VLUPLEV	Volume update level	9	Numeric	CRI=(VLUPLEV,GT,5)
VLCSMS1	Confirmed SMS status - system 1	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLCSMS1,EQ,NOSTATUS)
VLCSMS2	Confirmed SMS status - system 2	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLCSMS2,EQ,NOSTATUS)
VLCSMS3	Confirmed SMS status - system 3	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLCSMS3,EQ,NOSTATUS)

Field Name	Definition	Max Length	Type	Example
VLCSMS4	Confirmed SMS status - system 4	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLCSMS4,EQ,NOSTATUS)
VLCSMS5	Confirmed SMS status - system 5	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLCSMS5,EQ,NOSTATUS)
VLCSMS6	Confirmed SMS status - system 6	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLCSMS6,EQ,NOSTATUS)
VLCSMS7	Confirmed SMS status - system 7	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLCSMS7,EQ,NOSTATUS)
VLCSMS8	Confirmed SMS status - system 8	N/A	Fixed parms- (ENABLED, QUIESALL, QUIESNEW, NOSTATUS, DISABALL, or DISABENE)	CRI=(VLCSMS8,EQ,NOSTATUS)

Criteria for CA—Select Interface

These criteria can be specified on the DSCL BACKUP command to provide additional selection criteria specific to CA-Select. Should any of these criteria be specified in an environment where CA-Select is not installed, a message will be issued and the program will terminate.

Field Name	Definition	Criteria
SELDBYT#	Number of bytes changed since last differential backup (number)	Decimal number between 1 and 2147483647
SELDBYT%	Percent of bytes changed since last differential backup (number)	Decimal number between 1 and 99
SELDMEM#	Number of members changed since last differential backup (number)	Decimal number between 1 and 65535
SELDMEM%	Percent of members changed since last differential backup (number)	Decimal number between 1 and 99
SELFBYT#	Number of bytes changed since last full backup (number)	Decimal number between 1 and 2147483647
SELFBYT%	Percent of bytes changed since last full backup (number)	Decimal number between 1 and 99
SELFMEM#	Number of members changed since last full backup (number)	Decimal number between 1 and 65535
SELFMEM%	Percent of members changed since last full backup (number)	Decimal number between 1 and 99
SELFPSDM	Has the data set changed since the last full backup (yes/no)	If specified as a value of YES, the data set is selected as being modified. If specified as a value of NO, modified data set will be bypassed.
SELFREC#	Number of records changed since last full backup (number)	Decimal number between 1 and 65535
SELFREC%	Percent of records changed since last full backup (number)	Decimal number between 1 and 99
SELIBYT#	Number of bytes changed since last incremental backup (number)	Decimal number between 1 and 2147483647
SELIBYT%	Percent of bytes changed since last incremental backup (number)	Decimal number between 1 and 99
SELIMEM#	Number of members changed since last incremental backup (number)	Decimal number between 1 and 65535

Field Name	Definition	Criteria
SELIMEM%	Percent of members changed since last incremental backup (number)	Decimal number between 1 and 99
SELMXDIF	Number of differential backups taken for the data set since the last full backup (number)	Decimal number between 1 and 32767
SELMXINC	Number of incremental backups taken for the data set since the last full backup	Decimal number between 1 and 32767

SELDBYT#—Since last differential backup (number)

This criterion indicates the number of bytes that have changed within a PDS since the last differential backup. The value specified must be a decimal number between 1 and 2147483647.

SELDBYT%—Since last differential backup (number)

This criterion represents the percentage of bytes that have changed within a PDS since the last differential backup. The value specified must be a decimal number between 1 and 99.

SELDMEM#—Since last differential backup (number)

This criterion indicates the number of members that have changed within a PDS since the last differential backup. The value specified must be a decimal number between 1 and 65535.

SELDMEM%—Since last differential backup (number)

This criterion represents the percentage of members that have changed within a PDS since the last differential backup. The value specified must be a decimal number between 1 and 99.

SELFBYT#—Since last full backup (number)

This criterion indicates the number of bytes that have changed within a PDS since the last full backup. The value specified must be a decimal number between 1 and 2147483647.

SELFBYT%—Since last full backup (number)

This criterion represents the percentage of bytes that have changed within a PDS since the last full backup. The value specified must be a decimal number between 1 and 99.

SELFMEM#—Since last full backup (number)

This criterion indicates the number of members that have changed within a PDS since the last full backup. The value specified must be a decimal number between 1 and 65535.

SELFMEM%—Since last full backup (number)

This criterion represents the percentage of members that have changed within a PDS since the last full backup. The value specified must be a decimal number between 1 and 99.

SELFPSDM—Modified since last full backup (yes/no)

This criterion indicates whether or not fingerprinting has determined if a data set has been modified since the last full backup. If specified as a value of YES, the data set is selected as being modified. If specified as a value of NO, modified data set will be bypassed.

SELFREC#—Changed since last full backup (number)

This criterion indicates the number of records that have changed within a VSAM data set since the last full backup. The value specified must be a decimal number between 1 and 65535.

SELFREC%—Changed since last full backup (number)

This criterion represents the percentage of records that have changed within a VSAM data set since the last full backup. The value specified must be a decimal number between 1 and 99.

SELIBYT#—Since last incremental backup (number)

This criterion indicates the number of bytes that have changed within a PDS since the last incremental backup. The value specified must be a decimal number between 1 and 2147483647.

SELIBYT%—Since last incremental backup (number)

This criterion represents the percentage of bytes that have changed within a PDS since the last incremental backup. The value specified must be a decimal number between 1 and 99.

SELIMEM#—Since last incremental backup (number)

This criterion indicates the number of members that have changed within a PDS since the last incremental backup. The value specified must be a decimal number between 1 and 65535.

SELIMEM%—Since last incremental backup (number)

This criterion represents the percentage of members that have changed within a PDS since the last incremental backup. The value specified must be a decimal number between 1 and 99.

SELMXDIF—Differential backups taken (number)

This criterion determines when a full backup should be taken of a PDS. The value specified must be a decimal number between 1 and 32767. This value is compared to the total number of differential backups taken for the PDS since the last full backup. When the total number of differential backups matches the value of this criterion, a full backup is issued.

SELMXINC—Incremental backups taken (number)

This criterion determines when a full backup should be taken of a PDS. The value specified must be a decimal number between 1 and 32767. This value is compared to the total number of incremental backups taken for the PDS since the last full or differential backup. When the total number of incremental backups matches the value of this criterion, a full backup is issued.

Chapter 3: Basic System FMS PROC

This section contains the following topics:

[About FMS PROC](#) (see page 147)

[Condition Codes](#) (see page 148)

[Commands](#) (see page 148)

[FMS Command Sequence](#) (see page 149)

[SET Command](#) (see page 150)

[PASSWORD Command \(For VSAM Only\)](#) (see page 152)

[SCAN Command](#) (see page 153)

[SELECT Command](#) (see page 153)

[FMS Select Criteria Boolean \(AND/OR\) Logic](#) (see page 165)

[EXCLUDE Command](#) (see page 166)

[ACTION Commands](#) (see page 166)

About FMS PROC

The commands that can be executed by the FMS PROC are: RECOVER

FMS processes both non-VSAM data sets and VSAM clusters. In its simplest form, the JCL required to use FMS would look like:

```
//jobname JOB (acct,info),...
//STEP EXEC FMS
        FMS command
```

The objective of adding the FMS PROC to CA Disk was to provide a DSCL-like command language for data set recovery. This PROC provides equivalent functionality available in the original RECOVER PROC, and is equipped with its own palette of commands and parameters.

You also have a rich set of criteria to select data sets for and exclude data sets from processing as desired. CA Disk preprocesses your command stream to find any overlapping requests, that is, those that request processing of the same volumes or catalogs.

To avoid unexpected execution of incomplete or partially bad command strings, the FMS PROC takes no action until all commands have been successfully understood.

To allow for sorted reports, the FMS PROC has separate process, sort and print steps. Sorted reports are also available in simulate mode.

Condition Codes

The FMS PROC returns the following JCL condition codes:

0

Successful completion.

4

No data sets selected or 1 or more data sets not restored.

8

Data set(s) resource cannot be acquired (for example, security or enqueues).

12

Unrecoverable error encountered in data set(s).

16

Command Error.

Commands

The FMS PROC is controlled by commands provided to the SYSIN DD statement. The commands are of four main types:

- Environment commands. Those commands that set up the environment in which CA Disk will run: SET and PASSWORD.
- Candidate commands. Those commands that find candidate data sets: SCAN.
- Selection commands. Those commands that select or exclude data sets to be processed, based on your own specific criteria: SELECT and EXCLUDE.
- Action commands. Those commands that take action upon the selected resources: RECOVER.

The minimum information required for you to execute FMS is a SCAN command to name the DSNINDEX records to be processed, a SELECT command, and an action command.

You can add other commands to gain additional selection or functionality.

FMS Command Sequence

FMS commands are listed in their proper sequence in the following table. Each command is optional, except as described as follows:

Type	Name	Parameters
Environment	SET	MODE=,GEN=,VTOCMSG=,MSGLEVEL=, FROMDATE=,TODATE=,FROMTIME=, TOTIME=
Environment	PASSWORD (VSAM Only)	CATALOG=,MASTER=
Candidate	SCAN	DSNINDEX
Selection	SELECT	DSNAMES=,TABLE=,VOLUMES=,VTOCS, CATALOGS=,CRITERIA=,TDSN=
Selection	EXCLUDE	DSNAMES=,TABLE=,VOLUMES=,VTOCS, CATALOGS=,CRITERIA=,TDSN=
Action	RECOVER	TOVOLUME=,POOL=,DISP=,NEWHLQ=, NEWHLQ2=,SCRATCH,CREATE,PDSECONV,ERASE,NOLOAD,AIX=,TO CAT=,EXPORTF=, DEFALIAS=,REPORT

General Rules Pertaining to the FMS Command Structure

SET and PASSWORD commands are optional. If you do specify a PASSWORD command, it applies to processing of VSAM password-protected clusters only. The SET command must be the first command in the command structure.

Since there is only one SCAN command, DSNINDEX is required, and must be specified before any action command.

SELECT commands are mandatory, while EXCLUDE commands are optional.

Since there is only one ACTION command, RECOVER is required and must be specified for each SCAN command.

SET Command

Use the SET command to define environment values that will be in effect for the duration of this execution only.

Two rules apply to the SET command:

- It must be specified before any other command.
- Set command cannot be specified more than once but multiple parameters can be specified on a single set command.

As with all CA Disk SYSIN commands, the SET command can be commented out (nullified) by placing an asterisk "*" in column 1 of the line containing the command.

SET Command Syntax

The syntax of the SET command is as follows:

SET MODE=, GEN=, VTOCMSG=, MSGLEVEL=, FROMDATE=, TODATE=, FROMTIME=, TOTIME=

MODE=

Specify SIMULATE or SIM to have CA Disk simulate the execution. In simulate mode, CA Disk produces messages and reports as if the supplied action command had really taken place. Specify LIVE, LIV, or remove the SET command entirely to have CA Disk execute the supplied action command. The default value is LIVE.

GEN=

There are two source locations containing data sets names that are available to the SET command:

Data set names and/or patterns specified by the TABLE= parameter on the SELECT command.

Data set names selected by the VTOCS parameter on the SELECT command.

By specifying GEN=YES, CA Disk will print the combined list of data set names to CMDPRINT. By specifying NO, the list of data set names will not be printed. The default value is YES.

Note: The combined list of data set names is the actual candidate list of data sets, not just the data sets that will be processed. Specifically, those data sets not selected for processing will still be listed.

VTOCMSG=

When the VTOCS parameter is specified on the SELECT command, you can optionally have CA Disk generate a message for data sets that are not processed. By specifying VTOCMSG=YES, message 3822 is issued for each data set not recovered. By specifying NO, this message is not issued. The default value is NO.

MSGLEVEL=

This parameter controls the degree (a range from 0 to 9) of debugging desired, recording the output in MSGPRINT.

For instance, a value of '6' will produce a message showing why each data set was or was not selected. The default value is 0, which disables the feature. Other values available for this option include:

0 or 1

Either one will disable the feature 1.

2, 3, 4, or 5

Provides ARCHVOL information.

6, 7, or 8

Displays SELECT/EXCLUDE information in addition to the ARCHVOL information supplied from value 2.

9

Prints all selection control blocks available.

FROMDATE=date,**TODATE=date**

These optional parameters establish a *date window* in which the creation date of each DSNINDEX record is compared to. FROMDATE= defaults to the beginning of time, while TODATE= defaults to today. If the creation date of the DSNINDEX record does NOT fall within this window, the record is excluded from processing. This includes user exits. It will be as if the data set does not exist in the files data set.

For example, using the following command will exclude from further processing any data set that wasn't created in January:

```
//SYSIN DD *  
SET FROMDATE=01JANyyyy,TODATE=31JANyyyy  
SCAN DSNINDEX  
RECOVER SCRATCH  
/*
```

Note: For a list of valid dates, see the section Date Formats in the chapter "General Information."

FROMTIME=time,TOTIME=time

These optional parameters establish a *time window* in which the creation time of each DSNINDEX record is compared to. FROMTIME= defaults to 0000 and TOTIME= defaults to 2359. If the creation time of the DSNINDEX record does NOT fall within this window, the record is excluded from processing. This includes user exits. It will be as if the data set does not exist in the FILES.

For example, using the following command will exclude from further processing any data set that wasn't created between 11PM on March 23 and 2:30PM on March 24.

```
//SYSIN DD *  
SET  MODE=SIM, FROMDATE=23MARxxxx, TODATE=24MARxxxx, TOTIME=1430, FROMTIME=2300  
SCAN DSNINDEX  
RECOVER SCRATCH  
/*
```

PASSWORD Command (For VSAM Only)

This optional command is used to provide CA Disk with any password necessary to carry out the functions requested in the command stream. The command should precede the SCAN command that it is intended to cover.

For clarity, we recommend that you specify this command just after the SET command, if any, and before all other commands.

PASSWORD Command (For VSAM Only) SYNTAX

The syntax of the PASSWORD command is as follows:

PASSWORD MASTER=,CATALOG=

This optional parameter MASTER= is used to supply CA Disk with the proper level of password it will need to carry out its requested functions. You can supply any of the following passwords:

- Catalog master password
- Cluster master password
- CA Disk master password

This optional parameter CATALOG= is used to supply CA Disk with the catalog name to which restored clusters are to be cataloged.

SCAN Command

The SCAN command is used to tell CA Disk to scan the DSNINDEX subfile. A SCAN command must precede any action command. Any SELECT and EXCLUDE commands must follow the SCAN command and precede the action command.

SCAN Command SYNTAX

The syntax of the SCAN command is as follows:

```
SCAN DSNINDEX
```

This REQUIRED parameter DSNINDEX causes CA Disk to scan the records located in the DSNINDEX subfile.

SELECT Command

This required command allows a user to process only those data sets that satisfy the specified selection test(s).

If a SELECT statement is specified, in order for a data set to be selected, ALL tests specified on a single SELECT command must be met. If only one parameter is specified, only that single test must be met for selection to take place.

For example, if the DSNAMES= parameter or the VOLUMES= parameter, but not both, are specified on a single command, conceptually think *process only these data sets* or *process only these volumes*. If both the DSNAMES= and VOLUMES= parameters are specified, conceptually think *process only these data sets if residing on one of these volumes*.

If multiple SELECT commands are specified for an action command, this has the effect of a logical *OR* between SELECT commands. That is, if any SELECT command is true, the data set is processed, regardless of whether other SELECT commands test true or not.

SELECT Command SYNTAX

The syntax of the SELECT command is as follows:

```
SELECT DSNAMES=, TABLE=, VOLUMES=, TDSN=, CATALOGS=, VTOCS, CRITERIA=
```

DSNAMES=

You can specify up to 30 fully qualified names, 30 CA Disk pattern names, or a combination of both. If specified, only those data sets matching any of the names or patterns will be candidates for recovery. This parameter has two default values.

If parameter CATALOGS= is not present on the SELECT command, the default is 'DSNAMES=/', meaning *all* data sets.

If parameter CATALOGS= is present on the SELECT command, the default of the DSNAMES= parameter is *null*, meaning *no* data sets.

Note: Both DSNAMES= and CATALOGS= parameters can be specified on the same SELECT command.

CATALOGS=

You can specify a list of up to 30 catalog names, 30 CA Disk pattern names, or a combination of both. If specified, only those catalogs matching any of these names or patterns will be candidates for recovery. If CATALOGS= is not specified, the default value is *null*, meaning *no* catalogs will be recovered. For more information, see previous DSNAMES= parameter.

TABLE=

You can specify a name of a member in PARMLIB containing the list (table) of fully qualified data set names or patterns. If specified, only those data sets matching any of the names or patterns in the table will be candidates for recovery. The format of the table is described in the section Selection Lists—Generalized in the chapter CA Disk Libraries in the *Systems Guide*.

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 in the *Systems Guide*.

VOLUMES=

You can specify a list of up to 50 volume serial numbers, 50 CA Disk pattern names or a combination of both where the backed up and archived data sets resided. If specified, only data sets that were backed up or archived from the specified list will be candidates for recovery. The default value is '/', meaning *all* volumes.

TDSN=

You can specify a list of up to 30 archive volume data set names, 30 CA Disk pattern names, or a combination of both. If specified, only those data sets that were backed up or archived onto an ARCHVOL matching this naming convention will be candidates for recovery.

VTOCS

Causes CA Disk to recover data sets based on a VTOC image created during BACKUP. The zero version backup copy of the VTOC is read for all volumes specified on the VOLUME= parameter; and a list is built of candidate data sets.

Note: If you are familiar with the RECOVER PROC, you will notice a significant difference in the way FMS processes this VTOC image. With the RECOVER PROC, all data sets backed up from the volume are recovered, as long as they have corresponding DSNINDEX records. The FMS PROC simply uses the VTOC image as another table of data set names to do selection (or exclusion) upon. It is for this reason that SET VTOCMSGS= defaults to NO.

If VSAM datasets are to be recovered using this process then Sysparm VBRIVSAM must be specified with a value of Y at the time that the backup which includes the VTOC is taken.

CRITERIA=

You can specify a series of *criteria tests* to limit the selection of data sets or catalogs to be considered as candidates for recovery. The criteria parameter is specified by a minimum of three subparameters as follows:

(operand1,operator,operand2)

The operands can consist of any combination of the criteria subparameters described in FMS Selection Criteria Field Names. The maximum number of subparameters allowed on a single SELECT or EXCLUDE command is 256. Because the CRITERIA= parameter is already very restrictive, you cannot specify the VTOCS parameter on the same SELECT command. You can, however, add the DSNAME=, TABLE=, VOLUMES=, or TDSN= parameters. For logical operators, you can specify any of the values listed in the section Criteria Operators and their Meanings in the chapter "Basic System – DMS PROC."

Criteria Statement Operands

For operands, you can specify any of the following:

- Decimal number constants, such as 10 or 1000.
- Date constants in any acceptable CA Disk format, such as 96200, 96.200, 96/200, 19JUL96, 07/19/96, 1996200, 1996.200, 19JUL1996, 1996/200 or 07/19/1996. If CA Disk has trouble differentiating between a digit Julian date constant, such as 96200, and a decimal number constant, it will issue an appropriate message and ask you to use another format.
- Constant values TRUE, FALSE, YES, NO, ON, or OFF. TRUE, YES, and ON are all equivalent. FALSE, NO, and OFF are all equivalent.

- Expressions that use addition (+) or subtraction (-) to combine a constant value with a field name or other constant value. For example, the expression TODAY-30 represents the date 30 days ago.
- Any of the DSCL select criteria field names listed in the following section.

FMS Selection Criteria Field Names

The following table shows the FMS Select Criteria field name, listed alphabetically.

ALLOBLKS

Allocated in blocks

ALLOCYLS

Allocated in cylinders

ALLOPRI

Primary allocation amount

ALLOTRKS

Allocated in tracks

ARCDT

Backup/archive date

ARCTIME

Backup/archive time

ARCTYPE

Type of DSNINDEX record

CREDIT

Creation date in the format-1 DSCB prior to archival

DCNAME

Data class name <character format>

DSNKEY

ARCHIVE KEY

DSORG

Data set organization

EXPDT

Expiration date in the format-1 DSCB prior to archival

JOBNAME

Jobname in the format-1 DSCB prior to archival

KBYTES

Kilobytes allocated to data set

MCNAME

Management class name <character format>

MODDT

Last modification date in format-1 DSCB prior to archival

PARTBKUP

Recover partially backed up PO data sets

RECAT

Recataloged to CA Disk Pseudo volume

RESDT

Restore date in DSNINDEX record

RESTIME

Restore time in DSNINDEX record

SCNAME

Storage class name <character format>

SGNAME

Storage group name <character format>

TODAY

Today's date

USEDT

Last use date in format-1 DSCB prior to archival

USERFLD

User field information in DSNINDEX record

VERSION

Version number of the DSNINDEX record

VERSIONARC

Version number of Archive DSNINDEX record

VERSIONBKP

Version number of Backup DSNINDEX record

The following describes each field name in detail.

ALLOBLKS

This criterion allows for the selection of DSNINDEX records that were allocated in blocks at the time the data set was backed up or archived. For VSAM, this selection criteria test is based on the data component information. For detail information about when this condition is true, see the section ALLOBLKS – Allocated in blocks (yes/no) in the chapter "Basic System – DMS PROC."

For example, to select only those DSNINDEX records that were allocated in blocks at backup time:

```
CRI=(ALLOBLKS,EQ,YES)
```

ALLOCYLS

This criterion allows for the selection of DSNINDEX records that were allocated in cylinders at the time the data set was backed up or archived. For VSAM, this selection criteria test is based on the data component information.

For detail information about when this condition is true, see the section ALLOCYLS – Allocated in cylinders (yes/no) in the chapter "Basic System – DMS PROC." For example, to select only those DSNINDEX records that were allocated in cylinders at backup time:

```
CRI=(ALLOCYLS,EQ,YES)
```

ALLOPRI

The value of ALLOPRI is set to the first extent as specified in the DSNINDEX record. Since the actual primary request value is not recorded in the VTOC, the size of the first extent is displayed. The first extent size is calculated from information in the format-1 DSCB. Unless the data set is very large, or the free space on a volume is badly fragmented, the first extent will accurately reflect the primary space. The value must be specified as a numeric value.

For more information, see the sysparm description for ALLOCPRI in the *Systems Guide*.

For example, to select only those DSNINDEX records that have a primary extent smaller the 15 tracks:

```
CRI=(ALLOPRI,LT,15,AND,ALLOTRKS,EQ,YES)
```

Note: If sysparm PRIALLOC is left at its default value, Recover processing will consolidate the original extents during allocation. This can increase the number of data sets selected using ALLOPRI. For more information, see the sysparm description for PRIALLOCn in the *Systems Guide*.

ALLOTRKS

This criterion allows for the selection of DSNINDEX records that were allocated in tracks at the time the data set was backed up or archived. For VSAM, this selection criteria test is based on the data component information.

For detail information about when this condition is true, see the section ALLOTRKS – Allocated in tracks (yes/no) in the chapter "Basic System – DMS PROC." For example, to select only those DSNINDEX records that were allocated in tracks at backup time:

```
CRI=(ALLOTRKS,EQ,YES)
```

ARCDT

This criterion allows for the selection of DSNINDEX records based upon their Written Date stamps. This parameter is functionally equivalent to the DATE= and EDATE= parameters used by the RECOVER PROC. Any valid CA Disk date format is supported, including TODAY. For more information, see ARCTIME.

Note: For a list of valid dates, see the section Date Formats in the chapter "General Information."

For example, to select only those DSNINDEX records that were written in the last 7 days:

```
CRI=(ARCDT,GE,TODAY-7)
```

ARCTIME

This criterion allows for the selection of DSNINDEX records based upon their Written Time stamps. This parameter is functionally equivalent to the TIME= and ETIME= parameters used by the RECOVER PROC.

For example, to select DSNINDEX records that were written on January 22, 1996 between the hours of 11AM and 11PM:

```
CRI=(ARCDT,EQ,96022,AND,ARCTIME,GT,1100,AND,ARCTIME,LT,2300)
```

ARCTYPE

This criterion allows for the selection of DSNINDEX records based upon whether they are an archive or backup copy of the data set. Valid operands are: BACKUP or ARCHIVE.

For example, to select only those DSNINDEX records that were created as a result of executing the BACKUP command:

```
CRI=(ARCTYPE,EQ,BACKUP)
```

Important! Using this example without any other criteria parameters can potentially *down-level* a data set. This will happen if version zero of the DSNINDEX record being recovered was created by the ARCHIVE command.

For more information, see VERSION.

CREDIT

This criterion allows for the selection of DSNINDEX records based upon the Format-1 DSCB creation date of the data set prior to being archived. Any valid CA Disk date format is supported, including TODAY.

Note: For a list of valid dates, see the section Date Formats in the chapter "General Information."

For example, to select only those DSNINDEX records of data sets created in the last three months:

```
CRI=(CREDIT,GT,TODAY-90)
```

DCNAME

This criterion allows for the selection of DSNINDEX records based upon the DFSMS Data Class name of the data set.

For example, to select only those DSNINDEX records of data sets that were assigned to the STANDARD Data Class:

```
CRI=(DCNAME,EQ,<STANDARD>)
```

DSNKEY

For example, to select those DSNINDEX records that have DISK12 as their ARCHIVE KEY:

This criterion allows for the selection of DSNINDEX records based upon their ARCHIVE KEY. This value must be enclosed in < >, and can represent a tape, cartridge, or disk archive. For an example of an ARCHIVE KEY, see the section LISTV Sample Report in the chapter "FILES Maintenance."

```
CRI=(DSNKEY,EQ,<DISK12>)
```

DSORG

This criterion allows for the selection of DSNINDEX records based upon the value in its DSORG field. For a list of valid values see the section DSORG – Data Set Organization in the chapter "Basic System – DMS PROC."

For example, to select only those DSNINDEX records that have AM in their DSORG field:

```
CRI=(DSORG,EQ,AM)
```

Note: This example is functionally equivalent to specifying VSAMONLY from the RECOVER PROC.

EXPDT

This criterion allows for the selection of DSNINDEX records based upon the Format-1 DSCB expiration date of the data set prior to being archived. Any valid CA Disk date format is supported, including TODAY.

Note: For a list of valid dates, see the section Date Formats in the chapter "General Information."

For example, to exclude only those DSNINDEX records that contain data sets which have Format-1 DSCB expiration dates between January 1 and July 1, 1996:

```
CRI=(EXPDT,GT,1/1/96,AND,EXPDT,LT,7/1/96)
```

JOBNAME

This criterion allows for the selection of DSNINDEX records based on the name of the job in the Format-1 DSCB of the data set prior to being archived. Depending of the OPEN SVC option chosen, this would either be the creating, modifying, or last using JOBNAME, or an account number. For information on OPEN SVC options, see The CA Disk SVC in the *Installation Guide*.

For example, to select only those DSNINDEX records that contain SB15BKUP in their JOBNAME field:

```
CRI=(JOBNAME,EQ,<SB15BKUP>)
```

KBYTES

This criterion allows for the selection of DSNINDEX records that contained nnnnn kilobytes at the time the data set was backed up or archived.

For example, to select only those DSNINDEX records that contain data sets that were larger than 50000 kilobytes at backup time:

```
CRI=(KBYTES,GT,50000)
```

MCNAME

This criterion allows for the selection of DSNINDEX records based upon the DFSMS Management Class name of the data set.

For example, to select only those DSNINDEX records of data sets that were assigned to the GDGS Management Class:

```
CRI=(MCNAME,EQ,<GDGS>)
```

MODDT

This criterion allows for the selection of DSNINDEX records based upon the Format-1 DSCB last modification date of the data set prior to being archived. Any valid CA Disk date format is supported, including TODAY.

Note: For a list of valid dates, see the section Date Formats in the chapter "General Information."

For example, to select only those DSNINDEX records that contain data sets last modified on can 2, 1993:

```
CRI=(MODDT,EQ,1993123)
```

PARTBKUP

Default processing excludes partially backed up PO data sets from being selected for recovery. Instead, FMS selects the latest non-partially backed up PO. The objective of this criterion is to recover partially backed up PO's.

For example, to include partially backed up PO's during recover, add the following criteria statement to your command stream:

```
CRI=(PARTBKUP,EQ,YES)
```

RECAT

This criterion allows for the selection of DSNINDEX records based on whether the data set was recataloged to the pseudo-volume at archive time. Valid values are: YES or NO.

For example, to select only those DSNINDEX records that contain data sets cataloged to ARCIVE:

```
CRI=(RECAT,EQ,YES)
```

RESDT

This criterion allows for the selection of DSNINDEX records based upon the Last Restore Date of the DSNINDEX record. It will only contain a valid date if sysparm SMSUPDAT (described in SMSUPDATy in the *Systems Guide*) is at the default value of 'Y'. This criterion can be used to restart a failed RECOVER job by placing it on an EXCLUDE statement in conjunction with the VERSION criteria. Any valid CA Disk date format is supported, including TODAY.

Note: For a list of valid dates, see the section Date Formats in the chapter "General Information." For other related information, see VERSION.

For example, the typical method of restarting a RECOVER that failed:

```
EXCLUDE CRI=(RESDT,EQ,TODAY,OR,VERSION,NE,0)
```

Important! Be aware that VERSION should always be used in conjunction with RESDT. This is because ALL versions (even back level copies) are candidates to FMS unless specifically excluded or disallowed by selection criteria.

RESTIME

This criterion allows for the selection of DSNINDEX records based upon the Last Restore Time of the DSNINDEX record. It will only contain a valid time if sysparm SMSUPDAT (described in SMSUPDATy in the *Systems Guide*) is set to Y.

Note: For more information, see VERSION.

For example, to select only those DSNINDEX records that were restored yesterday between 8AM and 5PM:

```
CRI=(RESTIME,GT,0800,AND,RESTIME,LT,1700,AND,RESDT,EQ,TODAY-1)
```

SCNAME

This criterion allows for the selection of DSNINDEX records based upon the DFSMS Storage Class name of the data set.

For example, to select only those DSNINDEX records of data sets that were assigned to the EXTRA Storage

Class:

```
CRI=(SCNAME,EQ,<EXTRA>)
```

SGNAME

This criterion allows for the selection of DSNINDEX records based upon the DFSMS Storage Group name of the data set.

For example, to select only those DSNINDEX records of data sets that were assigned to the PLENTY Storage Group:

```
CRI=(SGNAME,EQ,<PLENTY>)
```

TODAY

This value is derived from the system clock. Comparison dates used in conjunction with this criterion can be entered in any acceptable CA Disk format.

Note: For a list of valid dates, see the section Date Formats in the chapter "General Information."

USEDT

This criterion allows for the selection of DSNINDEX records based upon the Format-1 DSCB last use date of the data set prior to being archived. Any valid CA Disk date format is supported, including TODAY.

Note: For a list of valid dates, see the section Date Formats in the chapter "General Information."

For example, to select only those DSNINDEX records that contain data sets which were last used on November 1, 1994:

```
CRI=(USEDT,EQ,11/01/94)
```

If you want to select or exclude entries with zero or invalid USEDTs, use 0, 'zero', or 2155.366 as the comparison value.

```
CRI=(USEDT,EQ,0)
```

USERFLD

This criterion allows for the selection of DSNINDEX records based upon the information contained within the 25 byte user field. The comparison is made against the left justified field, and blanks and non-printable Hex characters are not allowed. For more information, see Subfile Record Format — (DSNINDEX) and User Exit DINXUFEX — Update DSNINDEX User Field in the *Systems Guide*.

For example, to select only those DSNINDEX records that contain Acct#18753 in the User Field:

```
CRI=(USERFLD,EQ,<Acct#18753>)
```

VERSION

This criterion allows for the selection of DSNINDEX records based upon the relative version number of the data set. Both archive and backup copies are considered when calculating the version number, and it can be used to restart a failed RECOVER job. For details and an example, see RESDT. Valid values are: Negative numeric or zero—the most current version.

For example, to select the latest backup of a data set that has not been archived:

```
CRI=(VERSION,EQ,0,AND,ARCTYPE,EQ,BACKUP)
```

Or, in a recover restart situation, consider the following statement:

```
EXCLUDE CRI=(RESDT,EQ,TODAY,AND,VERSION,NE,0)
```

Note: In the second example, there are two important items to point out: First, the value of RESDT is very important, especially if the recover began before midnight, and abended after midnight. If this is the case, the appropriate date would be *GE,TODAY-1*. Second, if VERSION was not specified as shown, there is a possibility of *down-leveling* a data set by mistake by selecting a non-zero version of the DSNINDEX record.

VERSIONARC

This criterion allows for the selection of DSNINDEX records based upon the relative version number of the data set. Only archive copies are considered when calculating the version number.

Important! Since this criterion automatically excludes backup versions, you can potentially *down-level* a data set that has a more current backup version in the archives.

Valid values are: Negative numeric or zero—the most current version.

For example, to select the -1 version of an archived copy, ignoring all backup copies, even a more current one:

```
CRI=(VERSIONARC,EQ,-1)
```

VERSIONBKP

This criterion allows for the selection of DSNINDEX records based upon the version number of the data set. Only backup copies are considered when calculating the version number.

Important! Because this criterion automatically excludes archive versions, you can potentially *down-level* a data set that has a more current archive version in the archives.

Valid values are: Negative numeric or zero—the most current version.

For example, to select the third oldest backup version, ignoring all archive copies, even a more current one:

```
CRI=(VERSIONBKP,EQ,-2)
```

FMS Select Criteria Boolean (AND/OR) Logic

The Boolean logic support of the CRITERIA parameter is invoked by specifying either an AND or an OR between conditional tests. Remember that a conditional test consists of an operand, an operator, and a second operand. You can specify a maximum of 256 conditional tests on a single CRITERIA parameter when each is separated by a compound condition.

For several examples of valid Boolean tests, see the section DSCL Select Criteria Boolean (AND/OR) Logic in the chapter "Basic System – DMS PROC."

EXCLUDE Command

The EXCLUDE command allows you to exclude data sets from processing, and it takes precedence over SELECT commands. For instance, if the data set being processed is excluded by any EXCLUDE command, the SELECT of the data set will not occur. In order for a data set to be excluded, all tests specified on a single EXCLUDE command must be met. If only one parameter is specified, only that single test must be met for the exclusion to be honored. If the EXCLUDE command is specified more than once for any given ACTION command, it has the effect of a logical *OR* between EXCLUDE commands. That is, if any EXCLUDE command is true, then the data set is bypassed, regardless if other EXCLUDE commands test true or not.

For example, if the DSNAMES= parameter or the VOLUMES= parameter, but not both, are specified, conceptually think *don't process these data sets* or *don't process these volumes*. If both the DSNAMES= and VOLUMES= parameters are specified on a single EXCLUDE command, conceptually think *don't process these data sets if they reside on these volumes*.

All parameters that can be specified on the SELECT command are valid parameters for the EXCLUDE command. They operate in exactly the same fashion and with the same defaults.

Syntax

The syntax of the EXCLUDE command is as follows:

```
EXCLUDE DSNAMES=, TABLE=, VOLUMES=, TDSN=, CATALOGS=, VTDCS, CRITERIA=
```

All syntax that can be specified on the SELECT command are valid syntax for the EXCLUDE command.

ACTION Commands

The command RECOVER is known as an action command because it causes CA Disk to take a specific action against a selected data set or volume. This command is required for the FMS proc to function properly. The RECOVER command and its parameters are described in the section RECOVER Command in the chapter "RESTORE/RECOVER."

Chapter 4: FMSGEN PROC

This section contains the following topics:

[About FMSGEN](#) (see page 167)

[REXX Program—ADSRX010](#) (see page 167)

[Sample Templates](#) (see page 168)

About FMSGEN

This PROC provides the ability of generating multiple restore jobs from a single FMS DSCL request. However, because this PROC uses the combination of a REXX program and a user-defined template, it can produce jobs for a wide variety of purposes, not just restore.

REXX Program—ADSRX010

FMSGEN uses a REXX program, ADSRX010, which combines DSNINDEX records and a user-defined template to generate reports, output files, and or JCL statements. Using this technique, you can design a template for virtually any type of output, including the ability to generate multiple restore jobs.

Logic Summary

The JCL procedure for this REXX program consists of the following steps:

1. Invokes the REXX program and outputs four sequential files from information contained in the @JOB() section:
 - Specified system parameters
 - FMS selection criteria used in step 2
 - Sort control statements for step 3
 - Sort control statements for step 5
2. A simulated FMS recover runs. It reads the DSNINDEX subfile and performs record selection based on the information gathered in step 1.
3. The DSNINDEX records are sorted controlled by sort information specified in step 1, using primary ARCHVOL key if DSNKEY was specified.
4. The execution step that processes the DSNINDEX records and adds a new field with the ARCHVOL key to be used based on sysparms RESCOPYT and RESCOPYS.

5. The extended DSNINDEX records are sorted controlled by sort information specified in step 1, using the ARCHVOL key to be used according to SYSPARMS RESCOPYT and RESCOPYS settings, if DSNKEY was specified.
6. The execution step that processes the sorted DSNINDEX file. Using the @RPT and @JCL sections of the template, it reads the extended DSNINDEX records and creates a report along with JCL to invoke one restore per archive volume.

Sample Templates

There are several sample templates located in the INSTALL library that is shipped with CA—Disk. These templates utilize variables defined in the REXX program that can be used as is, or customized to suit special needs.

The main sections within the templates that control processing options are:

- @JOB ()—This section defines system parameters, DSCL selection criteria, sort options, and control break information. For a detailed explanation, see @JOB (OPTION, OUTPUTDD in this chapter.
- @JCL ()—This section details skeleton JCL statements used to generate the desired output. For a detailed explanation, see @JOB (OPTION, OUTPUTDD in this chapter.
- @RPT ()—This section defines report titles, report headings, report totals, footers, and the format to use for detail and content lines. For a detailed explanation, see the section Creating Customized Reports in this chapter.

The following templates are discussed in this section:

- FMSSAMP1—Demonstrates the ability of creating multiple RESTORE jobs.
- FMSSAMP2—Demonstrates how you can submit a separate RESTORE job for each ARCHVOL tape volume. Also generates a report showing which tape volumes will be needed.
- FMSSAMP3—Demonstrates the ability of creating IMPLICIT IXUPDATE commands.

FMSSAMP1

This template, once customized to conform to site standards, can create multiple restore jobs.

The following illustration is a sample INSTALL Member —FMSSAMP1.

```

EDIT IISK988.QUAL.INSTALL (FMSSAMP1) - 01.03      Column 00001 00072
Command ==>                                     Scroll ==> CSR
000004 * + + + + +      J O B      S E C T I O N      + + + + + + + +
000005 @JOB (OPTION, OUTPUTDD = "SIMULATE")
000006 @JOB (SELECTION)
000007     SELECT  ISN=/,CRITERIA= (DSNKEY,NE,<0/>,AND,ARCTYPE,EQ,BACKUP)
000008 @JOB (SORT)
000009     DSNKEY, FILENO
000010 * + + + + +      J C L      S E C T I O N      + + + + + + + +
000011 @JCL (BEGIN)
000012 //PDSJOB1X JOB (ACCT,XXXXXX,XXX,00),'ACCT NAME',CLASS=C,MSGCLASS=X,
000013 // NOTIFY=USERID,MSGLEVEL=(1,1)
000014 //STEP20  EXEC PCN=IEBUPITE,PARM=MOD
000015 //SYSPRINT DD  SYSOUT=*
000016 //SYSUT2  DD  DISP=SHR,ISN=USERID.PDS
000017 //SYSUT1  DD  DISP=SHR,ISN=USERID.PDS
000018 //SYSIN   DD  DATA,DLN='$$'
000019 @JCL (BEGIN, ISNKEY)
000020 . /      ADD NAME=JXDSNKEY,LIST=ALL
000021 //JXDSNKEY JOB (ACCT,XXXXXX,XXX,00),'ACCT NAME',CLASS=C,MSGCLASS=X,
000022 // NOTIFY=USERID1,MSGLEVEL=(1,1)
000023 //S%ISNKEY EXEC RESTORE
  
```

Notes: For Sample INSTALL Member:

- @JOB (OPTION, OUTPUTDD—This statement controls what type of output is desired.
- The presence of *SIMULATE* causes a simulation of any selection criteria specified. This would be the equivalent (meaning the end result would be the same) of executing FMS in simulate mode with one big difference—the FMMSGEN job splits the selection criteria into separate restore jobs, one job per ARCHVOL. For an illustration of this, see FMSSAMP1.
- Another OUTPUTDD option available is *INTRDR*—The presence of this option immediately executes the JCL SECTION of the template.
- @JOB (SELECTION)—FMS criteria field names.
- These are the standard field names described in the section FMS Selection Criteria Field Names in the chapter "Basic System – FMS PROC."
- @JOB (SORT)—This is where you define the sort options and control break information.
- @JCL (BEGIN)—JCL statements to execute when *INTRDR* is supplied in the JOB SECTION.

The example shows IEBUPDTE control statements, but virtually any command statements can be coded here. By changing *SIMULATE* to *INTRDR* and supplying a dsname to SYSUT1 and SYSUT2, the IEBUPDTE control statements are invoked creating a member in the SYSUT2 data set for each ARCHVOL required to satisfy the selection criteria. For an illustration of this, see Sample IEBUPDTE output from FMSSGEN.

Sample output from FMSSAMP1 is the result of 2 separate jobs (the first is FMS, the second is FMSSGEN) using identical criteria. The criteria used was:

```
SELECT DSN=/, CRITERIA=(DSNKEY, NE, <@/>, AND, ARCTYPE, EQ, BACKUP)
```

The following illustration is a sample output from FMSSAMP1:

```

EDIT      QDMSTEST.SMP900.INSTAL (FMSSAMP2) - 01.00      Columns 00001 00072
Command ==> _____ Scroll ==> CSR
000024  ----- RPT      SECTION -----
000025  @RPT (TITLE)
000026  Multi-task Recover
000027  Generate a separate job per archive volume
000028  Each Job consists of explicit Restore requests
000029  for each data set on a given archive volume
000030  @RPT (HEADER)
000031
000032  The following is a list of the required tape archive volumes
000033
000034  DSNKEY  #Datasets
000035  -----
000036  @RPT (END, DSNKEY)
000037  %dsnkey  %count
000038  @RPT (TOTAL)
000039
000040  %count  Datasets Processed.
000041  JOB (FMSSAMP2)
000042  Date (%today) Time (%Time)
000043  Page# %Page
  
```

Notes:

- The top half of the SYSOUT is an FMS job, resulting in 1 tape drive being used for the duration. If you were to scroll down to the bottom, the criteria used here would total 196 data sets on 6 different ARCHVOLS.
- The identical selection criteria were used to create the bottom half with FMSSGEN. It's important to realize that with FMSSGEN, even though the same 196 data set s will be restored, it's a much more efficient process because the job is automatically split up by ARCHVOL. In this particular case, the 196 data set s will be restored by 6 different jobs executing concurrently.

Creating Multiple RESTORE Jobs

Having the ability of creating multiple RESTORE jobs will benefit any Storage Management department. But the efficient manner in which FMMSGEN accomplishes this task will also benefit Disaster Recovery testing.

Using FMSSAMP1 as the template (see Sample INSTALL Member), customize the following areas:

- @JOB (OPTION, OUTPUTDD = "SIMULATE")— Change SIMULATE to INTRDR
- @JOB (SELECTION)— Replace the supplied criteria statements as desired
- @JCL (BEGIN)— Alter the IEBUPDTE statements as desired. Also change SYSUT1 and SYSUT2 to a PO data set. This data set is used to store the generated RESTORE jobs.

After this is complete, executing FMMSGEN will result in output similar to the following:

The screenshot displays the FMMSGEN output window. The top section shows a list of jobs under the heading 'ISPDLH1.FMS.OUTPUT'. A red box highlights the job 'J208259'. A red arrow points from a text box 'Value specified for SYSUT1 and SYSUT2' to the job name. Another red arrow points from a text box '6 members created in the SYSUT2 data set whose names are created based on the information given to IEBUPDTE. These 6 ARCHVOLS are the results from the supplied FMS selection criteria' to the job list. The bottom section shows the JCL statements for job 'J208259'. A red box highlights the job name in the header. A red arrow points from a text box 'JCL statements created by the IEBUPDTE control statements' to the JCL statements.

```

File Edit Transfer Options Connection Macro Window Help
Menu Functions Utilities Help

VIEW      ISPDLH1.FMS.OUTPUT                               Row 00001 of 00006
Command ==>                                         Scroll ==> CSR
Name      UU MM Created   Changed   Size Init Mod ID
. J200982
. J202059
. J204323
. J207298
s J208259
. J208674
**End**

File Edit Confirm Menu Utilities Compilers Test Help

EDIT      ISPDLH1.FMS.OUTPUT (J208259) - 01.00       Column 00001 00072
Command ==>                                         Scroll ==> CSR
***** ***** Top of Data *****
000001 //J208259 JOB (SMDSK,27930Z,SSS,38), 'ACCT NAME',CLASS=C,MSGCLASS=X,
000002 // NOTIFY=USERID1,MSGLEVEL=(1,1),TYPRUN=HOLD
000003 //S208259 EXEC RESTORE
000004 //SYSIN DD *
000005 RESTORE DSN=QDMSPROD.MODMS.CBT.03J
000006 RESTORE DSN=LABIUPS.JEG.TEST.SMS2

Aa A TCPIP R 10 C 4 15:14 8/11/98
  
```

FMSSAMP2

This template, once customized to conform to site standards, can submit a separate RESTORE job for each ARCHVOL tape volume. It also demonstrates the process you need to follow in order to generate a report showing which tape volumes will be needed.

Since the multiple RESTORE job section of this template was covered using FMSSAMP1 in Creating Multiple RESTORE Jobs, the focus of this discussion will be on generating the report.

The following illustration is a Sample INSTALL Member – FMSSAMP2 which is the second half of the template, the RPT SECTION.

```

EDIT      QDMSTEST.SMP900.INSTALL (FMSSAMP2) - 01.00      Columns 00001 00072
Command ===>
000024 ----- R P T      S E C T I O N -----
000025 @RPT (TITLE)
000026 Multi-task Recover
000027 Generate a separate job per archive volume
000028 Each Job consists of explicit Restore requests
000029 For each data set on a given archive volume
000030 @RPT (HEADER)
000031
000032 The following is a list of the required tape archive volumes
000033
000034 DSNKEY Datasets
000035 -----
000036 @RPT (END, DSNKEY)
000037 %dsnkey %count
000038 @RPT (TOTAL)
000039
000040 %count Datasets Processed.
000041 JOB (FMSSAMP2)
000042 Date (%today) Time (%Time)
000043 Page# %Page
  
```

Notes:

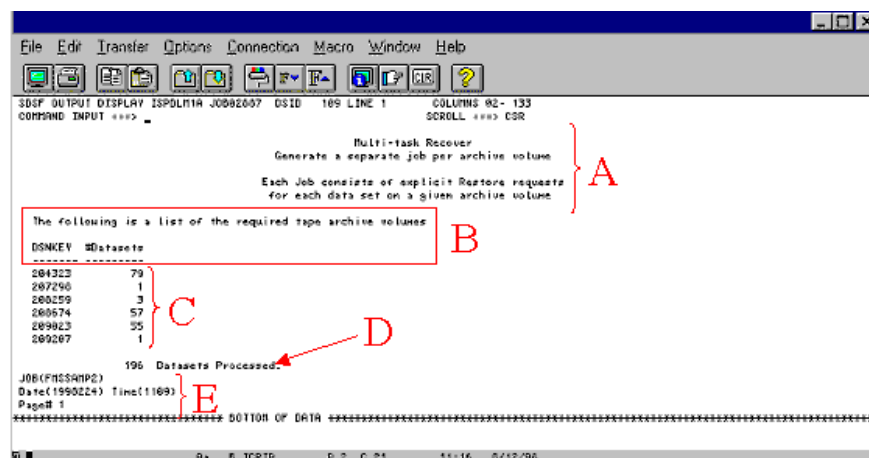
- Member is split up into 5 parts labeled A-E. Each of these labels corresponds to sections within the sample report illustrated in Sample INSTALL Member ? FMSSAMP2.

Creating Customized Reports

Using templates to generate reports is very useful. With FMSSAMP2 as the template see Sample INSTALL Member ? FMSSAMP2, customize the following areas:

- @RPT (TITLE)—Define the report title here. It can be as many lines as desired, and if you want a blank line simply add it to the template. Note that title lines default as center aligned.
- @RPT (HEADER)—Define headings here. Header lines default to left justified.
- @RPT ()—Define the detail lines as needed. For a list of possible detail line values that can be used as symbolic substitutions, see List of Symbolic Substitution Values.

After this is complete, executing FMMSGEN will result in output similar to the following illustration which is a Sample output from RPT SECTION of FMSSAMP2.



Notes:

- A—Defined in the @RPT (TITLE) statement.
- B—Defined in the @RPT (HEADER) statement.
- C—Created using the %DSNKEY and %COUNT symbols. A complete list of the available symbolic values can be found in
- D—Created using the %COUNT symbol.
- E—Created using the %DATE, %TIME, and %PAGENO symbols. These are calculated fields that are available as symbolic substitutions. Symbolic Substitutions for Templates.

You can use several symbolic substitutions within a template. Using these substitutions make templates more powerful. These symbolics are divided into three categories as follows:

Substitution symbolics

%ALLOBLKS

Allocated in blocks Yes/No.

%ALLOCYLS

Allocated in Cylinders Yes/No.

%ALLOPRI

Primary Allocation Amount.

%ALLOTRKS

Allocated in Tracks Yes/No.

%ALLOTYPE

Allocated in *BLKS/TRKS/CYLS*.

%ARCDT

Backup/Archive Date—format YYYYddd.

%ARCTIME

Backup/Archive Time—format HHMM.

%ARCTYPE

Type of DSNINDEX—BACKUP/ARCHIVE.

%CREDIT

Creation date from the format-1 DSCB at the time of archival—format YYYYddd.

%DCNAME

SMS Data Class Name.

%DSNAME

Data set Name.

%DSNKEY

Archive Volume or Key.

%DSORG

Data set Organization.

%EXPDT

Expiration date from the format-1 DSCB at the time of archival—format YYYYddd.

%FILENO

Logical file sequence within the archive volume.

%JOBNAME

Jobname from the format-1 DSCB (prior to archival).

##BYTES

Number of bytes archived.

##KBYTES

Number of bytes archived / 1000.

%KBYTES

Number of bytes archived or backed up divided by 1000.

%MCNAME

SMS Management Class Name.

%MODDT

Last modification date from the format-1 DSCB (prior to archival)—format YYYYddd.

%RECAT

(Y)es/(N)o Flag indicating if the was re-cataloged to ARCIVE.

%RESDT

Restore date in the DSNINDEX—format YYYYddd.

%RESTIME

Restore time in the DSNINDEX—format HHMM.

%SCNAME

SMS Storage Class Name.

%SGNAME

SMS Storage Group Name.

%TIME

Current Time.

%TODAY

Today's Date.

%USEDT

Last used date from the format-1 DSCB (prior to archival)—YYYYddd.

%USEKEY

Archive Volume or Key to be used, based on SYSPARMS RESCOPYT and RESCOPYS settings as well as flag ARCFDISA of the primary ARCHVOL.

%USERFLD

User field in the DSNINDEX.

Calculated symbolics

%COUNT

Records processed at the current break level.

%DATE

Current date character format also see TODAY.

%PAGENO

Current page number.

%RECNO

Current input record number.

%TIME

Current Time HHMM.

%TODAY

Current date YYYYddd format.

Special symbolics

%SETNDX

This variable's value is incremented every time its symbolic name %SETNDX is referenced

%CURNDX.

Allows you to reference the current value of SETNDX without incrementing its value.

%N0...N9

User defined Numeric Fields.

%C0...C9

User defined Character fields.

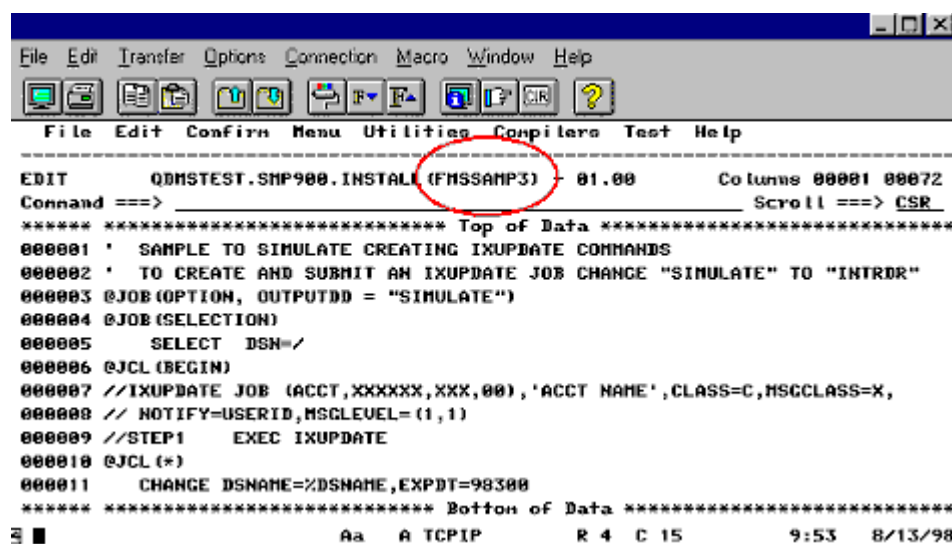
Note: In order to utilize the ADSAX010 keyword USEKEY and for FMMSGEN to work correctly with sysparm RESCOPYTy, see these changes in your JCL:

```
./          ADD  NAME=J%USEKEY,LIST=ALL
//J%USEKEY JOB (ACCT,XXXXXX,XXX,00), 'ACCT NAME', CLASS=C,MSGCLASS=X,
// NOTIFY=USERID1,MSGLEVEL=(1,1)
//S%USEKEY EXEC RESTORE
```

FMSSAMP3

This template, once customized to conform to site standards, demonstrates how to create implicit IXUPDATE commands. The default for @JOB (OPTION, OUTPUTDD) is *SIMULATE*, so when you are ready for the real run, simply change *SIMULATE* to *INTRDR*.

The following illustration is the sample FMSSAMP3 from the INSTALL library.



Notes:

- Change the @JOB (SELECTION) as desired
- Provide proper JOBCARD information for @JCL (BEGIN)
- If you change SIMULATE to INTRDR in the @JOB (OPTION) section, the provided IXUPDATE job will begin executing immediately after the FMMSGEN job completes. If you would rather have the IXUPDATE job be stored into a data set instead of executing, see the discussion regarding @JCL (BEGIN) in Creating Multiple RESTORE Jobs.

After this is complete, executing FMSEGEN will result in output similar to the following:

```

SDSF OUTPUT DISPLAY ISPDLMIA JOB00531  DSID  111 LINE 4      COLUMNS 02- 81
COMMAND INPUT ==>                                SCROLL ==> CSR

DATA SET NAME / NEWNAME                                SOURCE DEVICE  ARCHIVED   TARGET
VOLUME  TYPE  TIME  DATE  VOLUME

***** THIS IS A SIMULATION RUN

CSLVL1.ISPF.TLIB                                SMS003 3300    1806 1998216S MS0033
DISK.BASE90.ANZCASE.PS                        SMS002 3300    1804 1998216S MS0023
DISK.PI061825.TST                            SMS003 3300    1805 1998216S MS0033
DISK900.WK654321.PDSE.CMDLIB1                SMS003 3300    1805 1998194S MS0033
DMS0S.GLIH.BACKMEUP.TST3                     SMS002 3300    1804 1998216S MS0023
ISPJMT1.AEG.BALT.CONVJCL                      SMS001 3300    1803 1998216S MS0013

SDSF OUTPUT DISPLAY ISPDLMIA JOB00533  DSID  110 LINE  CHARS * JOB * FOUND
COMMAND INPUT ==>                                SCROLL ==> CSR
//J208259 JOB (ACCT,XXXXXX,XXX,00), 'ACCT NAME',CLASS=C,MSGCLASS=X,
// NOTIFY=USERID1,MSGLEVEL=(1,1)
// J208259 EXEC RESTORE
//SYSTIN DD *
RESTORE DSN=QDMSPROD.NODMS.CBT.OBJ
RESTORE DSN=LADIUP5.JEG.TEST.SMS2
RESTORE DSN=QDMSPROD.NODMS.CBT.LKEDCNL
RESTORE DSN=ISPJMT1.AEG.CONV.JCLLIB

```

Selected data sets from an FMS job

These are part of the same data sets selected above, the difference being they all belong to ARCHVOL 208259.

Chapter 5: Reports

This section contains the following topics:

[About Reports](#) (see page 179)
[General Information](#) (see page 181)
[REPORT Commands and Parameters](#) (see page 183)
[REPORT Field Legend](#) (see page 192)
[Using the ISAM Reports](#) (see page 194)
[CLDATA Description](#) (see page 198)
[CIA Report Description](#) (see page 203)
[SMF Report](#) (see page 230)

About Reports

The objective of the CA Disk reporting facilities is to provide a rich, comprehensive palette of commands for efficient, common reporting on mixtures of non-VSAM data sets and VSAM clusters.

Within a single execution you can report on non-VSAM data sets, individual VSAM components, ICF catalogs, and VSAM clusters.

Note: 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*.

You can also define and dynamically change your own reports in a matter of seconds using the ISPF panels. You can then execute them online and review the output, or submit them as batch requests. See the section Overview of ISPF Online Reporting Facility in the chapter "ISPF", as well as the ISPF help text for more information regarding the online reports.

You can do this based on: scans of VTOCs disk volumes, scans of ICF or VSAM (but not OS CVOL) catalogs, or the results of a catalog superlocate.

You also have a rich set of criteria to select data sets for and exclude data sets from reporting as desired, and a rich set of reports to let you control exactly what you want to see.

Command	PROC	Parameters
EXCLUDE	DMS	
FIND	DMS	
PASSWORD	DMS	

Command	PROC	Parameters
REPORT	DMS	CLDATA, CIARPT, DSINDX, DSUTIL, ISAMDS, MAPPER, MEMBER, MVDICT, PDSTAT, POCCHH, PODUMP, POTTRD, POXREF, PO1REF, PO2REF, SVDICT
VREPORT	DMS	ALLOCS, ATTRBS, DISTRs, FREESP, LAYOUT
SCAN	DMS	
SELECT	DMS	
SELECT	SMFRPT	SMFRPT
SET	DMS	

General Information

Although managing data storage requires many different tasks to be performed, the ability to generate meaningful reports probably remains as the most common of them. Become familiar with the wide range of reporting facilities that CA Disk provides.

First there is the set of *standard reports*, which display data fields proven through the years to be highly relevant. These are *fixed format reports*, that is, reports in which the contents and format of the output have been predefined. They display data obtained from volume VTOC entries and catalog entries, and are commonly produced via batch execution. When combined with the select and exclude capabilities of the command language (DSCL), you have a high degree of flexibility in obtaining just the information you need.

A common complaint in trying to manage VSAM data sets is that even the report information is *completely different*. To help you in this regard, three of the *standard reports* MVDICT, SVDICT and MAPPER produced for non-VSAM data sets from VTOC information are also produced for VSAM clusters from information in the catalog. The catalog-based reports are formatted identically to the VTOC-based reports; however, the results can differ for multivolume data sets. VTOC-based reports will show only the volumes specified in the selection criteria, whereas the catalog-based reports will show all volumes the data set resides on even if the primary volume is the only volume selected. Additional reports CLDATA and CIARPT are also provided that analyze and display information, which is indeed unique to VSAM.

Five of the *standard reports*, ALLOCS, ATTRBS, DISTR, FREESP, and LAYOUT, are considered volume-level reports. These reports do not perform data set level processing. By eliminating this overhead, performance for these reports is greatly improved.

One more category of reporting is available through the ISPF support. This *online reporting* facility gives you the ability to actually define your own report(s). All of the possible reporting fields are displayed for you on the ISPF panel(s), from which you choose the ones that are needed. You also get to specify the *print position* for each field, sub and grand totals desired, and multiple sorting fields in ascending or descending sequence. When you finish defining the report, you can execute it interactively (online), or submit a batch job to produce it. Full details regarding these reports are contained in the Overview of ISPF Online Reporting Facility and the ISPF help text for the function itself.

The material presented in the remainder of this section applies to the standard reports obtained via batch, and to those parameters used by the ISPF online reports, which can also be executed in batch. (Use of the ISPF-related parameters requires that you have previously defined the needed data by making use of the online reporting option.)

Requests for report data can be made in two ways: by naming specific data sets (explicit requests) or by selecting certain data sets while scanning the VTOCs or catalogs (implicit requests). The implicit reporting function makes use of the DSCL commands to scan volumes and select or exclude the appropriate data sets. Explicit requests use the FIND command, which is also presented in this section.

Multiple reports can be generated during a single scan of a set of volumes. Records for each type of desired report are generated during the scan phase, are sorted into the appropriate sequence, and then printed by the report writer component of CA Disk.

Performance notes:

- If you are not interested in VSAM cluster information and are not running any concurrent BACKUP, BACKUPCC, ARCHIVE, DELETE, RELEASE, or BILLING commands, you can dramatically improve execution time by using the COMPONENTS parameter on the REPORT command. This is due to bypassing VVDS and catalog overhead for simple REPORT COMPONENTS processing.
- As described in the Data Storage Command Language section, only one scan of the VTOCs is made, regardless of the number of DSCL commands entered.

REPORT Condition Codes

Condition codes issued while executing the REPORT or VREPORT commands are described in the section Condition Codes in the chapter "Basic System – DMS PROC."

Restrictions

Volumes to be scanned and reported on must be online.

REPORT Commands and Parameters

The following JCL statements and DSCL commands are used to produce all of the CA Disk reports. For details on DSCL, see the section Data Storage Command Language in the chapter "Basic System – DMS PROC."

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SCAN REALVOLS  
SCAN CATALOGS=  
FIND DSN=, CATALOG=, PASSWORD=  
SELECT VOL=, DSN=, TABLE=, CRITERIA=, VTOCS, CATALOGS=  
EXCLUDE VOL=, DSN=, TABLE=, CRITERIA=, VTOCS, CATALOGS=  
REPORT CLDATA, CLDOPT=, CIARPT, CIAOPT=, DSINDX, DSUTIL, ISAMDS, MAPPER,  
        MEMBER, MVDICT, PDSTAT, POCCHH, PODUMP, POTTTRD, POXREF, PO1REF,  
        PO2REF, SVDICT, CMDGENDSN=, COMPONENTS, EXCLUDE, FLAGDAYS=,  
        MEMBERS=, MODULES=, SPFDSNAME=, SPFRPTS=, RPT=, SELECT=  
VREPORT ALLOCS, ATTRBS, DISTR, FREESP, LAYOUT
```

REPORT Command Names

The names of the reports available to the REPORT command shown in the previous figure are described as follows. The additional parameters that can be included on the REPORT command are described in Report Parameters.

CLDATA — Cluster Data Analysis Report

Generate a tailored report (see CLDATA) displaying the information associated with each cluster. Several different groups of items can be selected for display. A detailed description of the contents of each group selected follows later in this section.

IBM does not reliably maintain the information in the catalog entry describing an ICF catalog as a cluster itself. Therefore, the usage information report lines for an ICF catalog cluster itself are also unreliable.

CLDOPT= is optional parameter can be used to override the default group selection list used by the Cluster Data Report (CLDATA). All other report types specified on the REPORT command will ignore it. Its value is a string of code letters indicating the detail groups to be shown. The code letters follow immediately after the CLDOPT= parameter and are not separated by commas. The order in which the code letters are supplied on the command determines the order of the detail groups on the report. If this parameter is omitted when a CLD report is requested, sysparm RPTCLSEL will be used to determine the default groups to be printed.

Following is a list of the valid codes. If an invalid code is supplied, an error message will be issued and execution will be terminated.

D

Data component detail.

I

Index component detail.

E

Extent data.

S

Space allocation values.

U

Use statistics (default).

P

Protection levels.

A

Attributes (non-defaults in use).

Z

Attributes (defaults in use).

R

Recovery volumes.

C

Catalog name where cluster is defined.

B

Print a blank line.

X

Print data set's associations.

CIARPT — Cluster Internal Analysis Report

Generate an in-depth analysis (see CIARPT) of the physical characteristics of data in VSAM clusters. It is composed of three sections:

- Catalog information
- Detailed graphic analysis
- Summary statistics

A detailed description of the contents of each follows later in this section.

CIA reports are not created for LDS clusters, DB2 database clusters, compressed extended KSDS clusters, KSDS extended addressability in multiple extents or index levels, or ICF catalogs as data sets themselves. IBM does not maintain all fields necessary to analyze them accurately for this report.

Generating a CIARPT requires CONTROL access to VSAM clusters, not READ access. CA Disk must read the clusters on a Control Interval level.

CIAOPT=(CATLG,DETAIL,DUMP)

With this optional parameter you can specify options for printing the Cluster Internal Analysis report (assuming you specified the CIA report). The CATLG keyword instructs CA Disk to print the catalog information for each data set. The DETAIL keyword produces a detailed analysis graph of each used control area in the cluster. This option should be used with prudence because it can produce a large amount of output (each control interval in the data set will occupy a single print line). The DUMP keyword causes a CI Dump to be produced. The order in which you specify these keywords is insignificant.

If this parameter is NOT specified, default values are retrieved from sysparms VSCIACAT and VSCIADET. However, if the parameter is specified, it overrides the sysparm defaults.

DSINDX— Data Set Index Report

Generate a three-column listing (see DSINDX) of data set names and the volume on which each resides.

Note: Total allocated tracks in the three volume summary reports (ALLOCS, ATTRBS, and DISTRs) include the VTOC itself and the volume label track. The total allocated tracks for the data set level reports (such as the MVDICT or SVDICT reports) do not include the VTOC or label tracks, but are the sum of the data sets listed on the report.

DSUTIL— Data Set Utilization Report

Generate a listing (see DSUTIL) that includes the special fields as maintained by the CA Disk SVC regarding the usage of each data set. The special fields are:

- Date of last use
- Date of last modification
- Number of times the data set has been used

CA Disk assumes a default location in the format-1 DSCB for each of the previous fields. If some of these fields are not being maintained, or if they are at offsets different from the defaults, the following six system parameters must be specified to indicate their presence and location. Enter a displacement of zeros for any field not being maintained.

DSCBLUSD

Default Displacement: 075 (the IBM-defined location)

DSCBLMOD

Default Displacement: 070

DSCBOPCD

Default Displacement: 073

By default, the data sets are listed in sequence by date of last use, and the catalog status is displayed. Three system parameters permit slight variations in these options and are described briefly here. For more information, see their system parameter descriptions starting with DSCBLMODnnnk in the *Systems Guide*.

DSUTILSQ

Sequence by data set name.

DSUTILCF

Sequence by last use within index.

Note: By default, VSAM data sets are not reported on, because these fields are not maintained by the CA Disk SVC for VSAM. To include VSAM data sets, the COMPONENTS parameter must be added. This will report on the VSAM FMT1 DSCB information in the VTOC and not the catalog information for the cluster.

ISAMDS— ISAM Data Sets Report

Generate a report (see ISAMDS) of all ISAM data sets on the scanned volumes in alphabetical order. All information available is reported, which can include allocation and DCB information, format-2 DSCB fields, and detailed calculations on space used in each of the index, prime, overflow and cylinder overflow areas.

MAPPER—Volume Map Report

Generate a multiple-column map report (see MAPPER) for a volume with data set name or other appropriate identifier for each extent type. The CCHH value for each extent is displayed, along with the relative track number, length in tracks and, where appropriate, the number of the extent.

MEMBER— Partitioned Data Set Member Directory Report

Generate a multiple-column listing (see MEMBER) of members of the partitioned data set in alphabetic sequence. By use of the prefix notation, a listing of subsets of a PDS can be obtained.

MVDICT— Multiple Volume Data Set Dictionary Report

Generate a report (see MVDICT) of data set attributes listed in alphabetic sequence without regard to volume. Allocation data is included for each data set. Multiple volume data sets will appear on the report with one line for each volume they reside on, but note that for the count of 'TOTAL DATA SETS' – they are counted as one.

By default, the MVD and SVD reports display each cluster's Control Interval size of each VSAM cluster in the BLKSZ field. If you prefer that the MVD and SVD reports display each cluster's true block size for each cluster, specify the system parameter VSUSEBLK with a value of Y.

Note: The REPORT command is used to process VSAM components that are selected with the parameters SCAN REALVOLS and SELECT COMPONENTS. When you specify the COMPONENTS parameter in this REPORT command, catalog information is not available. As a result, fields such as RECFM, BLKSIZE and LRECL display Format 1 DSCB information.

PDSTAT— Partitioned Data Set Status Report

Generate a report (see PDSTAT) of one line per data set, which includes allocation and utilization information about member space, directory blocks and directory entries.

This report uses information from the directory and calculations from that information to determine directory entries allocated, used, and average length. The average length (for example, LEN) of the directory entry is divided into the directory blocks unused (and partially used) to determine the number of available directory entries (for example, IDLE) in the PDSTAT report field. The Directory Entries IDLE, therefore, is an estimate based on previous directory entries added.

POCCHH— Partitioned Data Set Directory (CCHHR Sequence)

Generate a formatted dump (see POCCHH) of the directory entries for partitioned data set members. The member entries are listed in their beginning CCHHR sequence. (Useful for matching system I/O error messages to specific PDS member names, or for use with performance analysis being done on a PDS.)

PODUMP— Partitioned Data Set Directory (Member Sequence)

Generate a formatted dump (see PODUMP) of the directory entries for partitioned data set members. The member entries are listed in member name sequence.

POTTRD— Partitioned Data Set Directory (TTR Sequence)

Generate a formatted dump (see POTTRD) of the directory entries for partitioned data set members. The member entries are listed in their beginning TTR sequence. (Useful for performance analysis of a PDS, or for matching known TTR values back to the actual member/alias names.)

POXREF— DSN Cross-Reference for PDS Member Names

Generate a list in member name sequence (see POXREF) of all the member names found in the partitioned data sets that were processed. Listed with each member name are the one or more data sets that contain a member by that name.

PO1REF— DSN Cross-Reference for Member in Only One PDS

Similar to the POXREF report except that the member is listed only when it is found in just one data set. Its suggested use is to find discrepancies between two libraries that should have a perfect member-for- member correspondence, such as a source and its matching load library. A sample can be found in PO1REF.

PO2REF— DSN Cross-Reference for Member in Two or More PDS Data Sets

Similar to the POXREF report except that the member is listed only when it is found in two or more data sets. Its suggested use is to identify those members that appear in more than one library, but are intended to be unique to a single library. Examples would be for checking system proclibs (JCL procedure libraries), or for monitoring test libraries versus their production counterparts. A sample can be found in PO2REF.

SVDICT— Single Volume Data Set Dictionary Report

Generate a report that is identical to the MVDICT report in all ways, except that the data sets are displayed alphabetically by device type, and then by volume name. A sample can be found in SVDICT. However, unlike the MVDICT, multiple volume data sets will be counted once per volume since the SVDICT report is in device type and volume sequence.

By default, the MVD and SVD reports display the Control Interval size of each VSAM cluster in the BLKSZ field. If you prefer that the MVD and SVD reports display the true block size for each cluster, specify the system parameter VSUSEBLK with a value of Y.

Note: The REPORT command is used to process VSAM components that are selected with the parameters SCAN REALVOLS and SELECT COMPONENTS. When you specify the COMPONENTS parameter in this REPORT command, catalog information is not available. As a result, fields such as RECFM, BLKSIZE and LRECL display Format 1 DSCB information.

Report Parameters

Several additional parameters can be included on any REPORT command. These parameters are described in the following sections.

CMDGENDSN=

This optional parameter can be specified to cause CA Disk to save the data set names selected for reporting. This is done by writing the name of each selected data set (and the volume where it resides) to the data set specified on this parameter. The data set must be defined as sequential and have a logical record length of at least 50. Any data set that meets all selection criteria and is displayed on the report will also have its name written to this data set. The records in this data set can then be used as input to generate other commands, whether they are for CA Disk or some other utility.

CA Disk, using the ISPF panels, can do the generation of new commands from the list of saved data set names automatically. The online reporting feature provides this capability.

Note: This facility is available only to MVS installations because the dynamic allocation feature of SVC99 is used to allocate the data set.

COMPONENTS

By default, CA Disk reports on non-VSAM data sets and VSAM clusters. Specify this parameter to have CA Disk report on non-VSAM data sets and VSAM components.

If you are not running any BACKUP, BACKUPCC, ARCHIVE, DELETE, RELEASE, or BILLING commands in the same SCAN/FIND group, you can dramatically improve execution time by specifying this parameter on the REPORT command. This is due to bypassing VVDS and catalog overhead for simple REPORT COMPONENTS processing.

EXCLUDE

This optional parameter applies only to the MEMBERS parameter. When omitted, a report involving PDS members will include only those corresponding to entries in a member list. Specification of this parameter will exclude listed members.

FLAGDAYS=

Specifying a one- to five-digit value for this parameter causes an asterisk (*) to appear in the flagdays field on the data set utilization (DSU) report for those data sets that have not been used within the supplied number of days prior to the current date. If the last used date is zero, a "\$" will appear instead, and the creation date will be displayed in the used date field.

MEMBERS=

A list of one or more member names and prefixes can be included when the report to be produced involves the output of PDS members. (A maximum of 20 names is supported.)

MODULES=

A list of one or more modules to be passed control when a non-format-0 DSCB has been retrieved. The parameters passed are the format-4 DSCB for the volume and the non-format-0 DSCB. (A maximum of 20 entries is supported.)

SPFDSNAME=

If your ISPF report definitions are not stored in the standard definition library specified in sysparm SPFDSNAM, you must supply the name of the data set with this parameter. All of the report(s) specified in the SPFRPTS= parameter must reside in the same definition library.

SPFRPTS=

Enter a list of up to 30 report names that were defined using the online CA Disk reporting facility (these are stored in a different format than the standard CA Disk reports). For detailed information on the ISPF reports, see the section Overview of ISPF Online Reporting Facility in the chapter "ISPF."

VREPORT Command Names

The names of the reports available to the VREPORT command shown in REPORT Commands and Parameters are as follows:

ALLOCS — Volume Allocation Summary Report

Generate one report line per volume (see ALLOCS) that provides the total allocated and free space, and the size of the three largest extents. The total number of DSCBs, by type, is also listed. See "DISTRs — Volume Distribution Summary Report" for additional information.

ATTRBS — Volume Attribute Summary Report

Generate one report line per volume (see ATTRBS) that provides the total number of data sets by data set organization. Totals are also given for device tracks and VTOC DSCBs. See "DISTRs — Volume Distribution Summary Report" for additional information.

DISTRs — Volume Distribution Summary Report

Generate one report line per volume (see DISTRs) that provides a distribution of data sets based on the number of tracks allocated. The distribution is according to a fixed set of ranges for the size of the data sets. Totals for allocated and free space extents are also included.

Note: Total allocated tracks in the three volume summary reports (ALLOCS, ATTRBS, and DISTRs) include the VTOC itself and the volume label track. The total allocated tracks for the data set level reports (such as the MVDICT or SVDICT reports) do not include the VTOC or label tracks, but are the sum of the data sets listed on the report.

FREESP — Free Space Detail Report

Generate a report (see FREESP) of all free space extents listed in ascending track order for a volume. Starting track and length are listed for each extent.

LAYOUT — Volume Layout Report

Generate a compact list (see LAYOUT) of all extents on a volume. The extents are shown in track sequence, by extent type and length in tracks.

REPORT Field Legend

Fields and indicators used in reports are explained as follows. Several self-explanatory fields have not been included. The CIARPT, CLDATA and ISAMDS report fields are listed separately in the section immediately following.

Used in DSU report to indicate that a data set has not been used within the period specified on the report command.

\$

Used in DSU report to indicate that a data set's last used date is zero, so the creation date has been displayed as the last used date.

ALLOC

The number of tracks allocated to data sets.

AVAIL

The number of free tracks available for allocation.

BLKSZ

For non-VSAM, the DCB blocksize of the data set For VSAM, the CISIZE of the data set.

CAT

Catalog status. Possible values are:

Y

Cataloged correctly

N

Not cataloged

-

Cataloged, but to a different volume

CREDIT

The Julian date the data set was created.

CREATING JOB

The first job to open the data set. Note that IEFBR14 allocation jobs will not appear since they merely allocate space and do not open and load any data into the data set.

PRO

Data set protection status. Possible values are:

F

Non-VSAM, RACF-indicated.

R

For read/write password protection.

W

For write password protection.

C

VSAM, control-interval password protected.

M

VSAM, master password required for access.

R

VSAM, read-only password protected.

U

VSAM, update access password protected.

EXPDT

The Julian date on which the data set expires.

FREE

The number of free (format-0) DSCBs in the VTOC.

IDLE

The number of unused tracks in a data set.

LRECL

Logical record size. For VSAM it is the logical record size from the data component.

MODDT

The Julian date the data set was last modified.

PRIM

Primary space allocation value, expressed in the type of units in which the allocation was made. Since the actual primary request value is not recorded in the VTOC, the size of the first extent is displayed. The first extent size is calculated from information in the format-1 DSCB. Unless the data set is very large, or the free space on a volume is badly fragmented, the first extent will accurately reflect the primary space. If you desire to alter this field, see the sysparm description for ALLOCPRI in the *Systems Guide*.

SEC

Secondary space allocation value, expressed in the type of units in which the allocation was made.

TYPE

The type of allocation unit: CYL, TRK, or BLK the Julian date the data set was last used (opened).

USEDT

The Julian date the data set was last used (opened).

Using the ISAM Reports

The design of the CA Disk ISAM reports allows for maximum information about the data set to be printed when scanning any single volume. Only the fields that can be calculated for the given volume are printed. Consequently, for ISAM data sets spread across two or three volumes, different fields will be printed for each volume.

The report is printed in data set name sequence. If several volumes are scanned in a single report, portions of the data set on different volumes will be grouped together, but the information for each volume is listed separately.

When it is determined that part of the data set resides on another volume, this information is printed in the first line of the report, specifying the volume if it can be determined.

For example:

SAMPLE.DATA SET PACK01 has O, PACK99 has I and/or P

where O, I, and P see overflow, index and prime areas.

The percentage of space that is idle is printed at the right side of the REPORT for easier reference. Since ISAM allocation is limited to multiples of whole cylinders, controlling space allocation is somewhat harder. The most important item to check is probably the percentage of prime that is idle. If this is high, then a check for the total tracks of prime allocated should be done. For a one-cylinder data set with only one track used (94 percent idle on a 3330), nothing can be done to reduce the size.

ISAM REPORT Legend

I

Index

P

Prime

O

Overflow (independent)

COV

Cylinder Overflow

VOLSQ

Volume Sequence Number

LRECL

Logical Record Length

BLKSZ

Blocksize

RECFM

Record Format

OPTCD

Option Codes

KEYL

Key Length

RKP

Relative Key Position

LOADED

The data set has been loaded/initialized

RES-I

Bytes of memory to make highest index resident

NTM

Number of tracks in master index before creating another level of the index

LVLS

Levels of index

1

Cylinder index.

2

First-level master.

3

Second-level master.

4

Third-level master.

RECS

Number of Records

DELS

Number of records marked for deletion

FCOV

Number of cylinder overflow areas which are full

T/COV

Number of tracks per cylinder overflow area

R/W

Number of reads/writes to independent overflow

ALLOC TRKS

Number of tracks available for use

USED

Number of tracks which have been used

IDLE

(Allocated tracks) - (used tracks)

***USED**

(Full CYL overflow areas) x (tracks/CYL OV area)

***IDLE**

(CYL OV areas not full) x (tracks/CYL OV area)

NCOV

Total number of cylinder overflow areas

EXTS

Number of extents in the data set

TYPE

Type of original space request:

ABS

Absolute Trk

ABSC

Absolute Trk Contig

CYL

Cylinder

CYLC

Cylinder Contig

PRIM

The original primary space request, expressed in the same units as the type; for example, units are in:

TRACKS—if TYPE=ABS or ABSC

CYLINDERS—if TYPE=CYL or CYLC

PCT I IDLE

Percent of index space that is idle

PCT P IDLE

Percent of prime space that is idle

PCT O IDLE

Percent of overflow space that is idle

PCT * IDLE

Percent of cylinder overflow tracks not full

CLDATA Description

The Cluster Data Report (see CLDATA) contains information about all clusters selected for reporting. The report is organized as follows:

1. The report is subdivided into groups and where possible a group will be shown on one print line. A user can optionally select to suppress any or all detail groups.
2. Most groups are further subdivided into items that are shown in a *keyword* fashion; that is, the item name followed by an equal sign (=) and then the value (for example, RECSZ=100).
3. VSAM definitions incorporate many default values. In order to produce a concise report, CA Disk can suppress a group item if the default value is in effect. For example, the default attribute for a cluster is NOSPAN, so if NOSPAN is in effect, NOSPAN will not be displayed in the attribute group. However, if SPAN is in effect, SPAN will be displayed, as it is a non-default value.
4. If all items in a group are suppressed, the printing of the entire group will also be suppressed. For example, if all attributes in effect are the default values, the attribute line will be suppressed.
5. Where *non-keyword* values are to be printed, CA Disk lists the values in alphabetical order. This simplifies locating a particular group item.
6. An attempt has been made to shorten the conventional report line length in order to improve the usability of CA Disk in a terminal environment.

Cluster Name Line

The cluster name line shows the name, the cluster type, the primary volume serial, the total tracks allocated, and the percentage of space used. This line is always displayed first and cannot be suppressed. If the cluster resides on multiple volumes, the primary volume for the data component will be displayed under the VOLUME heading, followed by a "+" sign (the remaining volumes on which the cluster resides can be found by looking at the extent detail lines). In addition, the total number of control area and control interval splits (data + index) will be shown if splits have occurred. If more than the minimum number of extents are allocated, the number of extents are also shown; that is, more than one for an ESDS or RRDS and more than two for a KSDS.

Note: KSDS clusters defined using the key-range attribute, all VSAM catalogs have at least three extents, and hence the number of extents will be shown.

A cluster data report in which all other detail lines have been suppressed provides a concise cluster summary report.

Data Component Detail Group (D)

The data component detail group will always provide the average record size (RECSZ=) associated with the cluster. If the maximum record size differs from the average, the maximum record size (MAX=) will also be shown. The control interval size (CISZ=) for the data component (whether specified by the user or chosen by VSAM) will always be provided.

If any control area or control interval splits have occurred in the data portion, the number of splits will be displayed here. If neither CA nor CI splits have occurred, this item will be suppressed.

The data set name associated with the data component will be printed on the subsequent line if sysparm VSCOMPNM is specified with a value of Y. When overriding this sysparm in job streams that use other than the DMS proc, specify PRINT.SYSPARMS.

If any freespace is defined for a KSDS, the freespace percentages will also be displayed (FREESPACE%-CI= and FREESPACE%-CA=).

Index Component Detail Group (I)

The index component detail line is printed only for KSDS type clusters. The length of the key field (KEYL=) is always provided. The relative position of the key field within the data record is provided if its value is greater than zero (RKP=).

If the index contains more than two levels, the number of index levels (LEVELS=) is provided. The control interval size (CISZ=) for the index component (whether specified by the user or chosen by VSAM) will always be provided.

If any control area or control interval splits have occurred in the index portion, the number of splits will be displayed here. If neither CA nor CI splits have occurred, this item will be suppressed.

The data set name associated with the index component will be printed on the subsequent line if sysparm VSCOMPNM is specified with a value of Y. When overriding this sysparm in the job stream, specify PRINT.SYSPARMS dd.

Extent Detail Line (R)

One extent detail line is printed for each unique extent that is allocated to the cluster. The extent detail line shows the volume serial that contains the extent, the first and last absolute track address of the extent, the total number of tracks allocated to the extent, and the word *INDEX* if the extent contains only the index of a KSDS. Since embedded indexes share some extents with the data portion of the cluster, these common extents will not be duplicated for the index component. This should be kept in mind when the user is comparing an IDCAMS LISTCAT output to the CA Disk CLDATA report.

The extent limits are shown in cylinder and head format (cccc—hh) and are displayed in hexadecimal. The maximum number of extents that can be displayed is 32.

Space Allocation Detail Line (S)

The allocation detail line shows the amount of space requested at definition time. The data and index component values can have been specified by the user or chosen by VSAM based on the SPACE parameter provided for the cluster or model. The format is similar to the OS/VS SPACE parameter used on a dd JCL statement. The first value indicates the type of allocation (CYL or TRK). Second is the primary or initial quantity requested of the indicated type. Third is the secondary quantity for subsequently extending the amount of allocated space. If no secondary quantity was requested, this value is suppressed.

If the cluster type is keyed, the index allocation is provided in the same format.

Note: When a cluster is defined with a space allocation of records, Access Method Services converts this amount to an equivalent amount of storage in either tracks or cylinders. Therefore, CA Disk will always report allocation amounts in either tracks (TRK) or cylinders (CYL), and never in records.

Use Statistical Detail Group (U)

The statistical detail group shows the number of records subjected to the indicated action. If any value is zero, that item will be suppressed. If all values are zero, the entire statistical detail line will be suppressed. Following is the list of actions and associated definitions:

ADD

Number of records added since completion of the initial load.

DEL

Number of records deleted.

EXCP

Count the number of EXCPs issued against this cluster.

GET

Number of times any record was retrieved.

TOT

Total number of user data records currently in the cluster.

UPD

Number of records updated.

Note: These statistics pertain to the data component only. LAST MOD DATE and TIME for each cluster is also shown.

Password Detail Group (P)

The password detail group shows either the levels of protection or the actual passwords required for access at each level. If the proper password has been supplied (either the catalog's master password, the CA Disk master password or the cluster's password), the actual passwords are provided at the following levels:

M

Password for master level access.

C

Password for control interval access.

U

Password for update or write access.

R

Password for read only access.

If the proper password has not been supplied, the level of access will replace each password as follows:

M

Master.

C

Control.

U

Update.

R

Read.

If access is not restricted at a particular level, that level will be suppressed. If access is not restricted at any level, the entire password detail line will be suppressed. If the cluster is RACF-protected, a literal of *RACF* will be displayed on the password report line.

If the number of attempts allowed to supply the entity's password is other than zero, the literal *ATTEMPTS=* will also appear on the password detail line. Beside the literal will be the actual number of attempts that are allowed.

Attribute Detail Group (A, Z)

The non-default attributes that are in effect for the cluster can be displayed by selecting the (A) group.

The default attributes will be displayed if the (Z) group is specified. Items within each group are listed alphabetically.

The following table lists the non-default and default values that will be displayed:

Non-Defaults (A)	Defaults (Z)	Description
ERASE	NO	Data secure erase
IMBED	NO	Store seq set w/data
ORDER	NO	Use vols in order
REPL	NO	Repeat index on trk
SHR(-,-)	SHR(1,3)	Share options
SPAN	NO	Allow RECSZ > CISZ
SPEED	RECOVERY	No preformat on load
UNIQUE	SUBALLOCATE	Cluster = space

Non-Defaults (A)	Defaults (Z)	Description
UNIQUEKEY	NON	AIX with unique keys
WCK	NO	Verify data after write

Catalog Name Line (C)

The Catalog Name report line displays the catalog name in which the cluster is defined.

Blank Line (B)

This SELECT type is used to generate blank line(s) in the Cluster Data Report. It is used primarily to enhance readability. You can print as many blank lines as desired. Double-spacing can be achieved by supplying two *B* SELECT types consecutively.

Association Lines (X)

Each association for the entity being reported on will be displayed on a separate line. Immediately following the association name on the print line is a literal indicating the type of association (for example, cluster, path, data, and so on).

CIA Report Description

The Cluster Internal Analysis report contains information about the internal structure of data in VSAM clusters. The report has a special format for each of the three different types of VSAM clusters:

- Key-sequenced data sets (see CIARPT)
- Entry-sequenced data sets
- Relative record data sets

These reports were designed to allow as much similarity as possible between them, while still allowing for the differences in data structures of each cluster type. The reports can be generated as a function of the REPORT command, or can be produced as a by-product of the ARCHIVE or BACKUP command. It should be noted that in order to produce these reports, CA Disk must read the clusters with control interval access (which is the method used by default with CA Disk).

Each of the reports has three distinct portions to them:

- Information obtained from the catalog (optional)
- A graphic display of each control interval in every control area that contains data (also optional)
- A statistics section summarizes the key information about the cluster.

The first two sections can be dynamically selected by specifying the CIAOPT= parameter of the REPORT, BACKUP and ARCHIVE commands, or can be controlled by specifying sysparms VSCIACAT and VSCIADET (for detail information, see their sysparm descriptions beginning on VSCIACATy in the *Systems Guide*).

CIA Report KSDS Catalog Information Section

The following terms and acronyms are used in the CIA Report KSDS Catalog:

DEV TYPE

This field is the device class of the volume(s) on which the component resides. All of the volumes for a particular component must be of the same device type, but the device type can be different for the data and index components.

EXTENTS

EXTENTS represent the total number of extents in which the component resides.

CI SIZE

The control interval (CI) is the size for the component. This is the unit of information transmitted to or from auxiliary storage by VSAM.

PHY RECSZ

This is the physical record size used to store information in the cluster. There is a whole number of physical records that comprise one control interval.

REC / TRK

This is the number of physical records stored on each track. This will give an indication of the effectiveness of the blocking factor on this particular DASD device.

HI USED RBA

The high used relative byte address (RBA) in this component. This address effectively marks where the first empty control area resides, or the end of the data set if all space has been used.

HI ALLOC RBA

The high allocated relative byte address (RBA) in this component. This address marks the end of the data set in its currently allocated extents.

TRKS / CA

TRKS/CA represents the number of tracks per control area. For larger clusters this value should be the same as the number of tracks per cylinder on this type of DASD. This would represent the ideal control area size of one cylinder.

CI / CA

The number of control intervals that are contained in one control area. This is an important number in key-sequenced data sets when determining the appropriate control interval size for the index. See Unreferenceable Data Control Intervals.

FREE %-CI

FREE%-CI represents the control interval freespace percentage with which the cluster was defined. This is the percentage of freespace to be reserved in each control interval when it is loaded.

FREE %-CA

This value represents the percentage of control intervals in every control area that is to be left empty when the cluster is loaded. It is specified when the cluster is initially defined.

AVG LRECL

This is the length of the average record (specified when the cluster was defined) that the user expects the data set to contain. When variable-length records occur in a data set, this value can or cannot be an accurate reflection of the actual data, but does the user supply an estimate by which VSAM estimates space requirements when loading the cluster. The actual average record length of the data set is shown in the AVG RECSZ field in the statistics summary of this report.

MAX LRECL

This is the maximum record length allowed in the cluster. If this field is the same value as AVG LRECL, the file contains fixed-length records; otherwise they are variable-length.

TOTAL RECORDS

The total number of records the catalog expects the data set to contain. This cannot reflect the true number of records. The catalog information is accurate only when data sets are properly closed after updating. If a system crash occurs during updating, the catalog statistics can be inaccurate (this value alone in no way affects the integrity of the cluster). Most database systems using VSAM do not perform normal close operations, so these data sets will rarely have the proper value in the catalog. For the true number of records in the data set, see the TOTAL RECORDS field in the statistical summary of this report. This number is obtained by actually counting the number of records that reside in the data.

CI SPLITS

CI SPLITS represents the total number of control interval splits that have occurred in the data set.

Index Component Catalog Information

Most of the catalog information displayed for the index component of the cluster is in the same format as the data fields, with the following additions:

SEQ SET RBA

This field represents the relative byte address of the first control interval in the sequence set of the index.

INX LVLS

This is the number of index levels that occur in the index. For direct access retrieval of records in the data component, this number represents the number of index control intervals that must be accessed to locate the proper control interval in the data component that contains the record.

HI LEVEL RBA

This is the relative byte address of the control interval that contains the highest level segment in the index. For direct access retrieval of records, this is the first index segment that is retrieved to begin the search for the specified key value.

KEY LTH

This is the length of the key in the data portion of the cluster, as specified by the user when the cluster is defined.

RKP

This is the relative key position. It represents the offset from the beginning of the data record to where the key of the record can be found. An RKP of zero indicates the key starts in the first character of the data record.

CIA Report Sequence Set Graphic Analysis

The sequence set analysis graph is produced only for key-sequenced clusters (since they are the only type that has an index). Its purpose is to give a graphic picture of the control intervals that comprise the sequence set portion of the index. Note that this does not include control intervals that are part of the index set, because these are not needed unless the data set is processed by keyed direct access.

This graph contains one print line for each control interval in the index's sequence set. The graph line can consist of four different print characters:

- An X represents header information overhead
- An F represents space used to hold free control interval pointers
- Spaces symbolize space, which is unused in the control interval
- An I represents space that is used to hold the compressed keys and pointers to the data control intervals.

A legend is printed under the final line of the graph to indicate the use of these characters. We will now take a closer look at each of these values.

Header Information

The header information is always 24 characters in length. This header contains control information about the index record itself. The contents of this information are not important to this discussion, but you should be aware that these 24 characters are not available to hold actual data. You can notice that on different clusters, however, a different number of X's can appear on each line of the graph. This is because of the scaling factor that was used on the graph, which is based on the control interval size of the component (you can see both of these values on the print line that precedes the graph). With a control interval size of 4096 (we will see these values as CI size from now on), the scaling factor is 1/64. This makes the 24 bytes insignificant for graphing purposes, and will therefore not show up. On the other hand, a CI size of 512 produces a graphing factor of 1/8, and therefore three X's will be displayed to represent the 24 characters.

Freespace Pointers

The freespace pointers (represented by F's) can or cannot exist in the control interval. These pointers are used to point to control intervals in the data component that are not being used in this control area. (Each sequence set CI *owns* one specific control area in the data component.) A field in the header information governs the length of each of these pointers. The column at the side of the graph labeled FREE PTRS is the number of free pointers found in the sequence set record. This is an actual count, and not the number of characters needed to hold the pointers. Therefore, this count represents the actual number of control intervals in the data component's control area that are not being used. We will discuss freespace pointers in more detail in a little while.

Freespace

The freespace in the sequence set (represented by spaces) is quite different from the freespace pointers. This is used to display the space that is unused in the sequence set record. If all of the CIs in the control area for the data are already being used, then this area is of no significance (remember, all of the CIs are being used if the freespace pointers value is zero). However, if there are unused CIs in the data's control area, this area can become much more important. We will discuss this more in Unreferenceable Data Control Intervals. For now, just remember that the value under the column FREE SPACE represents the number of empty positions in the sequence set record that can be used for other purposes.

Index Entries

The / in the graph symbolize space that is used to hold the pointers to actual user data in the data component. This consists of some characters for control information, the pointers to the data control intervals, and most importantly the key value (in compressed format) of the highest record in each control interval. This key is used by VSAM in keyed access to determine if a particular data control interval has the record it is looking for, without having to read each data control interval itself.

There are two column headings that relate directly to the index entries in the graph. The first one is labeled AVG ENTL, which stands for the average entry length of an index entry. This value is the average number of characters required to reference a single data control interval, which includes control information overhead and the compressed key of the data. This value is important when looking at the issue of unreferenceable control intervals, which we will look at in Unreferenceable Data Control Intervals.

The second column applicable to index entries is AVG KEYL. This value represents the average number of characters required to hold each compressed key. What this value tells you is how well VSAM is compressing the keys used to reference the data. This information can be used when deciding upon an explicit CI size for the index if you want to reorganize the cluster with CA Disk (using ARCHIVE and RESTORE). The new CI size can be explicitly stated on the RESTORE command.

Description of Sequence Set Graphic Headings

This section will explain each of the values printed next to the graphic display. Those columns that were described in preceding sections will only be summarized at this point.

CONTROLS CA AT RBA

This is the relative byte address of the start of the control area that this index CI controls. This number is used by the VSAM access method to determine where the data corresponding to the keys in this index starts. This value is provided for informational purposes only.

FREE PTRS

This is the number of free control interval pointers that exist in this particular index control interval. In other words, this number represents the number of data control intervals that are not being used in the corresponding data control area.

USED PTRS

This is the number of pointers being used in this index CI. This corresponds to the number of data control intervals being used in the control area. This value, added to the FREE PTRS value, is the same as the number of control intervals per control area in the data component.

FREE SPACE

This is the number of empty bytes or characters in the index control interval that can be used to hold pointers to the data control area. This value was discussed previously.

AVG ENTL

The average number of bytes required to hold both the control information and compressed data key for each control interval being referenced in this data control area.

AVG KEYL

The average length to which each data key compresses, excluding any system control information.

UNREF CI'S

The number of unreferenceable data control intervals in this control area. The following paragraph describes this value in detail. If this value is not zero, DASD space is being wasted. If this number is large, redefine the cluster with different control interval sizes.

CA SEQ

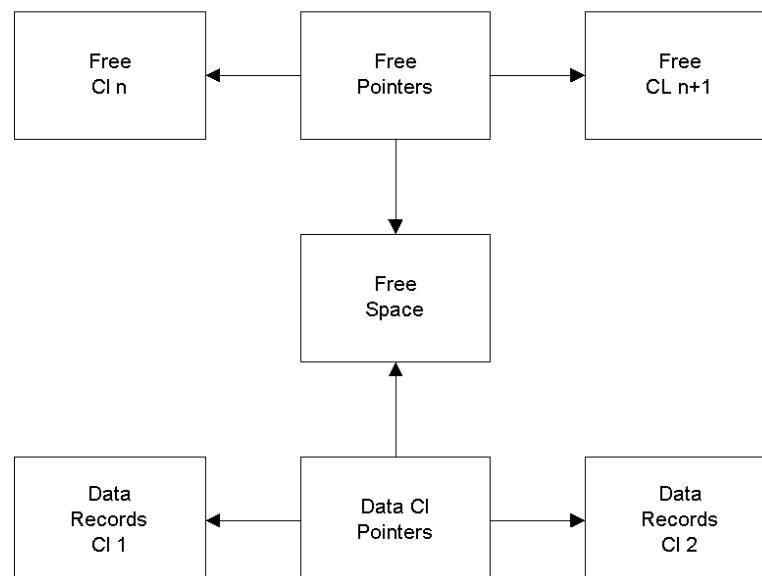
This value represents the relative order of the control area in key sequence. In a cluster that has had no control area splits; this value should always be incremented by one for each line in the graph.

CA DEV

This value represents the relative deviation between the current control area and the one that preceded it in key sequence. In an organized cluster, this value will be 1. Since this value is *normal*, it is suppressed on the report so you will see blanks instead. If the value is other than 1, the control area does not physically reside next to the control area beneath it in key sequence. What does this mean? Assuming a control area size of one cylinder (the optimal value for most clusters), the arm that reads the data on DASD has to move the number of cylinders shown under the REL DEV column to read the data in keyed sequential mode. If the number printed has a - after it, the seek would be backwards. It is important to understand that this number has no relevance when the cluster is accessed in direct key mode, since the index is searched to find where the record resides (such as in online databases). But if the file is accessed in keyed sequential mode—as CA Disk does to produce a backup copy of the data set— then the head movement can have a negative impact on the performance of the read operations.

Unreferenceable Data Control Intervals

To explain what causes data control intervals to become unreferenceable requires an explanation of how a sequence set control interval is structured. Remember that one sequence set control interval governs a single control area in the data component of the VSAM cluster. The following illustration is a sample Sequence Set Control Interval:



Before a data area contains any records, the sequence set record contains only free pointers to the control intervals in the data control area. This lets VSAM know that all of the control intervals are available and ready for use. When VSAM has a request to add a record, it checks to see if there is space available in any of the used control intervals. If it determines that a free control interval must be used to hold the new record, the sequence set is checked to see if there are any available. If there are, the key of the record is compressed and placed in the right side of the sequence set record, along with a pointer to the appropriate data control interval. At the same time, the free pointer is removed from the left side of the sequence set CI. This process repeats itself every time a new control interval is required.

As you can see, each time a free pointer is used, some space is reclaimed from the left side of the sequence set CI, while additional space is used on the right side. The free pointer will always be from one to three characters in length. However, the information that gets added on the right side is almost always larger, since it must contain pointers, compressed keys and control information. So the space in the center of the sequence set CI (labeled free space) slowly gets used up as each new control interval is used in the control area.

Normally this is not a problem, because catalog management tries to not let you define an index control interval size that is too small to reference all of these data control intervals. But there are occasions, however rare, when the keys of the data do not compress as well as expected. When this occurs, there is not enough free space remaining in the sequence set CI to hold the compressed key of the data record. Therefore, even though there are free control intervals in the data area, they cannot be used because the sequence set CI has run out of room to hold the necessary information. This space in the data component effectively becomes dead space.

Obviously this is not a good condition. There are several methods that can be used to rectify this condition, and the best choice depends on the circumstances in your environment. The first alternative is to reorganize your cluster and define it with a larger index control interval size. This will add more free space to the sequence set CIs. Another approach is to increase the data CI size, since this would mean that fewer data control intervals would have to be referenced by the same sized index. The last alternative is to redefine the cluster with different space attributes, such that the control area size would be decreased (see the IBM publication *Access Methods Services* for a discussion of how space allocation affects control area size). This last approach would be the least desirable from a performance point of view, unless the control area size of the cluster is already less than one cylinder.

You can have encountered a situation that is caused by this very issue during device conversions of VSAM data sets. For example, say you are moving a cluster from a 3350 to a 3380 and the control area size is one cylinder. You archive the cluster and attempt to do a restore, only to get an error message (IDC3009I-212) during the define that says *control interval size calculation unsolvable*. What occurred is that catalog management failed the define because it is trying to avoid this very situation. Since the control area size is one cylinder, there are now more data control intervals that have to be referenced by the same-sized index CI, and catalog management decided you would end up with unreferenceable data control intervals. So what was valid for the smaller device is not valid for the larger one. The solution to this problem is the same as the previous one outlined.

CIA Report Data Component Graphic Analysis

The data component analysis graph provides a picture of what each used control area in the data component looks like. Each control area is represented by a separate box graph. At the conclusion of each control area, summary totals are printed for the appropriate fields in the control area.

This graph contains one print line for each control interval in the control area. Most graph lines will consist of four different print characters:

- Ds represent space used to hold data records.
- Spaces symbolize space that is unused in the control interval.
- An * represents space considered unusable for data storage (dead space).
- An X represents control information overhead.

A legend is printed under the final line of the graph to indicate the use of these characters. In control areas that contain spanned records, you will also see the following symbols:

F

Represents the control interval containing the first segment of a spanned record.

I

Represents any intermediate segments.

L

Represents the last segment of a spanned record, followed by *'s to indicate the dead space in the last control interval.

The legend for spanned records will only appear after those control areas that actually contain at least one spanned record. You can also notice some control intervals with a literal of — *FREE* — inside them. These are control intervals that are marked as free in the sequence set CI and therefore contain no data.

We will now take a closer look at each of these values.

Data Records

The characters (D) are used to represent the space actually used to contain data records. The number of characters in all of the data records in this control interval can be found under the column BYTE TOTAL.

Freespace

This is space that is currently not being used in the control interval and is therefore available either to add new records or to expand existing ones in variable length files. This space is represented by spaces.

Dead Space

This is the amount of free space in a control interval that cannot be used to add other records, so CA Disk categorizes it as dead space. The character * represents dead space. In fixed-length record files this amount is a constant value and is calculated as follows:

$$\text{CISIZE} - (\text{RECSIZE} \times \text{RECORDS PER CI}) - 10$$

For variable-length files CA Disk makes an estimation of dead space based upon the records processed to that point. Any free space quantity that is less than the average record size +3 is considered dead space. The +3 is added in to assume the worst case of not more than two consecutive records having the same length records, which would cause three bytes of control information to be added to the control interval. It should be noted that this space is available to expand existing records. Also, even though the space is considered dead now, a control interval split later on would make the area usable again.

The unused space in the last segment of every spanned record is also marked as dead space. It is a VSAM restriction that any control interval that contains a portion of a spanned record cannot contain any other records, even if there is enough free space to hold additional records.

Control Information Overhead

The character X is used to depict the space used to hold control information for the control interval. This contains such information as the length of the data in the control interval and the length of any freespace. For fixed-length files the length of this area is always 10. For variable-length files it is 4 bytes, plus 3 bytes for each data record in the control interval (actually, VSAM does save some space when more than two consecutive records have the same length of data).

Description of Data Component Graphic Headings

The following table describes different data component graphic headings:

REC TOTAL

This is the total number of data records found in the control interval. When the CI contains a portion of a spanned record, the literal SPAN will appear in front of the record count. A value of zero is displayed for the first and any intermediate segments of the spanned record, and a value of one is displayed for the last segment.

BYTE TOTAL

This is the total number of characters found in all of the data records in this control interval.

FREE SPACE

This is the amount of unused space in the control interval. It can be used to add new records or to expand records in a file that can contain variable-length records.

DEAD SPACE

This is the number of bytes that CA Disk considers unusable in the control interval.

OVERHEAD

This is the number of bytes required to hold the system control information.

% USED

This is the percentage of the cluster currently being used. For the purposes of this figure, used space can either be space used to hold data records or space required by VSAM for control information.

% DEAD

This is the percentage of the control interval considered to be unusable. It is calculated by dividing the number of dead bytes by the control interval size.

REL SEQ

This value represents the relative order of each control interval in the control area by key sequence. In a control area that has all of its control intervals in key sequence, this value will always be incremented by one for each line in the graph.

REL DEV

This value represents the relative deviation between the current control interval and the one that preceded it in key sequence. In an organized cluster, this value will be 1. Since this value is *normal* it is suppressed on the report so you will see blanks instead. If the value is other than 1, the control interval does not physically reside next to the control interval beneath it in key sequence. This normally would have no impact on the performance of the cluster. The only exception to this would be when the control area size is less than one cylinder and the deviation would cause the DASD device to have to shuttle between two different cylinders to read the control intervals in keyed-sequence mode.

CIA Report KSDS Data Component

The summary information is the only portion of this report that is not optional. It contains all of the statistics applicable to the data set as a whole. We will now examine each of the fields on this report.

TOTAL RECORDS

The total number of data records in the data set. This can or cannot have the same value as the total records value in the catalog section of this report. Remember that the total records value printed in the catalog section is based on the value maintained in the catalog while the number in this section is based on the actual records in the cluster. This value is therefore more accurate.

TOTAL BYTES

This is the total number of bytes or characters of data that are used to hold the data records in the cluster.

SPACE %—USED

The percent used gives an indication of the space already consumed in the data component. It accounts for space used to hold data records, control information overhead, and any dead space encountered in the data set.

USED CI COUNT

This field gives the number of control intervals in the data component that contain data records.

FREE CI COUNT

This is the number of control intervals in the data component that are marked as free by the sequence set in the index. Because these control intervals are marked as free, CA Disk does not attempt to read them.

UNUSED CI COUNT

This is the number of control intervals that do not contain any data records but that are not marked as free in the index. The significance of this is that the control interval does have a compressed key pointing to it from the index, and therefore can only be used by records that have keys that fall into the range of the control interval. This condition can occur in some database management systems, such as IMS or CICS, when the DFP control interval reclamation facility is disabled. This number can be an indication of wasted space in a cluster, particularly in files that have their primary activity occur randomly throughout the cluster. If this number becomes significantly large in a data set, reorganization can be in order so that the unused control intervals can be reclaimed.

CI %—USED

This value is the percentage of control intervals allocated to the data component that contain at least one data record.

The remaining portion of this print line has several estimated values listed. These values are based upon the data found in the cluster and assume that the same data characteristics will apply to future records added to the data set.

Usable Free Bytes

This is the number of bytes that can be used to store new data records. It does not include space in unreferenceable control intervals, or dead space in control intervals (either from spanned records or unusable space). This should give a true picture of what space is available in which to add records.

Estimated New Record Additions

There are two values that CA Disk can calculate for a cluster, depending on whether or not the cluster contains any spanned records. (The cluster must actually have at least one spanned record, not merely be defined with the SPANNED attribute.) If the cluster has no spanned records, the only value to be calculated will be the number of non-spanned records that can be added. This is based on both free space in used control intervals and also on all unused control intervals. For control intervals that already contain data records, the calculation is based on the free space in the control interval and the average record size of records processed to that point.

For unused control intervals, the value is based on how well records fit into the used control intervals.

When a cluster contains spanned records, the calculation becomes more difficult. In effect, CA Disk tries to simulate the addition of both spanned and non-spanned records at the same percentage as that which occurs in the data itself. Without going into the elaborate details, CA Disk basically does the following:

- Any free space in used control intervals is assignable to non-spanned record additions only.
- Spanned records require whole numbers of free control intervals to be allocated, based on the average number of control intervals required to hold a spanned record.
- An iterative process is used by CA Disk that simulates the addition of records throughout the unused portion of the data set at the prescribed percentage.

Because of the method used in simulation, the percentage of new record additions can vary slightly from the actual percentage found in the data.

Estimated Dead Space

This is the number of bytes that CA Disk estimates are not usable for data storage. The value is based on both absolute dead space (such as unused bytes in the last segment of a spanned record or unused space in data sets with fixed-length records) and assumed dead space (in variable-length files, space that is available for record expansion but that is not large enough to contain a data record). Keep in mind that the estimated dead space in a control interval can be reclaimed later by a control interval split.

MIN RECSZ

This is the size of the smallest record that was found in the data set.

MAX RECSZ

This is the size of the largest record that was found in the data set.

AVG RECSZ

This is the average size of each record in the data set. For fixed-length files, MIN, MAX and AVG record sizes will be the same value.

MIN RECS/CI

This value is the minimum number of records that were found in any used control interval.

MAX RECS/CI

This value is the maximum number of records that were found in any used control interval.

AVG RECS/CI

This value is the average number of records that were found in each used control interval.

MIN FSPC/CI

This is the minimum number of unused bytes found in any used control interval. This space can actually be considered dead space in other calculations of this report if CA Disk determines the space cannot be used to hold data records. Control intervals that contain a segment of a spanned record are ignored in all four of the freespace per CI values.

MAX FSPC/CI

This is the maximum number of unused bytes found in any used control interval.

AVG FSPC/CI

This is the average number of unused bytes found in any used control interval.

AVG—% FSPC/CI

This is the average percentage of freespace to use space in all control intervals read by CA Disk.

AVG CI OVHD/REC

This value represents the average number of bytes required for control information for each data record. This information is required by VSAM to access the records but is not part of the user's data. This value can be useful in estimating storage requirements when reorganizing a cluster (add this value to average record size to come up with an accurate value of space required to hold each data record).

MAX CI DEV

This value represents the maximum relative deviation between any two control intervals when processed in key sequence.

AVG CI DEV

This value represents the average relative deviation between control intervals when processed in key sequence. A value of 1.0 represents the *best case* of each control interval in key sequence physically residing next to one another. As this number increases in size, so does the disorder of the data set.

SPANNED RECORD INFORMATION

A separate print line will be printed for all clusters that have spanned records. Because of the poor performance associated with spanned records — both in CPU and DASD utilization — these values should be checked closely. If a fairly large percentage of records are spanned, you should consider increasing the control interval size of the data set (this cannot be an option in an IMS or CICS-type environment).

TOTAL SPAN RECORDS

This is the total number of spanned records that reside in the data set.

% of TOTAL RECS THAT ARE SPANNED

This is the percentage of records in the data set that do span control intervals.

MIN SPAN RECSZ

This is the size of the smallest spanned record in the data set.

MAX SPAN RECSZ

This is the size of the largest spanned record in the data set.

AVG SPAN RECSZ

This is the average size of all spanned records in the data set.

SPAN DEAD SPACE

This is the amount of dead space in the data set attributable to spanned records. In essence, this is the total number of bytes that are not being used in the last control interval of each spanned record.

CIA Report KSDS Index Component Summary

The summary information for the index component of the data set is for the sequence set only. Since CA Disk processes the cluster in key sequence, it does not need any of the records in the index set, which is used to access the cluster in keyed direct access mode.

CI TOTAL IN SEQ SET

This is the total number of control intervals used to hold the index's sequence set. This also equates to the number of control areas being used in the data set, since each sequence set record controls a single data control area.

FULL CI TOTAL

This is the number of control intervals in the sequence set that have no free control interval pointers to the data component. In other words, this is the number of sequence set records that point to control areas in which all of the data control intervals are being used.

UNREF CI TOTAL

This is the total number of data control intervals that are currently — or are projected to be in the future — unreferenceable because the index's control interval size is too small. For more information, see Unreferenceable Data Control Intervals.

FREE CI TOTAL

This is the number of control intervals in the index component that are currently unused.

MINIMUM FSPC/CI

This is the minimum amount of free (unused) space found in any sequence set control interval.

MAXIMUM FSPC/CI

This is the maximum amount of free (unused) space found in any sequence set control interval.

AVERAGE FSPC/CI

This is the average amount of free (unused) space found in all sequence set control intervals.

AVERAGE COMP KEYL

This is the average number of bytes to which each key in the sequence set is compressed. This value will give an idea as to how effective the compression algorithm is. It is of most value when looking at what control interval size to use for the index. If the keys are compressing well and the MINIMUM FSPC/CI value is sufficiently high, you can want to consider reorganizing the cluster and specifying a smaller control interval size for the index.

AVG INX ENTRY LTH

This value is similar to the AVERAGE COMP KEYL, except it includes the overhead information required to store keys in the sequence set. Therefore, if you subtract the AVERAGE COMP KEYL from this value, you will have the average number of bytes of control information overhead that is required for each index entry.

MAX DATA CA DEV

This value represents the maximum relative deviation that occurred between two consecutive data control areas in key sequence. In a completely organized cluster with no control area splits, this value will be 1. See the description of CA DEV in the graphic analysis section for a more detailed explanation of this value.

AVG DATA CA DEV

This is the average relative deviation of all control areas in the data component. Again, the ideal value is 1.0. As the number gets higher, greater seek distances are required to process the cluster in key-sequenced mode. This value is not relevant to direct key retrieval, since the sequence set is not used. Also, if the data set is on a high activity volume, this value can be meaningless, since head movement will probably occur between control area reads to satisfy requests for other data on the pack.

CIA Report ESDS Catalog Information Section

This section of the report is identical to the catalog section described for KSDS clusters, with the exception of four fields that are not applicable:

- Free %-CI
- Free %-CA
- CI splits
- CA splits

See the description provided for KSDS clusters for a description of the catalog fields.

CIA Report ESDS Graphic Analysis

As with the catalog information, the graphic analysis section for ESDS clusters is very similar to the data component analysis for KSDS clusters. Since the graphs themselves are identical, we don't need to go over them again, but the column totals are slightly different.

REC TOTAL

This is the total number of data records found in the control interval. When the CI contains a portion of a spanned record, the literal SPAN will appear in front of the record count. A value of zero is displayed for the first and any intermediate segments of the spanned record, and a value of one is displayed for the last segment.

BYTE TOTAL

This is the total number of characters found in all of the data records in this control interval.

DEAD SPACE

This is the number of bytes that CA Disk considers to be unusable in the control interval. Since ESDS clusters can only have records added to the end of the data set, any unused space in control intervals prior to the last one is dead space (that's why there's no freespace column on this graph). You can see some ESDS clusters — particularly IMS, CICS, and VVDSs — with a lot of dead space shown for them. These data sets are preformatted with dummy records (for various reasons) and do not look identical to *pure* ESDS clusters. Because of this, CA Disk cannot give an accurate picture of free and dead space in the last control area of the data set.

OVERHEAD

This is the number of bytes required to hold the system control information.

% DEAD

This is the percentage of the control interval considered to be unusable. It is calculated by dividing the number of dead bytes by the control interval size.

CIA Report ESDS Data Component Summary

As with the preceding sections of the ESDS report, this section is basically a subset of fields described for the KSDS data component. See that section of the manual for a description of the appropriate fields.

CIA Report RRDS Catalog Information Section

The catalog information for relative record data sets is also similar to KSDS and ESDS clusters. The two new fields are LRECL and SLOTS/CI. Since RRDS clusters must have fixed-length records by definition, LRECL gives the length of each record *slot* in the data set. The field SLOTS/CI tells how many record slots fit in each control interval.

CIA Report RRDS Graphic Analysis

The graph provided for relative record clusters is identical to the preceding cluster types in format, with the exception that spanned records are not supported by VSAM. You will therefore never see a second legend line under the graph box.

REC TOTAL

This is the total number of used or *active* slots (records) in the control interval.

BYTE TOTAL

This is the total number of characters found in all of the active slots in this control interval. This is a simple calculation of used slots times the logical record length.

EMPTY SLOTS

This is the number of slots that are currently marked as empty in the control interval.

CIA Report RRDS Data Component Summary

The summary information for relative record data sets is relatively straightforward, especially compared to the previous cluster types.

TOTAL RECORDS

The total number of data records in the data set. This is the number of used or *active* slots that contain user data.

TOTAL BYTES

This is the total number of bytes or characters contained in the data records of the cluster.

TOTAL SLOTS IN DATA SET

This is the total number of slots that exist in the data set. It combines the total used and the total empty.

SLOTS %—USED

The percent used tells what percentage of the slots in the cluster are already being used.

FREE SLOTS PAST HIGH RRN

This number tells you how many empty slots exist beyond the last slot that is being used.

USED CI COUNT

This is the number of control intervals that occur prior to and including the last control interval that contains a data record.

UNUSED CI COUNT

This is the number of control intervals between the last control interval that contains a data record and the end of the data set (high allocated RBA).

CI %—USED

This value is the percentage of control intervals in the data set that occur prior to and including the last control interval that contains an active record.

SPACE %—DEAD

Unlike the other types of clusters, this value is a constant value for each data set. It is dependent on control interval size and the number of slots per control interval. The amount of dead space in any RRDS's control intervals can be calculated as follows:

$(CI\ size - 4) - ((LRECL + 3) \times Slots\ per\ CI)$

Analyzing the CIA Report

Obviously there is a lot of information provided in these reports, especially the report for key-sequenced data sets. It's probably even more obvious that the information isn't much good unless it can be interpreted and put to good use. Unfortunately, it is impossible to develop absolute guidelines for interpreting all of this information and applying it universally to all data sets. We can, however, point out some things to look for in these reports if you are trying to tune your VSAM environment. We will begin with the simplest report (RRDS) and proceed to the ESDS and KSDS reports.

The one performance item that all three of these reports have in common is the EXTENTS value in the catalog information section. This will tell you how many extents the cluster occupies. A high value will not only impair performance when processing the data set, but it also leads to fragmented storage volumes. If possible, clusters should be archived and restored to consolidate the extents into one contiguous area.

Analyzing RRDS Relative Record Data Sets

There's really not much that can be done to tune RRDS clusters. Other than checking the extent count, the only other value of interest is the dead space percentage printed in the summary section. If an inefficient control interval size is used, an inordinate amount of space would be wasted. Using the formula listed under SPACE %?DEAD, try substituting different control interval sizes to see which one produces the least dead space.

Keep in mind the type of processing that is normally used on the data set. For direct processing, it is generally more efficient to use smaller control intervals, while sequential processing is more efficient with larger control interval sizes. Even though a particular CI size cannot use DASD as efficiently as another, it can be a better choice if the data set is used frequently.

Analyzing ESDS Entry-Sequenced Data Sets

As with the other types of VSAM clusters, entry-sequenced data sets generally perform better with larger control interval sizes, both in terms of read/write performance and DASD utilization. As with the other types also, this will vary depending on the primary access mode used to process the data set — either direct or sequential. Since not many applications process entry-sequenced data sets in direct mode by RBA address, the larger CI size will probably be more effective. This cannot be an alternative, however, if the data set is used by IMS or CICS. Your installation can have requirements limiting the CI size to decrease the system resources required for VSAM buffers. If this is the case, do not change CI size without checking first (with IMS and CICS databases it can cause problems if you change the CI size anyway, regardless of whether the buffer space is a restriction — it depends on the use of the ESDS cluster).

A very important set of values is printed if the data set is defined with the SPANNED attribute and contains at least one spanned record. Because of the tremendous overhead involved with spanned records (both performance and DASD), they should definitely be the exception and not the rule in a cluster. Check the percentage of spanned records in the data set. Depending on the cluster, even 1 percent can be too high for performance reasons. Check the minimum, maximum, and average spanned record size. It can be possible to redefine the cluster with the next-larger control interval size and eliminate the spanned records altogether. We must again, however, raise the caveat concerning IMS and CICS clusters: Don't change the CI size of the cluster unless the cluster can tolerate being reorganized with a new CI size, and unless the buffer space is available for your online applications.

Analyzing KSDS Key-Sequenced Data Sets

Key-sequenced data sets are the hardest of the cluster types to tune properly because of all the variables involved. Since the parameters governing tuning are installation- and data set- dependent, it would be impractical (if not impossible) to list an exhaustive set of criteria to follow. We will try to list the more important values to check.

- As with the other cluster types, check the number of extents the data set occupies. Consolidate extents with an archive and restore to reduce volume fragmentation. (Archiving the data set to disk and then restoring it will avoid operator intervention to mount archive tapes!)
- As with entry-sequenced data sets, see if the data set has any spanned records. If it does, consider increasing the CI size of the data component if practical. Again, be careful with IMS and CICS databases — you can be limited as to the buffer space that can be used. Also, if the cluster is accessed primarily in an online environment, the spanned records can still be a proper choice because of the extra overhead involved in transferring larger control intervals.
- Check the index's control interval size. Many clusters get defined with a CI size that is too large for the index. If the CI size is greater than 512, verify that the index is indeed the proper size. This can be done in part by looking at the minimum freespace per CI value and at the average index entry length found in the index's statistics summary section. If the minimum freespace is less than the difference between the current and the next-lower valid CI size (valid CI sizes for the index are 512, 1024, 2048, and 4096), you are already at the lowest CI size. If this is not the case, multiply the average index entry length by the number of control intervals per control area in the data component (field CI/CA in the catalog information section). This will give you a rough idea of the space needed for indexing the average data control area. There is one important point to keep in mind: if you make the CI size too small, you can force VSAM to more index levels, which is undesirable in an online, direct access environment. There is an entire section devoted to optimizing VSAM's performance in IBM's *VSAM Programmer's Guide*, which discusses the impact of control interval size on performance. Before doing any tuning on clusters, this entire section should be read thoroughly.
- Compare the CI size to the physical block size. The optimal situation is for the two to have the same value. If not, a different control interval size can be in order. Again, this cannot be an option with IMS and CICS databases. Consult the *VSAM Programmer's Guide* for a table showing the correspondence of CI size to physical block size for space utilization guidelines (in the section on optimizing VSAM's performance).
- For clusters used extensively in sequential access mode (as opposed to direct key), check the average control area deviation value. In a data set with a large number of CA splits, the seek movement can contribute significant overhead. You can want to consider reorganizing the cluster, and can be also increasing the freespace percentages at restore time to reduce the frequency of splits in the future.

- Look at the unreferenceable CI count. If it is other than zero, you can be wasting DASD space. It is impossible to tell from just the statistics summary if the space is currently unusable or if it is a projection of future problems. Remember that CA Disk simulates the process that VSAM uses when allocating new control intervals from the freespace pointers. Therefore, there can indeed be enough freespace for current usage, but CA Disk is predicting a problem with future additions. If you want to know if the problem exists currently, look at the sequence set graph that is produced. Under the UNREF CIs heading you will see the sequence set records that have unreferenceable CIs. If the FREESPACE value for the control interval is less than the AVG ENTL, the space problem exists right now.
- Look at the dead space percentage. If a high percentage of your data set contains dead space, you should consider going to a higher data control interval size. As a rule, DASD utilization is more efficient at the higher CI sizes. Also, if spanned records are present in the data set, they can be contributing significantly to the problem. Again, consider increasing the CI size.
- Look at the space usage percentage and estimated record additions values. For data sets that are continually expanding, these numbers will give an indication that reorganization with additional space can be in order. This can keep your data set from going into additional extents or, worse yet, running out of space.

SMF Report

To obtain this report, execute a four-step job that:

- Selects abbreviated data from SMF type 14 and 15 records
- Sorts the selected data into data set name sequence
- Converts start/stop date and time into grids
- Writes the report

Note: The SMF report requires its input from a sequential data set. If your SMF records reside in a VSAM cluster, the supplied SMFRPT procedure should be modified to add an IBM IDCAMS utility step at the beginning to REPRO the records into a sequential data set to be passed to CA Disk. Verify that this has been done before executing the report for the first time.

Syntax

Only one SELECT command per execution is allowed.

```
//SMFDATA EXEC SMFRPT
//SYSIN DD *
        SELECT COUNT=,DSN=,START=,STOP=,CLOSE,TAPE
```

COUNT=

The maximum number of SMF records to be read in this execution, specified as a 1- to 10-digit number between 1 and 2147483647. SMF records past this number will be bypassed. If not specified, this parameter will default to 10000000.

DSN=

A list of one or more data set names or patterns for which a report is desired. Omission of the parameter results in SMF records for all data set names being selected. (A maximum of 50 entries is supported.)

START=

The date in an accepted CA Disk format for which reporting is to begin. Records with a start date earlier than this date will be bypassed.

STOP=

The date in an accepted CA Disk format for which reporting is to terminate. Records with a close date later than this date will be bypassed.

CLOSE

The presence of this simple parameter causes the start date to be set equal to the stop (SMF CLOSE) date. Otherwise, the start date is obtained from the reader time stamp field in the SMF type 14 or 15 record.

TAPE

The presence of this parameter results in tape data sets being included in the report. The volume serial number field is be filled with blanks for all tape records.

The generation and version numbers will be reset to zero for all GDG data sets.

JCL Example

```
//SMFDATA EXEC SMFRPT
        SELECT COUNT=1000,TAPE
```


Chapter 6: Backup/Archive

This section contains the following topics:

[About Backing Up and Archiving](#) (see page 233)
[General Information](#) (see page 234)
[BACKUP and ARCHIVE Considerations](#) (see page 238)
[Non-VSAM Considerations](#) (see page 244)
[VSAM Considerations](#) (see page 245)
[Incremental Backups](#) (see page 257)
[BACKUP Command](#) (see page 262)
[SMS and BACKUP](#) (see page 268)
[BACKUPCC Command](#) (see page 269)
[ARCHIVE Command](#) (see page 272)
[SMS and ARCHIVE](#) (see page 277)
[ARCHIVE Command Examples](#) (see page 278)
[DELETE Command](#) (see page 280)
[DEL3435 Command](#) (see page 282)
[EXPIRE Command](#) (see page 283)
[Sample Backup/Archive/Delete Report](#) (see page 287)
[Explicit ARCHIVE](#) (see page 291)

About Backing Up and Archiving

CA Disk provides comprehensive, efficient, common processing of mixtures of non-VSAM data sets and VSAM clusters. Within a single execution you can back up, archive, delete and expire partitioned (PO), partitioned extended (POE and PO2), physical sequential (PS), physical sequential extended (PSE), direct access (DA), indexed sequential (ISAM), and OS CVOL catalog non-VSAM data sets. You can also back up, archive, delete and expire VSAM clusters, and back up ICF catalogs.

You can do the backups and archives in either an immediate mode, or in a deferred mode in which the back up or archive is actually done at a later time.

You can also direct the output to different media types. For instance, by default backups and archives are written to tape. However, by overriding a couple of sysparms, they can easily be directed to disk. For details, see the step-by-step instructions in Archiving to Disk: Requirements and Recommendations in the *Systems Guide*.

You can do this based on: scans of VTOCs of disk volumes, scans of ICF or VSAM (but not OS CVOL) catalogs, or the results of a catalog superlocate.

You also have a rich set of criteria to select data sets for and exclude data sets from processing as desired, and a rich set of action command parameters to let you control exactly what you want to do.

Commands Available	JCL Proc Executed:
ARCHIVE	ARCHIVE
ARCHIVE	DMS
BACKUP	DMS
BACKUPCC	DMS
DARCHIVE	ARCHIVE
(dispose)	ARCHIVE with overridden EXEC parm
DERASE	DERASE
DELETE	DMS
EXPIRE	DMS
LISTREQ	LISTREQ

General Information

These functions provide the backup, archive and delete capabilities for non-VSAM data sets, VSAM clusters and ICF catalogs at the data set level. For information on volume-level backup and recovery, see the chapter "VBACKUP/VRECOVER."

Under your control, the CA Disk archives can be on tape, cartridge or disk.

Note: Recent advancements in the cartridge technology field have substantially improved the CA Disk Backup/Archive processes. Cartridge capacities have increased allowing Data Centers to maintain a smaller number of ARCHVOLS. For an example of this technology, see the section's Storage Tek Redwood Device and IBM'S Magstar Tape Device in the chapter "General Information."

Terminology

The terminology used in this chapter is:

Backup

Backing up (copying) data from your online system to the CA Disk archives, leaving the original data set in place.

Incremental Backup

The process of backing up only those data sets that have been modified since the previous backup. This type of backup runs faster than a full volume backup and does not needlessly use space in the CA Disk archives for redundant copies of data sets that have not changed.

Archive

Backing up data sets to the CA Disk archives, and then deleting the original. This is equivalent to *moving* or *migrating* the data set from your online system to the CA Disk archives. From the CA Disk archives, data sets can be *restored* or *recalled* to your online system, or data sets can eventually be *expired* or *rolled off* the archives.

Delete

Refers to deleting the original data set without making a backup copy in the CA Disk archives.

Expire

Refers to deleting an expired data set without making a backup copy in the CA Disk archives. The expiration date can reside in any of the following areas:

- Format-1 DSCB for non-VSAM
- Catalog entry for VSAM
- Management Class for SMS-Managed

These concepts are implemented with the BACKUP, ARCHIVE, DELETE, and EXPIRE commands described as follows. These commands in turn run under the umbrella of DSCL, documented in the section Data Storage Command Language in the chapter "Basic System – DMS PROC."

Depending on how you specify selection criteria and the action commands, you can use these backup/archive/delete/expire features to perform these and other data storage management tasks:

- Back up full volumes of data sets.
- Back up changed data sets.
- Move unused data sets to the CA Disk archives on tape or disk.
- Move old versions of GDG data sets to the archives.
- Delete old system temporary data sets from work volumes.
- Delete uncataloged data sets.
- Delete unused data sets.
- Delete expired data sets.
- Verify that data sets are on authorized volumes.

Concurrent Copy Support

Concurrent Copy is a function of DFSMS/MVS designed to eliminate prolonged outages of critical online data. CA Disk provides support for this function through its DSCL BACKUPCC command. By invoking this command, CA Disk will back up your critical data even while the data is being updated by online applications.

For a complete overview of this support, see the section Concurrent Copy Support in the chapter "General Information."

Using the Concurrent Copy Backup Support

Before the CA Disk backup process can begin; a synchronization point must be established to ensure the integrity of the data being backed up. This is a five step process:

1. Create and/or verify that the DSCL command stream with the action command BACKUPCC is ready for execution.
2. The online system targeted for BACKUPCC must be quiesced. This ensures that all data set I/O buffers are written to disk.
3. Execute the BACKUPCC job created in step 1.
4. Verify that the INITIALIZATION COMPLETE message issued from ADSDI106 was sent to the SYSLOG. This message indicates to the online administrator that it is now safe to resume online processing.
5. Restart the online system.

The Backup Process

The Concurrent Copy support groups all candidate data sets with the BACKUPCC command into a single backup process. When you start the command, all data sets matching the selection criteria are backed up as a unit. CA Disk manages these units as synchronized backup sets.

Although the entire backup process can take hours, the DFSMS/MVS Concurrent Copy function monitors the process through its completion. This ensures that even though multiple updates to the data can occur during the backup process, the actual data being backed up is exactly the way it appeared when the INITIALIZATION COMPLETE message was issued.

If the BACKUPCC command fails at any point during its execution, or if the MVS system on which the job is running fails, you must restart the entire backup process again (see Using the Concurrent Copy Backup Support). This requirement ensures that a correct synchronized Concurrent Copy is established.

Structure of the BACKUPCC Command

For detailed information on the command structure used in supporting Concurrent Copy, see BACKUPCC Command.

Special Considerations

Concurrent Copy processing of CICS, DB2, and IMS systems will differ, depending on which release of these products you have. For more information on this subject, see the section Processing CICS, DB2, and IMS in the chapter "General Information."

DFSMSdss Data Mover Interface

This interface provides CA Disk users a convenient way to backup HFS and zFS data set organizations. Data sets backed up using this interface are maintained on standard CA Disk archive volumes creating normal ARCHVOL and DSNINDEX records. CA Disk utilities, such as IXMAINT, are used as normal to manage them.

Note: DFSMSdss must be active and at a sufficient operating level (MVS/SP 5.22 with DFSMS 1.2 or above). Otherwise, CA Disk bypasses the selected data sets and issues an appropriate error message.

DSCL Selection Criteria

Using this interface is the only way CA Disk can back up HFS or zFS data sets. With this interface active, CA Disk automatically switches to the DFSMSdss data mover when HFS or zFS data set types are detected. CA Disk switches back to the normal CA Disk data mover for all other data set types.

If you want to select only the HFS type data sets, use the following SELECT statement:

```
SELECT CRITERIA=(PDSORG,EQ,HFS)
```

To select only zFS type data sets, use the following SELECT statement:

```
SELECT CRITERIA=(ZFS,EQ,YES)
```

Note: HFS data sets are only backed up at the container level. You cannot back up individual data sets within an HFS container.

Activation

The DFSMSdss Data Mover interface is controlled by sysparm USEDSSIO. For details, see its sysparm description under USEDSSIO in the *Systems Guide*.

Limitations

Since CA Disk is more flexible than DFSMSdss, not all of the existing CA Disk BACKUP parameters are supported by DFSMSdss. Whereas some of these limitations are not appropriate for HFS data sets anyway, the complete list is as follows:

- ACCESS=
- MEMBER=
- MODE=
- NEWHLQ=
- NEWNAME=
- RPT=
- SELTYPE=

BACKUP and ARCHIVE Considerations

The following paragraphs detail the BACKUP and ARCHIVE considerations. Each option suits a different need.

ARCRELTM SYSPARM Releases Tapes When Restore is Waiting

If an Archive/Backup job can delay a Restore, use SYSPARM ARCRELTMnn. Specify an Archive/Backup release time of 01 to 99 minutes (default is 00) to use in both Archive and Restore. During Archive, if ARCRELTM is non-zero and there is room on the current volume for the next data set, the time interval since the last output volume swap is compared to ARCRELTM.

If that time interval is greater than or equal to ARCRELTM, a test ENQ for DMSWVOLS is issued for that volume. If a Restore has been waiting for that volume, Archive releases the volume. Restore only issues the DMSWVOLS ENQ if ARCRELTM or MERRELTM is non-zero. Tuning can be done within Archive/Backup with ARCRELTM from 01 to 99 minutes, depending on individual needs.

Change Bit

IBM's SU60 formally defined and introduced support for a *change bit* DS1IND02 in the format-1 DSCB for each non-VSAM data set. The CA Disk SVC recognizes and supports this same change bit. Both the CA Disk SVC and SU60 intercept processing for a data set that is being opened for output and turn on the change bit. The CA Disk incremental backup function identifies the modified data sets by this bit in each VTOC entry. As each changed data set is identified, CA Disk creates and indexes a backup copy, and then optionally turns the change bit off.

Midnight "Rapid Aging" Problems

When CA Disk jobs that examine date fields are run very shortly after midnight, you must take into account the *rapid aging* that occurs at the stroke of midnight. Data sets that can have been created or used merely minutes ago suddenly appear as if the action took place a day ago.

When batch jobs create data sets, the create date used by the operating system is the date the job started, not the date that each specific step of the job started. Hence data sets actually created in steps running after midnight, but part of a multi-step job that began before midnight, will have create dates of the previous day—and immediately appear to be one day old!

Any selection criteria based upon date values should take these two items into account, or you can end up selecting many more data sets for action than you intended.

Exempting Data Sets

Data set names or data set name patterns that are to be exempted from selection should be specified by an EXCLUDE command (see the section EXCLUDE Command in the chapter "Basic System – DMS PROC" for the command format.)

Exempting Restored Data Sets

If your installation runs archives regularly, you can find that data sets that were restored recently are being archived again. This is because the same selection criteria that caused a data set's original archival can cause it to be re-archived after it has been restored, if the data set is not modified before the next archive run.

Normally the condition causing the data set's archival is changed by the requestor, either when it is restored or very shortly thereafter. However, you can also specify an installation option that allows a grace period to be given to any restored data set before it will be re-archived. For more information, see the sysparm descriptions for RESRETPDnnnnn and RETEXCLn in the *Systems Guide*.

Note: This grace period does not apply to backups, since they do not delete the data set from disk.

Backup of Volume VTOCS

If you specify the SELECT VTOCS and BACKUP command, CA Disk will back up VTOCs from all volumes that are scanned, regardless of whether or not any data sets from these volumes are selected for processing. A backup copy of a VTOC can subsequently be used by RECOVER processing to restrict the recovery to only those data sets listed in the backup copy of the VTOC, excluding from recovery those data sets that were deleted from the volume prior to the backup run.

CA Disk, does not back up the index for an indexed VTOC because it is not needed. It contains no useful or helpful data for either the archive or the restore function. The entries in the index for a VTOC are automatically rebuilt at restore time due to the standard allocation interfaces used by CA Disk.

The following restrictions apply to this support:

- Only the most recent copy of a volume VTOC found in the CA Disk archives will be used as the screening list during recovery processing.
- The VTOC copy cannot be used to overlay the current VTOC on a volume. CA Disk dynamically rebuilds each VTOC entry as each data set is restored.
- If you have a CA Disk Security Interface installed, you will need access to the data set name VTOC.volser, where volser is the volume on which the VTOC resides. Both BACKUP and IXMAINT functions will query any CA Disk Security Interfaces for authority to process this name.

If you are going to backup VTOCs, the following security setup should be done prior to setting RACF protectall on:

- Create profile with ADDSD VTOC.* UACC(NONE) to provide a profile so that the name is legal. The OPERATIONS attribute of the userid associated with user#1's backup job will then provide access to that data set name. The data set name is fictitious and it refers to nothing on spinning DASD. The data set name is only a name so that CA Disk can manage the VTOC backups it optionally takes to help volume recovery operations.

Task Performance

You can execute multiple backup or archive jobs concurrently on the same or different CPUs. By default, CA Disk generates unique archive data set names during dynamic allocation of the ARCHIVE0 and ARCHIVEC dd statements. This prevents the different jobs from waiting for a tape data set name enqueue held by another job.

If you provide the ARCHIVE0 or ARCHIVEC dd statements in your JCL and specify sysparm ARCTNAME with a value of *D*, CA Disk will append a 22-character date/time stamp to the name specified in the JCL. See Naming the Archives for details.

Each backup or archive job should also specify a list of volumes to be processed that is different from the other jobs. To reduce channel contention among the jobs, the volume lists should accomplish as much channel separation as possible.

Graceful Shutdown

Backup and archive jobs can be shutdown gracefully by using sysparm DSCLRPLY. A graceful shutdown will allow the jobs to complete all work in progress and produce the normal reports for error conditions encountered and work successfully completed before the jobs terminate. If jobs are canceled instead of being gracefully shutdown, data sets could be left in a partially processed state and the normal reports will not be produced.

With DSCLRPLY active, a message is issued to the operator at the beginning of the job and remains on the console for the duration of the job. If the operator replies, *END* to this message, it indicates processing is to be stopped early. For details, see the section DSCL Graceful Shutdown in the chapter "Basic System – DMS PROC."

Task Restart Considerations

After each DASD data set is written to tape successfully, a CA Disk index entry is placed in the DSNINDEX subfile and the delete and catalog actions are taken as needed. With this design, any tape created by CA Disk is a good tape, even though a system failure or job cancel can have occurred.

If your installation uses a tape library management system such as CA 1 or CA Dynam/TLMS, you must ensure that they do not designate such tapes as scratch tapes. EDM support was designed for this contingency. For more information, see the user exit description for TMSCLEX-Tape Management System Control Interface Exit in the *Systems Guide* or contact your tape management vendor.

If a job fails, the exact same archive or incremental backup command request can be resubmitted. CA Disk will request a new scratch tape and processing will continue where it left off. In reality, all volumes and all data sets are actually rescanned for selection, but if they were successfully selected and processed in the immediately preceding partial run, most selection types will not reselect them. For example, an unused data set that was archived and deleted will not be seen by the restarted job. Changed data sets backed up successfully will no longer have the change bit on and will therefore not be reselected, unless the NORESET parameter is also specified.

If an archive or backup fails because the data set is in use, the data set can be automatically queued for deferred processing by specifying sysparm ARCFDEFR with a value of Y. For more information on this sysparm, see ARCFDERn in the *Systems Guide*.

ARCHIVE Report Sequencing Options

The archive report can be produced in three sequences:

- Data set name sequence
- Data set name sequence within volume
- Exact processing sequence (unsorted)

By default, all three reports are produced. See sysparm ARCHSORTudv in the *Systems Guide* if you want to change the defaults. Any combination of the three report sequences can be specified.

Data Set Compression and Reorganization

When a data set is being read from disk, CA Disk processes the data blocks within it *sequentially* as dictated by the organization type. Thus only active and used data blocks are read. This results in PDS, ISAM and VSAM data sets being copied to the archive tape in compressed format. Any subsequent restore or volume recovery results not only in the consolidation of several extents into one (defragmentation), but also in the compression/reorganization of the data within the single extent (eliminating the additional need to compress and reorganize these same data sets).

Fast Archive Processing

When Fast Archive processing is enabled, CA Disk can reuse existing backups or archives to back up a data set with no data movement. CA Disk checks the data set to determine whether it has been modified. The product uses the same variables since the most recent Archive/Backup that the Modified criteria uses. When an eligible data set is found, a new DSNINDEX record is created using a new Archive/ Backup date and time. The ARCHVOL information from the existing DSNINDEX record is flagged as a FAST record. The ARCHVOLS record and tape expiration are updated to keep the longest expiration date but not all dates are compatible. The new Fast Archive DSNINDEX record is kept until IXMAINT expires it and merges forward if processed by Merge. All copies of the DSNINDEX are treated as different versions for Restore and IXMAINT processing.

Note: Verify that Sysparm DSNDUPVL is set to Y if you are using volume pooling. Setting DSNDUPVL to Y prevents the older DSNINDEX versions from being selected.

Non-VSAM Considerations

The following section details special processing options that can be used to tailor the non-VSAM backup and archival functions.

Backup of SYSCTLG

CA Disk will back up OS CVOL catalogs (DSN=SYSCTLG), but you will probably need to specify sysparm ARCDORG with a value of Y. This is because most SYSCTLG data sets have no DSORG indicated in the VTOC DSCB. Due to the embedded TTR values, SYSCTLG data sets must be restored to the same device type from which they were archived. (The Move function, described in MOVE/COPY, has the ability to move these data sets to different device types, updating the internal TTR pointers.)

PDS Anomalies

CA Disk does extensive testing for errors and inconsistencies in partitioned data sets, and provides installation options for handling PDSs in which logical errors have been detected. Standard default processing is to issue diagnostic messages for the errors, and then bypass them. Consult with your local CA Disk installation and support staff to determine what options are in effect if any of your PDS data sets are diagnosed to have anomalies.

This facility provides an automated means of deciding which data sets on disk should be backed up or archived, and which should be bypassed. This facility does not support PDSE data sets. For more information, see Processing PDSs that Contain Anomalies in the *Systems Guide*.

Model DSCB Processing

Data sets with zero extents and no secondary allocation amount are most commonly used as model DSCBs. They can be identified with the DSCL selection criteria MODEL DSC.

Since model DSCBs are not opened and used like other data sets, their modify bit and last used date are never modified. Because they take up no space on the volume, by default CA Disk will not archive model DSCBs unless you specify sysparm ARCMODEL with a value of Y. Regardless of the setting of ARCMODEL, CA Disk backs up and restores model DSCBs.

PSE Considerations

Extended sequential data sets are simply multi-volume sequential data sets that are managed by DFSMS. This combination resulted in some special considerations that each user should be aware of. A complete list of these considerations is located in the section Special Considerations While Processing PSEs in the chapter "Basic System – DMS PROC."

VSAM Considerations

The following section details special processing options that can be used to tailor the VSAM backup and archival functions.

Verify of VSAM Clusters

While attempting to open a cluster for input, CA Disk can receive a failure from open processing indicating that the data set was not properly closed by a previous task. When this error is encountered, an IDCAMS VERIFY is required to synchronize the end-of-file marker in the data set with the value found in the catalog. By specifying sysparm VSVERIFY with a value of Y, CA Disk will automatically issue a verify when this condition is encountered and will then proceed to reopen the data set and continue archival processing.

DFP automatically issues a verify against a cluster if the need is detected at open time. Since this feature is viewed as an integrity exposure to some, the user can turn this feature off for CA Disk processing by specifying sysparm VSSYSVER with a value of N (the default). Note that if this feature is turned off, CA Disk can still be instructed to issue the verify by specification of sysparm VSVERIFY with a value of Y.

Backing Up or Archiving Empty Clusters

CA Disk will archive any valid cluster that passes selection testing, regardless of whether or not it contains data. If the cluster is empty, only its attributes are copied to tape.

Note: Empty SMS-managed VSAM will not be selected for Archive/Backup processing, unless sysparm VSUNOPEN is specified with a value Y. For details, see the sysparm description for VSUNOPENn in the *Systems Guide*.

CA Disk selects a VSAM cluster for Archive/Backup processing if the volume serial for any component or extent of the cluster meets the volume selection criteria. This occurs even if other components of the cluster do not meet the volume selection criteria.

The cluster-level selection criteria must also be met before the cluster is selected for processing. However, CA Disk will not select a VSAM data set for Archive/Backup processing more than once within the same job step.

The files data set contains the volume serial of the first data component of a VSAM cluster. During Archive/Backup processing, CA Disk issues a message that shows this volume serial. The volume serial must be correctly specified for a RECOVER run that uses the VOL= parameter. CA Disk can only RECOVER clusters if the first data component volume is selected.

Never Opened Clusters

A never opened VSAM cluster is one that has been defined but has never been opened for output. These are empty clusters because their VSAM TIMESTAMP in the catalog entry and the last use date in the format-1 DSCB are both zero.

By default, these clusters will not be selected for backup or archive. If you want to select these clusters for processing, you can do so by specifying sysparm VSUNOPEN with a value of Y.

With VSUNOPEN set to Y, CA Disk compares CREDIT to ARCDT. If the CREDIT is greater than or equal to the ARCDT, CA Disk will select these clusters with the criteria MODIFIED,EQ,YES.

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.

For more information, see the sysparm description for VSUNOPENn in the *Systems Guide*.

Processing Clusters with Alternate Indexes

CA Disk automatically retrieves all of the information necessary to recreate a base cluster's alternate index(es) and/or path(s), and will archive the information with the base cluster. In addition, the alternate index can be processed as a separate entry from the cluster (which is referred to as a physical alternate index). If you intend to process physical alternate indexes, see Alternate Index Support and the section Restoring Alternate Indexes in the chapter "RESTORE/RECOVER."

Deleting Unexpired Data Sets

By specifying sysparm VSMPURGE with a value of Y (the default value), CA Disk allows deleting unexpired data sets. This is equivalent to using the PURGE parameter on an IDCAMS DELETE request.

Processing Clusters in Use by Other Tasks

Normal CA Disk processing determines the disposition type to use for data set allocation based on the share options with which a cluster is defined. If a cluster is defined with SHAREOPTIONS (1 3), it is allocated with DISP=SHR; otherwise it is allocated with DISP=OLD. By specifying sysparm VSALOSHR with a value of Y, all allocations will be made with DISP=SHR. In addition, the user exit USERVENQ can be invoked to determine the allocation desired on the data set level.

Note: Since CA Disk does not issue ENQs for VSAM data sets, it cannot be held responsible for the integrity of a data set that was archived with a disposition type other than that which it would have selected. CA Disk relies solely on the integrity features offered by VSAM through its share options.

Recataloging for VSAM Auto-Restore

When a VSAM cluster is archived and recataloged to the CA Disk pseudo-volume, alias entries are defined for each of the alternate index and path names as well as the data and index component names. This enables the auto-restore function to be invoked when you reference, the base cluster, a component name, an alternate index, or a path name.

Restore processing removes the catalog entry to the pseudo-volume and all of the aliases associated with it, and redefines the correct VSAM catalog entries. Should the restore process fail, CA Disk attempts to recreate the initial status by recataloging the cluster to the pseudo-volume and associating the proper alias names to it.

VSAM Backup and Archive Restrictions

KSDS clusters that use the KEY-RANGE attribute are processed, but if the data set was archived prior to Release 7.8 of CA Disk, the KEY-RANGE attribute is not retained when restored.

VSAM and ICF catalogs cannot be archived. However, backup copies of ICF catalogs can be taken.

Relative Record data sets (RRDS) that contain more than 16,776,000 active slots cannot be archived.

If the proper catalog cannot be located via standard MVS catalog searching orders, the //STEP CAT dd statement must be provided.

Access Methods for VSAM Clusters

During backup and restore processing, CA Disk processes VSAM data at the cluster level rather than the data space level. Logically, data processing accesses records within a file. Physically, the file is a group of areas (extents) on one or more DASD volumes. For VSAM, the cluster entity is equivalent to the logical file, while the data space is equivalent to the physical storage. Processing data at the physical (data space) level makes the data dependent on both the storage medium (device type) and the method of access.

An archive system can retain data offline for an extended period of time and when restoration of the data file becomes necessary, the storage medium can no longer be available. Also, in the event of a disaster, the recovery site cannot have the same physical medium or access method available. Processing data at the logical file level is one way of providing data independence; that is, the capability to restore the logical data to a different physical medium or possibly using an alternative method of access. In addition, logical processing allows the data to be reorganized as a by-product of processing.

CA Disk uses normal access methods to read VSAM clusters. It can read them in one of two ways:

- Logical record processing
- Control interval processing

The method CA Disk will use is controlled by sysparm VSACCESS. The basic difference in the methods is that with logical record processing, CA Disk issues requests for individual data records—it is up to VSAM to read the control intervals and perform buffer management functions. With control interval access, CA Disk requests specific control intervals that it must, in turn, deblock into their component logical records. This deblocking process can be suppressed, however, when necessary (such as when processing DB2 data bases).

With control interval access, CA Disk performs its own buffer management. The size of the buffer area is equal to the size of one data component Control Area size.

CA Disk will read all Control Intervals sequentially within a Control Area and store them before processing the logical records within the Control Intervals. This technique of reading the entire Control Area is done to take advantage of the read ahead functions of VSAM. By reading the Control Intervals sequentially, VSAM will automatically read the entire Control Area when the first read is issued. The result is a single read I/O for each Control Area processed.

ICF Catalog Backup and Recovery

Backup and recovery of catalogs is a complex and critical procedure. To assume that it can be reduced to a process that is as simple as *pushing a button* is to make a serious mistake! The myriad of dependencies among the operating system, the type of catalog (OS CVOL, VSAM user catalog, or ICF catalog), and the data related to the catalog entries almost always demands unique recovery steps to obtain a catalog that is truly useful.

Recovering a catalog from a backup copy can be meaningful or useful for both ICF catalogs and OS CVOLS, but rarely (if ever) for VSAM user catalogs. The old VSAM user catalogs have many dependencies such as time stamps and volume ownership that make it nearly impossible to use them if they are recovered independently from all of their clusters. The ability (IDCAMS functions) to re synchronize them is just not available. IBM addressed these concerns by providing ICF catalogs, and the enhanced abilities in IDCAMS to re synchronize them. For these reasons, CA Disk provides the ability to automate and extend the catalog backup process for ICF catalogs, but you must provide the knowledge and analysis to see that the correct recovery steps are taken—steps that suit your particular case and achieve the results you desire.

Backing up ICF catalogs can be automated simply by specifying `SELECT CATALOGS=` in front of a `BACKUP` command, or by adding the `BACKUP` parameter to the `SCAN CATALOGS=` command. If a `SELECT` statement is not specified with the `SCAN CATALOGS=` command, CA Disk will backup ICF catalogs. As each catalog is selected for processing, CA Disk links to the IDCAMS `EXPORT` function, intercepts the records, and writes them in CA Disk format to the archive media. To extend or enhance the backup and recovery process, the `ALIAS` entries for the catalog are also extracted from the master catalog and appended to the exported catalog data. This provides CA Disk with the ability to dynamically redefine them in the master catalog per the `DEFALIAS=YES/no` parameter on the `RECOVER` command. Without this ability, all alias pointers in the master catalog must be rebuilt manually because they are deleted (lost) when a catalog is imported.

Parameter `EXPORTF=NO/yes` is also available for the `RECOVER` command. Specifying `YES` instructs CA Disk not to `IMPORT` the catalog, but to create a sequential data set on disk that can be used as direct input to a native `IMPORT`; that is, the data set will be in *export format* that `IMPORT` can use. When running with this option, messages are issued to identify the `ALIAS` entries that you can need to redefine in the master catalog after you actually do the `IMPORT`. (As mentioned in previous, a backup copy and an `IMPORT` done under DFP 2.3 or above should preserve the `ALIAS` entries. For earlier versions of DFP you will need to redefine them manually.) This option can be used as just described, but is also provided for use with IBM's catalog forward recovery utility (ICFRU 5798-DXQ), which requires an `EXPORT` copy as input.

It should be evident from the descriptions previous that CA Disk automates and extends the use of IBM's IDCAMS EXPORT and IMPORT functions to provide for much easier backup and recovery of ICF catalogs. It is not the intent of this documentation, however, to present a tutorial or guideline regarding a multitude of recovery scenarios, the problems encountered in each, and the steps necessary to recover successfully. Other publications or classes specifically designed to address the complexity of these topics should be consulted if needed. Your analysis and planning should be based upon the abilities provided by EXPORT and IMPORT (as automated and extended by CA Disk) and the other ICF resynchronization functions within IDCAMS.

If a catalog being recovered still exists on disk, IDCAMS will overlay it only if its temporary export flag is on. Otherwise the existing catalog must be deleted before the IMPORT can be done. See sysparm CATBKDELn in the *Systems Guide*, for an option to have CA Disk attempt this delete automatically for you. A *DELETE usercat UCAT RECOVERY* command is issued.

If a catalog being recovered does not exist on disk, but its entry in the master catalog still exists (for example, the volume containing the catalog has been reinitialized due to a hardware failure), you will probably need to issue an IDCAMS "EXPORT usercat/masterpassword DISCONNECT" command before the recovery will succeed.

One very important planning item to consider is the following. Depending upon which catalog is lost, the ability to submit and initiate jobs can be impossible without hardcoding volume information in the JCL (because the catalog normally used to provide this information is the one that was lost). One of these jobs can be the CA Disk recovery run, or perhaps the IDCAMS IMPORT job! Being prepared to provide the volumes in the JCL for either of these jobs can be essential!

IDCAMS can dynamically allocate each ICF catalog as it begins the EXPORT function, but it does so with DISP=OLD and does not deallocate (free) the catalog when it is done (it waits for step termination to do so). To avoid what can be a lengthy tie-up, CA Disk uses its own dynamic allocation to allocate each needed catalog, and then deallocates (frees) it immediately after the EXPORT. For importing a catalog, however, CA Disk lets IDCAMS perform its own allocation.

Before recovering an ICF catalog and its associated clusters, review the value specified for sysparm VSPREDEF carefully. The default value of N prevents CA Disk from overwriting preallocated (or in this case, pre-existing) VSAM clusters during restore processing. If it is specified with a value of Y (not recommended), CA Disk will overwrite the existing clusters with the REUSE attribute. However, if the information in the VVDS is incorrect or not present, this can cause errors in recovering the VSAM clusters. The best method would be to leave VSPREDEF with the default value (N). Recover the ICF catalog and the clusters. Then restore the exceptions (if necessary) using the SCRATCH parameter of Restore.

Processing IMS Databases

CA Disk provides several options that can prove beneficial in managing IMS/VS databases that consist of VSAM data sets. The ARCHIVE (BACKUP) and RESTORE functions can be used as a backup alternative to IBM's image-copy utility. The speed of CA Disk processing and the data compression feature provide savings in both time and space. If directed to disk, substantial disk space is saved. If directed to tape, both the number of tape mounts (operator intervention) and the quantity of tapes used can be reduced substantially.

You can also use IBM's image-copy and direct its output to disk, thus avoiding operator intervention, and then use either the sequential migrate to tape function or the archive function to consolidate several disk data sets onto a single tape.

In order to produce a valid backup copy of an IMS/VS database, all of the data sets that comprise the database must be backed up individually. It is important that no updating be allowed to occur against the database until the backup of all of its component data sets has completed. If updating should take place before a complete database backup is obtained, a loss of synchronization between the various portions of the database can result if a recovery is attempted. Appropriate care must be exercised when establishing backup procedures to ensure that all of the related database data sets are identified.

The IMS/VS Backout or Recovery Utility has been changed to use the key values, instead of the RBA pointers, during recovery. Therefore, no extra considerations are necessary when using CA Disk to back up and restore IMS/VS databases. This is because the automatic reorganization CA Disk does at restore time no longer presents a problem for the IMS/VS Backout or Recovery Utility.

IMS databases do not have alternate indexes as defined by VSAM. The *secondary indexes* in an IMS database are actually separate KSDS clusters that IMS updates as an index. When an IMS database is initialized, each CI is formatted and appears to VSAM (and also to CA Disk, image-copy and IDCAMS) to contain a single logical record. This logical record/CI is updated by IMS. As updates are made to the database, these indexes can undergo CA/CI splits.

Whenever an IMS cluster is moved or restored, take care to ensure that the CI size is not changed. Default parameters prevent CA Disk from changing the CI size. The CI size can be changed inadvertently by unknowingly requesting a CI size change in the RESTORE command. It can also be changed by VSAM allocation routines when the CI size is invalid for the device type being used. Any change in CI size will destroy the IMS database unless specifically planned and accounted for by the database administrator. Therefore, do not specify sysparm VSATTRIB, which allows IDCAMS to select a CI size every time the cluster is defined, when processing IMS databases.

Processing Controlled by CA Compress

If you have VSAM clusters compressed with CA Compress for MVS product, see sysparm SHRNSYxxxx in the *Systems Guide*. The sysparm documentation covers how CA Disk and CA Compress can work together to improve CA Disk performance and make it easier to maintain the CA Compress FDT.

Alternate Index Support

There are two methods CA Disk can use to archive and restore alternate indexes. The first method is always performed and cannot be overridden: the catalog information required to redefine an alternate index is always archived with its base cluster. When the base cluster is restored, the alternate index can be redefined from this information. In this case, an entry is made in the CA Disk FILES for only the base cluster and not its alternate index(es).

The second method of archiving an alternate index is to explicitly name it on an archive command with sysparm VSARCAIX specified with a value of Y, or to run a backup of the base cluster with sysparm VSARCAIX specified with a value of Y. This method causes the alternate index to be physically processed as a data set; as such it will have its own entry in the CA Disk FILES and the data records from the alternate index will be read and archived along with the catalog information. Using this latter method, the cluster index will be referred to in this section as the *physical* alternate index.

If a disposition of DELETE is specified or assumed on the command, alternate indexes will not be archived. This is to ensure the alternate index will not be deleted before the base cluster is processed, which would remove the association from the base cluster.

The alternate index can always be redefined and built based on the catalog information archived with the base cluster. There are special processing considerations required to restore a physical alternate index, and they are detailed later in this section of the manual. Prior to reading the sections on restoring VSAM clusters and alternate indexes, make sure you understand the difference between restoring a physical alternate index as opposed to rebuilding the alternate index from catalog information saved with the base cluster.

It should be noted that clusters that were archived with a release of CA Disk prior to Release 7.5 will not have the alternate index information with the base cluster. It is therefore necessary to redefine the alternate indexes of these clusters manually after the base cluster is restored by CA Disk.

DBRC Support

CA Disk also provides an option that will automatically detect when a data set under DBRC's control has been processed. The appropriate control statements (NOTIFY commands) are also produced for the DBRC update utility to inform it of the action that has been taken. For example, if CA Disk is used to replace IBM's image-copy utility as mentioned previous, this option will cause DBRC to be notified that the CA Disk backup copy has been taken and should be used for recovery purposes. See your local CA Disk installation and support staff to determine if these options have been implemented.

For important information, see sysparm descriptions that begin with *DBRC* in the *Systems Guide*.

Processing DB2 Databases

VSAM files created through the DB2 database management software normally observe the naming convention of using *DSNDBC* as the second-level node of the data set name. CA Disk automatically detects these data sets and formats them in control interval images rather than logical record format.

A DB2 system consists of the following files:

Catalog Files

LDS data sets that consist of a collection of tablespaces that contain information on objects defined to DB2. This catalog is where backup and reorg information (that is, image copy backups) is stored.

Bootstrap Files

The bootstrap data set (BSDS) is a KSDS file used to keep track of the RBAs during startup and shutdown. This file is also used for recovery and restart operations.

Log Files

A logging system that records all changes (that is, COMMITS) into an ESDS database.

Data Tablespace

One or more LDS files (data tables) that make up one or more DB2 Tables. Tables can be simple, partitioned, or segmented.

Index Tablespace

A set of pointers that reside in one or more physical LDS files (index tables). There are unique indexes that protect against two rows in a table having the same key value. Cluster indexes are used to determine how rows are divided between partitions and control the order in which table rows are stored.

CA Disk supports all of the files mentioned previously. CA Disk can be used to logically backup, restore, and move both system related (Log, Bootstrap, Catalog) and application related (Data and Index tablespace) files.

Even though CA Disk does not interface to the DB2 catalog, sites using DB2 release 2.2 and 2.3 can use CA Disk as a backup supplement to the DB2 IMAGE Copy Utility. For example, page level incremental backups can be made with DB2 IMAGE Copy and Tablespace level backups with CA Disk by using the SELECT TABLE= parameter.

CA Disk can be used to restore large or partitioned tables in the event of a DASD failure or data corruption instead of using IMAGE Copy backups.

DB2 release 3.1 has the ability to perform a Log-only recovery. This allows a DB2 table to be restored from a non-DB2 utility such as CA Disk. After CA Disk has restored the table; the user has the option to specify the starting log RBA from where to apply the log changes. The speed and flexibility of CA Disk over the DB2 utilities makes the restore of large tables both easier and faster. This, in turn, reduces the downtime for the application.

CA Disk has the ability to backup VSAM files that are in use by specifying sysparm VSALOSHRy. However, it is best to have the files backed up closed or in read-only status, or the DB2 system can be down altogether.

Processing Clusters that have Non-Standard Formats

Certain program products use VSAM but have a non-standard internal format in the VSAM control interval. Among those known to have such a problem are IBM's DB2 database management system, IBM's linear data sets (LDS) and certain data sets created by MSA products. CA Disk, must process these data sets in a special manner.

The typical control interval in the data component of a VSAM cluster looks something like this:

- Record n = data record (actual user data)
- Free space = space that is unused in the control interval
- RDF = Record Descriptor Field. There can be one for each data record in the control interval, but this is not always the case.
- CIDF = Control Interval Descriptor Field. This field normally describes the amount of space being used by the data records portion, as well as the number of free space bytes remaining.

When CA Disk processes a VSAM data set, it can process in one of two modes: logical record or with control interval access. With logical record access, CA Disk issues requests for individual data records and will never see the RDF and CIDF fields. With control interval access, CA Disk is returned the entire control interval with each read request; it must deblock the control interval using the CIDF and RDF control information.

The problem with these non-standard data sets is that they do not maintain the CIDF and/or RDF records, or they use their own techniques in managing the internal data. When CA Disk (or IDCAMS, for that matter!) tries to process these records using logical record techniques, unpredictable results will occur. There is no way of knowing that the CIDF and RDF are not present, or that they have assumed a different meaning from what the initial design of VSAM intended.

The way to avoid these problems is to read the data set with control interval access (as CA Disk has been able to do for some time) and then archive the entire control interval as a single record. In this way CA Disk can ignore what should be the CIDF and RDF records; it does not attempt to interpret the data inside the control interval. The only disadvantage to this technique is that at restore time the control interval size cannot change. This means that CA Disk must put the cluster back exactly the way it archived it.

By default, CA Disk always writes archived data sets in logical record format. CA Disk automatically detects data sets that do not observe logical record definitions, such as DB2 data sets, Oracle data sets and some data sets created by MSA products. When one of these data sets is found, CA Disk automatically shifts to control interval image copy format when archiving the data set. If you have VSAM data sets in non-standard formats that CA Disk cannot detect, you have two options for processing these data sets. You can code sysparm VSARCFMT with a value of C, which causes all VSAM ESDS data sets to be processed in control interval image copy format, or you can code your own user exit called VSACESEX to selectively determine which mode of archiving to use for each data set.

At this time, entry-sequenced data sets (ESDS) are the only type supported by control interval image copy format. KSDS and RRDS clusters are always archived in logical record format.

Physical Sequential Data sets that have Non-Standard Formats

Certain applications create non-standard physical sequential data sets. CA Disk bypasses these data sets and identifies the non-standard characteristic in messages 0435 and/or 3990. Typical non-standard characteristics look something like this:

- BLKSIZE is not a multiple of LRECL for a data set with RECFM=FB
- RECFM=F with UNEQUAL LRECL/BLKSIZE and BLKSIZE is not a multiple of LRECL

You cannot force CA Disk to process non-standard data sets with the exception of RECFM=F and BLKSIZE is a multiple of LRECL, but you can suppress these annoying messages from MSGPRINT. For details, see ARCNOSTDn in the *Systems Guide*.

Incremental Backups

The CA Disk incremental backup function is implemented as a selection criteria MODIFIED under DSCL.

Volume Level Full Backups

Your installation can also want to include volume-level backup runs as part of its backup plan. This function is described in the chapter "VBACKUP/VRECOVER."

CA Disk volume-level backup or recover runs do not use the traditional data set-level processing technique that is used by normal DSCL backup and archive processing. Therefore, running the volume-level backup will take much less time than the full volume backup method described earlier in this section. However, because VBACKUP and VRECOVER functions are volume-oriented, you cannot use the volume-level backup tape to restore individual data sets.

Therefore, you can want to use the volume-level backup as your primary backup protection only for volumes with a high activity level; that is, most of the data sets are changed every day or every week. This will save time when running the daily or weekly backups.

If you need to restore data sets on this type of volume, you can issue the VRECOVER command with the UPDATE parameter specified, which will restore the entire volume, then update the modified data sets from your incremental backups. For more details, see the chapter "VBACKUP/VRECOVER."

Implementing Incremental Backup

The following questions and answers represent some of the typical concerns that need to be addressed in order to implement incremental backup.

How do I get an initial copy of all my data sets?

By using CA Disk to produce a full data set backup. Running does this:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
PASSWORD MASTER=the-CA Disk-master-password  
SCAN REALVOLS  
SELECT VTACS  
SELECT CATOLOGS  
SELECT DSNAMES=/  
BACKUP
```

How do I establish proper retention periods for the Archive/Backup volumes, and all of the data sets contained on those volumes?

First, recall how CA Disk for non-VSAM data sets, assigns retention periods. As each data set is copied to the archives (on tape or disk), an expiration date is assigned to its index entry. The value of the expiration date is determined as follows: A default expiration date is calculated using the default retention period value of sysparm RETRETPD, or from the RETPD or EXPDT parameter if it is present on the BACKUP command.

In addition to each archived copy of a data set being assigned its own retention period, the containing archive volume (the tape, or a sequential data set on disk) is also assigned an expiration date. CA Disk default processing assigns the never scratch Julian date of "99365" to each of the archive volumes. This ensures that they will not be written over until all contained archived data sets have expired. The DSNDELETE function of index maintenance will return tape volumes to scratch status (and delete backup or archive data sets kept on disk) when it determines that all contained data sets have indeed expired.

For important information, see the user exit description for TMSCTLEX-Tape Management System Control Interface Exit in the *Systems Guide*.

You can assign an alternate expiration date for archive volumes that are dynamically allocated by specifying sysparm DYNEXPDT with either an expiration date or retention period value. You can also provide the //ARCHIVE0 or //ARCHIVEC dd statement in your JCL. The value you specify for the RETPD or EXPDT parameter will be assigned to the archive volume.

For more information, see the section Year 2000 Considerations in the chapter "General Information."

With this in mind, it is appropriate to decide what the default retention period should be — 15, 30, or however many days.

Since non-SMS incremental backup creates a new copy only when a data set is modified, and assigns only that copy an expiration date, how can I protect that copy from expiring — and thus being dropped from the archives — if the data set is not modified again sometime within the retention period of the prior copy?

(For example, a read-only data set will never have new backup copies generated after its initial backup if backup uses only the change bit selection criteria.)

One technique is to coordinate the retention period assigned to each archived data set with the frequency with which you take full volume backups.

For example, assume we want to ensure that there will always be one backup copy of every data set, and full volume backups should be done only once a week. This can be accomplished by taking a full volume backup one day each week and incremental backup runs every intervening day. Both the full volume backup and the incremental backup runs should specify `RETPD=8` on the `BACKUP` command. The incremental backup run should also specify `MODIFIED,EQ,YES` on the `SELECT` command.

A second and more efficient technique can be used if the CA Disk SVC is being used to record the last modify date in the data set's VTOC entry at the same time the change bit is being turned on.

Using this technique, instead of doing full volume backups once a week (usually on the weekend), CA Disk will select and back up only those data sets that have NOT been modified during the seven-day cycle. This eliminates time wasted in taking redundant copies of data sets that have already been backed up because they changed during the week.

For example, on Monday through Saturday, back up only the data sets being changed (`MODIFIED,EQ,YES`). On Sunday, back up the changed ones, but also back up any data set NOT changed within the last seven days by using this statement:

```
MODIFIED,EQ,YES,OR,MODDT,LT,TODAY-7
```

To ensure that you always have at least one copy of every data set in the archives, set the `RETPD` value to 15 days. A data set changed (and backed up) on Monday won't be picked up again by the *unmodified in seven days* test until the second following Sunday.

What if I want to keep two backup copies in the archives?

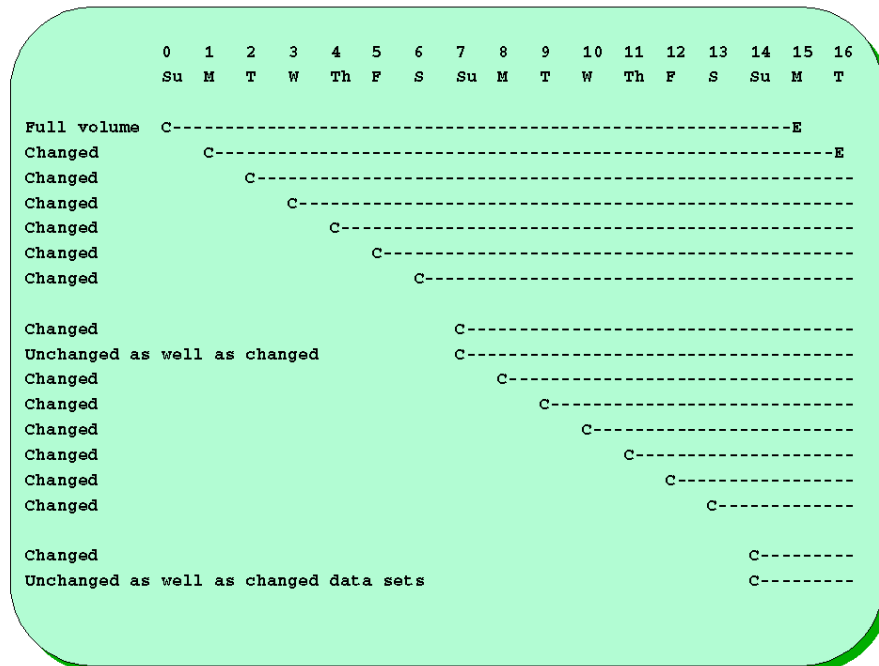
For the previous example, simply specify `RETPD=22` on the `BACKUP` commands.

In general, set the `RETPD=` parameter to $(X+1)$ times the number of days between the runs that pick up the unmodified data sets, where X is the number of backup copies you want to ensure. Add 1 to the result to allow a day overlap.

For example, with monthly runs to pick up the unmodified data sets, and ensuring one copy, use `RETPD=63` (that is, $(2*31+1)$). To ensure two copies, use `RETPD=94` (that is, $(3*31+1)$).

Can you draw a picture to demonstrate this?

Consider the simple case of weekly runs to pick up the unmodified data sets, using RETPD=15 to ensure one copy of all data sets. In the two-week following illustration, C represents the creation of a tape and E its earliest possible expiration (the RETPD value).



Notice that if a data set existed on Sunday (day 0) but is never modified, a new copy is generated each successive Sunday, before the previous tape (and version of the data set) has expired. If the data set is modified every day, each incremental backup tape will also contain a copy. At the end of the week, there can be from one to seven copies of the data set, but a minimum of one is ensured at all times. A data set modified on Monday (day 1) will not be backed up again until Sunday (day 14), but the prior copy is being kept until Tuesday (day 16), so one copy is guaranteed.

Also notice that running incremental backup on a daily basis is a somewhat arbitrary factor so far as it concerns the previous objective. Running it twice daily or only twice a week ensures the minimum copy goal as long as the RETPD= value is the same. But, most important, notice that full volume backups are never needed again!

How to archive MERGE and index maintenance fit into this scheme?

It depends on the characteristics of your data sets and the archive volumes that are created (be they tape or disk data sets). If all data sets backed up are assigned the same RETPD, the entire archive volume (for example, a tape) will expire at the end of the period. If each tape is also nearly completely filled, deleting the expired data sets is preferred because MERGE will accomplish very little. However, if the percentage of tape used in several cases is quite low, merging them can indeed be useful. See the chapter "MERGE" for a better understanding of this capability.

Incremental Backup Restrictions

The following restriction should be observed when planning your installation's incremental backup plan. This restriction applies solely to the incremental backup function, not to backup processing in general.

Data sets marked UNMOVABLE are not supported by CA Disk and are automatically bypassed. An alternative backup means must be used. For details, see the sysparm description for UNMOVEARn in the *Systems Guide*

BACKUP Command

The BACKUP command instructs CA Disk to create a backup copy of an online data set, and to index this copy in the CA Disk FILES. Only those data sets satisfying the DSCL selection test(s) will be subject to CA Disk processing.

Using the MODE parameter on the DSCL SET command can simulate the backup function. For details, see the description of the parameter MODE= in the section SET Command in the chapter "Basic System – DMS PROC."

Deferred Requests

Requests to back up a data set can be deferred over a period of time until a batch DISPOSE run can be performed to process all requests deferred since the previous batch run. Use of this facility is desirable, in that a single backup tape mount is required instead of individual mounts for each data set.

A deferred request is made by adding the DEFER parameter to the BACKUP command.

Deferred backup requests are queued and kept in the ARCHCMDS subfile of the CA Disk FILES. After a queued request has been processed, its completion status is maintained for a default period of five days, after which time it is deleted automatically during the next DISPOSE run. Through the ISPF panels, the saved status can also be displayed online. Otherwise, the output from the DISPOSE run or a batch LISTREQ must be inspected to determine the outcome of processing for the request. See Sysparm DARCKEExxx to alter the default period for which the status is maintained in the ARCHCMDS file.

Member Level Backup

A subset of a partitioned data set can be designated by specifying a list of one or more of its members or member alias names. Associated member and alias names are always maintained, regardless of which are specified in the list.

A member level backup is requested by adding the MEMBER= parameter to the BACKUP command. The MEMBER parameter cannot be used with the DEFER parameter.

BACKUP Command Syntax

The syntax of the BACKUP command is as follows:

```
BACKUP RETPD=, EXPDT=, ACCESS=, RPT=CIA, CIAOPT=, NORESET, DEFER, NEWNAME=, NEWHLQ=
      MEMBER=, SELTYPE=, MODE=, FAST=
```

Plus the following parameters honored for PRIMARY2 processing
only: RETPD2=, EXPDT2= Parameters Honored with the PRIMARY2 Parameter of SET

RETPD2=

Specify a one- to five-digit number indicating the number of days the selected data sets is to be retained in the CA Disk archives. The date calculated from this retention period cannot exceed the year 2155. For more information, see the section Year 2000 Considerations in the chapter "General Information" and the sysparm description of EXPDTUSEn in the *Systems Guide*.

If multiple BACKUP (and ARCHIVE) commands are entered, each can have a different RETPD2= or EXPDT2= value. You cannot specify both the RETPD2 and EXPDT2 parameters

EXPDT2=

Specify a date constant in any acceptable CA Disk format. For a list of valid dates, see the section Date Formats in the chapter "General Information." For more information, see the sysparm description of EXPDTUSEn in the *Systems Guide*.

CA Disk treats DSNINDEX Julian expiration dates *99000* (except as described in the next paragraph), *99365*, *99366* and *99999* as *never scratch* dates. That is, data sets whose ARCHVOLS or DSNINDEX records use these dates as expiration dates will not be deleted automatically as expired by the CA Disk function IXMAINT. For more information, see the section Year 2000 Considerations in the chapter "General Information."

By default, CA Disk treats Julian expiration date *99000* as a never scratch date. 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 are deleted automatically by the CA Disk function IXMAINT when the data sets become uncatalogued. What this means is, if your catalog goes bad or someone uncatalogs some data sets, all backups and archives of these data sets will be deleted on the next IXMAINT run. Although you can use EXPDT2=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 DISP=RECAT,EXPDT2=99000
```


If multiple BACKUP (and ARCHIVE) commands are entered, each can have a different RETPD2= or EXPDT2= value.

You cannot specify both the RETPD2 and EXPDT2 parameters.

RETPD=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a one- to five-digit number indicating the number of days the selected data sets are to be retained in the CA Disk archives. The date calculated from this retention period cannot exceed the year 2155. For more information, see the section Year 2000 Considerations in the chapter "General Information" and the sysparm description of EXPDTUSEn in the *Systems Guide*.

If multiple BACKUP (and ARCHIVE) commands are entered, each can have a different RETPD= or EXPDT= value. You cannot specify both the RETPD and EXPDT parameters.

EXPDT=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a date constant in any acceptable CA Disk format. For a list of valid dates, see the section Date Formats in the chapter "General Information." For more information, see the sysparm description of EXPDTUSEn in the *Systems Guide*.

CA Disk treats DSNINDEX Julian expiration dates 99000 (except as described in the next paragraph), "99365", 99366 and 99999 as *never scratch* dates. That is, data sets whose ARCHVOLS or DSNINDEX records use these dates as expiration dates will not be deleted automatically as expired by the CA Disk function IXMAINT. For more information, see the section Year 2000 Considerations in the chapter "General Information."

By default, CA Disk treats Julian expiration date 99000 as a never scratch date. 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 are 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, all backups and archives of these data sets are 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 DISP=RECAT,EXPDT=99000
```

If multiple BACKUP (and ARCHIVE) commands are entered, each can have a different RETPD= or EXPDT= value.

You cannot specify both the RETPD and EXPDT parameters.

ACCESS=

You can override the value of sysparm VSACCESS for this BACKUP command by specifying ACCESS= with a value of C for Control Interval (CI-level) access, or L for logical-level access.

RPT=CIA

This option allows the Cluster Internal Analysis (CIA) report to be produced as a by-product of live mode backup of VSAM clusters. The CIA report will not be produced during simulate runs. See the section CIARPT – Cluster Internal Analysis Report in the chapter "Reports" for further information.

CIAOPT=

If the RPT=CIA parameter is specified, this parameter can be used to specify which options are to be used. See the section CIAOPT = (CATLG, DETAIL, DUMP) in the chapter "Reports" for further information.

NORESET

If a data set is backed up, normal processing resets (to zero) the change bit DS1IND02 in the format-1 DSCB. Specification of this parameter will cause processing to bypass updating the change bit. This feature would be useful for full-volume backup runs for off-site storage, where you do not want to prevent the data set from being processed by the next incremental backups.

NEWNAME=

An alternative data set name to be assigned to the DSNINDEX record for the backed up data set (a relative GDG name is also permitted). The backed up version is indexed under this name, which can be up to 44 characters in length.

DEFER

This parameter can be specified on the BACKUP command to indicate that the request is to be queued for later processing. For an overview of this function, see Deferred Requests.

NEWHLQ=

An alternative high level qualifier can be assigned to each data set selected for backup (a relative GDG name is also permitted). The backed up version is indexed in the FILES under this name, which can be up to 44 characters in length.

MEMBER=

To back up a subset of a partitioned data set, this parameter is used to specify one or more member and/or pattern names to constitute the backed up version of the data set. A valid pattern for member level processing is any character followed by the '/' pattern character. This is the only valid pattern character available for this parameter. As an example:

MEMBER=(A/,GEN/)

For detailed information on the '/' pattern character, see Slash (/) Pattern in DSN= in the section Data Set Names and Parameters in the chapter "General Information."

A maximum of 25 entries is supported and is only valid when an immediate backup is being done (not a deferred backup).

For an overview of this function, see Member Level Backup.

SELTYPE=

This is for CA—Select customers only. See SELTYPE= for a description of this parameter.

MODE=

This is for CA—Select customers only. See MODE= for a description of this parameter.

FAST=

Specify YES or NO to indicate whether the data sets selected for processing should use the Fast Archive path.

SMS and BACKUP

Backup uses SMS Storage Group and Management Class information to determine the eligibility of a data set for backup.

The SMS fields that CA Disk supports are as follows:

- Auto-backup (SG) — This required field can be defined Y or N. The default is Y. When this field is defined Y, CA Disk will automatically allow backup. When this field is defined N, CA Disk will not allow backup of any of the data sets for the Storage Group.
- Guaranteed Backup Frequency (SG) — This field can be defined 1 to 9999, or NOLIMIT. When this field is defined as a number, CA Disk will not allow more than that number of days to pass without backing up the data set. When this field is defined NOLIMIT, Management Class attributes will be used to process the data set. See the sysparm description for SMSGBRST99 in the *Systems Guide* for deletion of previous Guaranteed Backup copies when a new Guaranteed Backup is taken.
- Backup Frequency (MC) — This field can be defined 0 to 9999, or blank. The default is 1. When defined as number that number of days must pass before CA Disk will back up the data sets that have changed since the last backup. When this field is defined with a blank, other Management Class attributes are used to process the data set.
- Admin or User Command Backup (MC) — This required field can be defined BOTH, ADMIN, or NONE. The default is BOTH. When this field is defined BOTH, both the Storage Administrator and the user can explicitly back up a data set. When this field is defined ADMIN, only the Storage Administrator can explicitly back up a data set. When this field is defined as NONE, neither can explicitly back up. To define a Storage Administrator to CA Disk, use the SMSSTGAD sysparm.
- Auto-backup (MC) — This required field can be defined Y or N. The default is Y. When this field is defined Y, the data set is eligible for implicit backup. When this field is defined N, it is not. For example:

```
SCAN REALVOLS  
SELECT SGNAME=/  
BACKUP
```

accomplishes the following:

- Select the volumes contained in all Storage Groups.
- Select the data sets meeting the appropriate Management Class and Storage Group attributes.
- Backup the data sets.

BACKUPCC Command

The BACKUPCC command instructs CA Disk to create a Concurrent Copy backup of an online data set. Using the MODE parameter on the DSCL SET command can simulate it. For details, see the description of the parameter MODE= in the section SET Command in the chapter "Basic System – DMS PROC." However, while in simulate mode, CA Disk will not interface with the DFSMSdss data mover to determine if the backup will be concurrent, whether the hardware is in place or not.

Important! Since this command is executed while the online application is quiesced, time is of the essence. It is for this reason that BACKUPCC must be the only action command in the DSCL command stream. Otherwise the outage to the online application will be longer than necessary.

DFSMSdss Data Mover

The Concurrent Copy support was implemented into CA Disk by interfacing to the DFSMSdss data mover directly. No specific action needs to be taken in order to use this support.

Before using the Concurrent Copy support, you should be aware of the following informational items:

- All messages issued from the BACKUPCC command are CA Disk messages.
- All output ARCHVOLS are written in CA Disk format. They cannot be used as input to a DFSMSdss restore.
- All data set organizations are supported. For a list of these organizations, see the following table.

Value	Definition	Value	Definition
AM	Access Method (VSAM)	PDS	Partitioned
DA	Direct access	PO	Partitioned
HFS	HFS	PS	Sequential
INVALID	Invalid pattern of DSORG bits in format-1 DSCB	SEQ	Sequential
IS	Indexed Sequential (ISAM)	UNKNOWN	No DSORG bits on in format-1 DSCB
ISAM	Indexed Sequential (ISAM)	VSAM	VSAM

- When processing VSAM clusters, only the base cluster is contained in the DSNINDEX record. However, the BACKUPCC sysout will report on all components, such as AIXs. During the restore process, CA Disk will recover the base cluster as a unit, including all paths and alternate index information.
- In order for a candidate data set to be processed, it must be cataloged.

BACKUPCC Command Limitations

The BACKUPCC command is limited to the capabilities of DFSMSdss because CA Disk uses the DFSMSdss data mover interface. One noticeable difference is in the restore process.

Most of the CA Disk RESTORE target allocation parameters are not supported. A data set backed up with BACKUPCC must be restored back to its original volume in its original format. For example:

- For non-VSAM data sets, the POOL= or VOLUME= parameters are not supported. Preallocate the data set before executing the restore to redirect the restore to a different volume.

Note: If you have a system allocation package such as CA Allocate or DFSMS, the portion of this limitation regarding redirection does not exist.

- For the VSAM clusters, the FREESPACE=, DNAME=, parameters are not supported. The only way to alter cluster characteristics is to predefine the cluster before executing the restore.

Note: For PDSE version 2 data sets, DSORG is displayed as POE rather than PO2 on BACKUPCC reports.

BACKUPCC Command Syntax

The syntax of the BACKUPCC command is as follows:

```
BACKUPCC RETPD=,EXPDT=,NOTOLENQF
```

RETPD=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a one- to five-digit number indicating the number of days the selected data sets are to be retained in the CA Disk archives. The date calculated from this retention period cannot exceed the year 2155. For more information, see the section Year 2000 Considerations in the chapter "General Information" and the sysparm description of EXPDTUSEn in the *Systems Guide*.

EXPDT=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a date constant in any acceptable CA Disk format. For a list of valid dates, see the section Date Formats in the chapter "General Information."

For more information, see the section Year 2000 Considerations in the chapter "General Information" and the sysparm description of EXPDTUSEn in the *Systems Guide*.

For a more detailed explanation of this parameter, see EXPDT.

NOTOLENQF

By default, selected data sets will be processed even though shared or exclusive access (ENQ) fails. To override the default, specify this parameter so TOL(ENQF) keyword will not be included in the DFSMSdss DUMP command stream.

Note: If multiple BACKUPCC commands are entered, each can have a different RETPD= or EXPDT= value.

Finally, you cannot specify both the RETPD and EXPDT parameters on the same BACKUPCC command.

BACKUPCC Command Example

The following example issues a Concurrent Copy backup for the production IMS system:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SCAN REALVOLS  
  SELECT DSN=IMSPROD./  
    BACKUPCC RETPD=14
```

ARCHIVE Command

The ARCHIVE command instructs CA Disk to create a backup copy of an online data set, to index this copy in the CA Disk FILES, and to dispose of the online copy as requested. Only those data sets satisfying the DSCL selection test(s) will be subject to CA Disk processing.

Using the MODE parameter on the DSCL SET command can simulate the ARCHIVE function. For details, see the description of the parameter MODE= in the section SET Command in the chapter "Basic System – DMS PROC."

Deferred Requests

Requests for data set archival can be deferred over a period of time until a batch DISPOSE run can be performed to process all requests deferred since the previous batch run. Use of this facility is desirable, in that a single archive tape mount is required instead of individual mounts for each data set.

A deferred request is made by adding the DEFER parameter to the ARCHIVE command. For details, see

Deferred archive requests are queued and kept in the ARCHCMDS subfile of the CA Disk FILES. After a queued request has been processed, its completion status is maintained for a default period of five days, after which time it is deleted automatically during the next DISPOSE run. Through the ISPF panels, the saved status can also be displayed online. Otherwise, the output from the DISPOSE run must be inspected to determine the outcome of processing for the request. See sysparm DARCKEEXxx in the *Systems Guide* to alter the default period for which the status is maintained in the ARCHCMDS file.

Member Level Archive

A subset of a partitioned data set can be designated by specifying a list of one or more of its members or member alias names. Associated member and alias names are always maintained, regardless of which are specified in the list.

Note: While only the specified members are processed, the entire data set will be deleted. If this particular DSNINDEX record is restored, only the members residing in this index record are recovered. Members residing in the partitioned data set prior to the ARCHIVE will not exist in the data set after RESTORE.

A member level archive is requested by adding the MEMBER= parameter to the ARCHIVE command. For details, see the section MEMBER in the chapter "Reports."

Deferred ARCHIVE Command Restrictions

A subset of a PDS (specified by the MEMBER= parameter) is not supported.

ARCHIVE Command Syntax

The syntax of the ARCHIVE command is as follows:

```
ARCHIVE DISP=, RETPD=, EXPDT=, ACCESS=, RPT=CIA, CIAOPT=, DEFER,  
NEWNAME=, NEWHLQ=, MEMBER=, FAST=
```

The following parameters honored for PRIMARY2 processing only:

RETPD2=, EXPDT2=

RETPD2=

Specify a one- to five-digit number indicating the number of days the selected data sets are to be retained in the CA Disk archives. For more information, see the section Year 2000 Considerations in the chapter "General Information" and the sysparm description of EXPDTUSEn in the Systems Guide.

If multiple ARCHIVE (and BACKUP) commands are entered, each can have a different RETPD2= or EXPDT2= value.

You cannot specify both the RETPD2 and EXPDT2 parameters.

EXPDT2=

Specify a date constant in any acceptable CA Disk format. For a list of valid dates, see the section Date Formats in the chapter "General Information." For more information, see the section Year 2000 Considerations in the chapter "General Information" and the sysparm description of EXPDTUSEn in the Systems Guide.

CA Disk treats DSNINDEX Julian expiration dates 99000 (except as described in the next paragraph), 99365, 99366 and 99999 as never scratch dates. That is, data sets whose ARCHVOLS or DSNINDEX records use these dates as expiration dates will not be deleted automatically as expired by the CA Disk function IXMAINT.

By default, CA Disk treats Julian expiration date 99000 as a never scratch date. 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 are 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, all backups and archives of these data sets are deleted on the next IXMAINT run. Although you can use EXPDT2=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 DISP=RECAT,EXPDT2=99000
```

If multiple ARCHIVE (and BACKUP) commands are entered, each can have a different RETPD2= or EXPDT2= value.

You cannot specify both the RETPD2 and EXPDT2 parameters.

DISP=

The disposition for each data set successfully processed. The default value is DELETE.

Specify RECAT for recatalog to the CA Disk pseudo-volume following deletion. This makes the data set available for auto-restore by the CA Disk Catalog Management hook. The data sets can later be made unavailable for auto-restore by the CA Disk Catalog Management hook by the MERGE command UNCATALOG parameter.

Specify NOCAT to have CA Disk take no catalog action against the data set. This causes CA Disk to delete the data set from DASD, and to leave it cataloged to its original volser.

Specify KEEP if the data set is not to be deleted, nor have its catalog status changed. Even better than specifying DISP=KEEP is to use the BACKUP command.

RETPD=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a one- to five-digit number indicating the number of days the selected data sets are to be retained in the CA Disk archives. For more information, see the section Year 2000 Considerations in the chapter "General Information" and the sysparm description of EXPDTUSEn in the Systems Guide.

If multiple ARCHIVE (and BACKUP) commands are entered, each can have a different RETPD= or EXPDT= value.

You cannot specify both the RETPD and EXPDT parameters.

EXPDT=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a date constant in any acceptable CA Disk format. For a list of valid dates, see the section Date Formats in the chapter "General Information." For more information, see the section Year 2000 Considerations in the chapter "General Information" and the sysparm description of EXPDTUSEn in the Systems Guide.

CA Disk treats DSNINDEX Julian expiration dates 99000 (except as described in the next paragraph), 99365, 99366 and 99999 as never scratch dates. That is, data sets whose ARCHVOLS or DSNINDEX records use these dates as expiration dates will not be deleted automatically as expired by the CA Disk function IXMAINT.

By default, CA Disk treats Julian expiration date 99000 as a never scratch date. 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, 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 DISP=RECAT, EXPDT=99000
```

If multiple ARCHIVE (and BACKUP) commands are entered, each can have a different RETPD= or EXPDT= value.

You cannot specify both the RETPD and EXPDT parameters.

ACCESS=

You can override the value of sysparm VSACCESS for this BACKUP command by specifying ACCESS= with a value of C for cluster-level access, or L for logical-level access.

RPT=CIA

This option allows the Cluster Internal Analysis (CIA) report to be produced as a by-product of live mode archive of VSAM clusters. The CIA report will not be produced during simulate runs. See the section CIARPT – Cluster Internal Analysis Report in the chapter "Reports" CIARPT – Cluster Internal Analysis Report for further information.

CIAOPT=

If the RPT=CIA parameter is specified, this parameter can be used to specify which options are to be used. See the section CIAOPT=(CATLG, DETAIL, DUMP) in the chapter "Reports" CIAOPT=(CATLG, DETAIL, DUMP) for further information.

NEWNAME=

An alternative data set name to be assigned to the DSNINDEX record for the archived data set (a relative GDG name is also permitted). The archived version is indexed under this name, which can be up to 44 characters in length.

DEFER

This parameter can be specified on the ARCHIVE command to indicate that the request is to be queued for later processing. For an overview of this function, see Deferred Requests Deferred Requests.

NEWHLQ=

An alternative high level qualifier can be assigned to each data set selected for archive (a relative GDG name is also permitted). The archived version is indexed under this name, which can be up to 44 characters in length.

MEMBER=

To archive a subset of a partitioned data set, this parameter is used to specify one or more member and/or pattern names to constitute the archived version of the data set. A valid pattern for member level processing is any character followed by the '/' pattern character. This is the only valid pattern character available for this parameter. As an example:

```
MEMBER=(A/, GEN/)
```

For detailed information on the '/' pattern character, see the description of Slash (/) Pattern in DSN= in the section Data Set Names and Patterns in the chapter "General Information Slash (/) Pattern in DSN=."

A maximum of 25 entries is supported and is only valid when an immediate archive is being done (not a deferred archive).

Note: While only the specified members are processed, the entire data set will be deleted. If this particular DSNINDEX record is restored, only the members residing in this index record are recovered. Members residing in the partitioned data set prior to the ARCHIVE will not exist in the data set after RESTORE.

FAST=

Specify YES or NO to indicate whether the data sets selected for processing should use the Fast Archive path.

SMS and ARCHIVE

Archive uses SMS Storage Group and Management Class information to determine the eligibility of a data set for archive.

The SMS fields that CA Disk supports are as follows:

- Auto-migrate (SG) This required field can be defined YES, NO, INTERVAL or PRIMARY. When you define this field to YES, INTERVAL, or PRIMARY, CA Disk automatically allows archival. When you define this as NO, CA Disk does not allow archival of any of the data sets for the Storage Group.
- GDG Elements on Primary (MC) This optional field can be defined 0 to 255 or blank. The default is blank. When this field is defined with a number, it indicates the maximum number of most recent generations of a GDG on primary. If there are more GDGs than the maximum, the extra ones are archived. If there are fewer than the maximum, none are archived. When this field is defined with a blank, other Management Class attributes will be used to process the data set.
- Rolled-off GDS Action (MC) This optional field can be defined MIGRATE, EXPIRE, or blank. The default is blank. When this field is defined MIGRATE, CA Disk makes the data set eligible to archive unless the expire attributes in the Management Class indicates that the data set should be expired in which case the data set will be deleted. When this field is defined EXPIRE, CA Disk deletes the rolled-off data set unless the explicit expiration date in the format-1 DSCB is not met. When this field is defined with a blank, CA Disk makes the data set not eligible to archive and will expire the data set based on the Management Class attributes.
- Primary Days Non-usage (MC) This optional field can be defined 0 to 9999 or blank. The default is 60. When this field is defined with a number, those number of days must pass since the last used date before the data set is eligible for archive. When this field is defined with a blank, other Management Class attributes will be used to process the data set.
- Level 1 Days Non-usage (MC) This optional field can be defined 0 to 9999, NOLIMIT, or blank. The default is 60.

Note: When this field is defined 0 and archive to disk is being performed, data sets with this attribute are bypassed.

- Command or Auto-migrate (MC) This optional field can be defined BOTH, COMMAND, or NONE. The default is BOTH. When this field is defined BOTH, CA Disk can archive the data set both implicitly and explicitly. When this field is defined COMMAND, CA Disk can archive the data set only explicitly. When this field is defined NONE, CA Disk cannot archive the data set.

This example:

```
SCAN REALVOLS
  SELECT SGNAME=/
  ARCHIVE
```

accomplishes the following:

- Select the volumes contained in all Storage Groups.
- Select the data sets meeting the appropriate Management Class and Storage Group attributes.
- Archive the data sets.

ARCHIVE Command Examples

This section contains examples of the ARCHIVE command.

ARCHIVE Command Example 1

Archive the unneeded data set LABS.EDP SOURCE for 60 days.

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
FIND DSN=LABS.EDP SOURCE
  ARCHIVE RETPD=60
```

ARCHIVE Command Example 2

Archive for 7 years, allowing auto-restore, non-VSAM data sets and VSAM clusters not used in 28 days, except for SYS data sets and DB2 database components.

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
PASSWORD MASTER=the-CA Disk-master-password
SCAN REALVOLS
  SELECT CRI=(USEDT,LE,TODAY-28)
  EXCLUDE DSNAME=SYS/
  ARCHIVE DISP=RECAT,RETPD=2557
```

not used in 28 days
leave my SYS/ on disk

ARCHIVE Command Example 3

Archive, allowing auto-restore, generation data group data sets, except leave the two most recent generations on disk. Have these data sets under *catalog control*, which means that they are to be expired from the CA Disk archives when they roll off of the catalog (see the IDCAMS DEFINE GENERATIONDATAGROUP command LIMIT parameter).

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SCAN REALVOLS  
  SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-2)  
  ARCHIVE DISP=RECAT,EXPDT=99000 catalog control
```

ARCHIVE Command Example 4

Archive for 7 years, allowing auto-restore, non-VSAM data sets and VSAM clusters not used in 28 days, except for SYS data sets.

Also archive, allowing auto-restore, generation data group data sets, except leave the two most recent generations of most GDGs on disk, and leave the five most recent generations of the rest. Have these data sets under "catalog control", which means that they are to be expired from the CA Disk archives when they roll off of the catalog (see the IDCAMS DEFINE GENERATIONDATAGROUP command LIMIT parameter).

```
//jobname JOB (acct,info),etc.  
// EXEC DMS  
PASSWORD MASTER=the-DMS-master-password  
SCAN REALVOLS  
  EXCLUDE DSNAMES=(data.set.name.patterns.that.should,  
    have.more.than.two.generations.left.on.disk),  
  SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-2)  
    ARCHIVE DISP=RECAT,EXPDT=99000  
  SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-5)  
    ARCHIVE DISP=RECAT,EXPDT=99000  
  SELECT CRI=(USEDT,LE,TODAY-28)  
  EXCLUDE DSNAMES=SYS/  
    ARCHIVE DISP=RECAT,RETPD=2557
```

DELETE Command

The DELETE command instructs CA Disk to delete the non-VSAM data sets and VSAM clusters that satisfy the selection criteria. No backup copy of the data set is taken.

Note: Use this command with care. If used improperly, unrecoverable DATA LOSS can occur.

Always run this command in SIMULATE mode first. Running in SIMULATE mode determines exactly which data sets will be deleted. Keep in mind that because CA Disk does not make a backup copy of the data set during delete, you cannot recover the data. By default, CA Disk does not allow this command. To use this command, specify sysparm DELETEDOK with a value of Y.

Even with DELETEDOK, this function does not delete an ICF or VSAM catalog. Also you cannot delete data sets beginning with SYS1 unless you set sysparm SCRASYS1 with a value of Y.

DELETE Command Syntax

DELETE DISP=

DELETE DISP= is the catalog action for each data set processed.

UNCAT

Default value. This value causes CA Disk to uncatalog the data set when it is deleted.

RECAT

To recatalog to the CA Disk pseudo-volser after deletion, specify this value. RECAT makes the data set a candidate for auto-restore through the CA Disk Catalog Management hook. Back up the data set before starting to auto-restore the data set successfully.

Important! This disposition parameter with the RECATPSUr system parameter, catalog to the pseudo-volser an uncataloged version of the selected data set. The recommended setting, unless you specifically know you want the uncataloged version, is RECATPSUn. RECATPSUr limits only cataloged versions that are selected for cataloging to the pseudo-volser.

NOCAT

To take no catalog action against the data set, specify NOCAT. CA Disk deletes the data set from DASD and leaves it cataloged to the original volser. If SMS-managed data sets are deleted with NOCAT, CA Disk uses the value UNCAT.

SMS and DELETE

CA Disk deletes a data set based on its Storage Group attributes. CA Disk supports the SMS field as follows:

- Auto-migrate (SG) Define this required field as YES, NO, INTERVAL, or PRIMARY.
 - YES, INTERVAL, or PRIMARY means that you can delete data sets.
 - NO, means thatCA Disk you cannot delete any of the Storage Group data sets.

DELETE Command Example 1

Delete (in simulate mode) the unneeded cataloged data set LABS.EDP. SOURCE.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SET MODE=SIMULATE  
FIND DSN=LABS.EDP.SOURCE  
DELETE
```

DELETE Command Example 2

Delete (in simulate mode) system temporary data sets and any incorrectly cataloged non-VSAM data sets that are at least two days old.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SET MODE=SIMULATE  
SCAN REALVOLS  
SELECT CRITERIA=(TEMP,EQ,YES,AND,CREDIT,LE,TODAY-2)  
SELECT CRITERIA=(CATSTAT,NE,YES,AND,CREDIT,LE,TODAY-2)  
DELETE
```

DEL3435 Command

CRI=(VSAMORG,EQ,BROKEN)

CA Disk can detect orphaned VSAM data sets or components with this command. The DEL3435 command lets you delete orphaned (broken) VSAM data sets.

Note: Use this command with care. If used improperly, unrecoverable DATA LOSS can occur.

The DEL3435 command tells CA Disk to delete VSAM orphaned VVRs that satisfy the selection criteria. The delete is done without taking a backup copy of the data set. DEL3435 must precede a SCAN REALVOLS command to select orphaned VVRs. The delete message identifies the VVRs in MSGPRINT by the following message:

ADSST203 3435 CLUSTER NAME COULD NOT BE DETERMINED FOR VSAM COMPONENT
dsn ON VOLUME volser

Use this command with care. Always run it in SIMULATE mode first to determine exactly which VVRs are deleted. CA Disk does not make a backup copy of the data set during DEL3435 delete processing. Recovering the data set, if necessary, can be in serious jeopardy.

DEL3435 Command Syntax

DEL3435

DEL3435 Command Examples

Delete (in simulate mode) orphaned VVR for LABJ1.MJB.T202A.AIX.DATA on volume SMS802

```
//JOBNAME JOB (ACCT,INFO),ETC.  
//STEP EXEC DMS  
SET MODE=SIMULATE  
SCAN REALVOLS  
  SELECT VOL=(SMS802),DSN=LABJ1.MJB.T202A.AIX.DATA  
  DEL3435
```

Delete orphaned VVRs on PRODxx volumes

```
//JOBNAME JOB (ACCT,INFO),ETC.  
//STEP EXEC DMS  
SCAN REALVOLS  
  SELECT VOL=(PROD/),CRITERIA=(DSORG,EQ,VSAM)  
  DEL3435
```

Troubleshooting DEL3435

Attempting to delete orphaned VSAM components can result in error messages like the following:

ADSST203 2546 CATALOG RETURN CODE IS 8 - REASON CODE IS 42

ADSST203 NO RECORD FOUND IN AN ICF CATALOG

ADSST203 3435 CLUSTER NAME COULD NOT BE DETERMINED FOR VSAM COMPONENT

component.name ON VOLUME volser

Reason:

The component is probably cataloged in a catalog not in the normal catalog search path. Using a STEPCAT when the VSAM file was created can cause this problem.

Action:

To determine the catalog that the component entry is in, run a PRINT VVDS for each volume that the stranded components are on.

```
//PRINT EXEC PGM=IDCAMS
//VVDS1 DD DSN=SYS1.VVDS.Vvolser,DISP=SHR,AMP=AMORG,UNIT=3390,
// VOL=SER=volser
//SYSIN DD *
PRINT INFILE(VVDS1) SKIP(000)
/*
```

To delete the VVRs that are not cataloged in the catalog indicated by the alias:

1. Prepare a job to delete the entries in a particular catalog.
2. Add a STEPCAT statement to the job:

```
//STEPCAT DD DSN=catalog.name,DISP=SHR
```

EXPIRE Command

The EXPIRE command instructs CA Disk to expire non-VSAM data sets and VSAM clusters. SMS data sets are expired based on DFSMS Management Class attributes if no format-1 EXPDT exists. Non-SMS non-VSAM data sets are expired based on the format-1 EXPDT. The catalog entry expiration date is used for VSAM clusters. The data set expiration date is compared against the current date, or the date specified in the DATE= parameter, to determine if a data set is expired. If a data set satisfies DSCL selection and is determined to be expired by CA Disk, it is deleted from DASD.

EXPIRE Command Syntax

The syntax of the EXPIRE command is as follows:

EXPIRE DATE=

The EXPIRE command parameter is DATE=. This optional parameter specifies the date upon which to base expire processing. Specify a date constant in any acceptable CA Disk date format. For a list of valid dates, see the section Date Formats in the chapter "General Information." For non-SMS data sets, the date specified in this parameter is compared against the data set expiration date. This comparison is used to determine if a data set should be expired. This parameter is useful in expiring data sets with expiration dates that have not yet been reached, by specifying a date greater than or equal to the not yet reached data set expiration date.

SMS and EXPIRE

CA Disk expires a data set based on its Management Class and Storage Group attributes. In the Management class, the *Expire after Days Non-usage* field and the *Expire after Date/Days* field work together. If both fields are numeric, both must be met for the data set to be deleted. If one field has a numeric value and the other is ', the data set will be deleted based only on the numeric value. If both fields are ', the data set will be kept indefinitely.

In the Storage Group, the *Auto Migrate* field can be defined as YES, NO, INTERVAL or PRIMARY. When this field is defined as YES, INTERVAL or PRIMARY, data sets will be eligible for expiration. When this field is defined NO, CA Disk will not allow the expiration of any of the data sets for the Storage Group.

For more information regarding the Management Class attributes or DFSMS definition in general, see the *DFSMS Storage Administration Guide*.

The SMS fields that CA Disk supports are as follows:

- **Expire After Days Non-usage (MC)** - This optional field can be defined 1 to 9999, or NOLIMIT. The default is NOLIMIT. When this field is defined with a number, the data set becomes eligible for expiration that number of days after the last use date. When this field is defined NOLIMIT, the data set can be kept indefinitely.
- **Expire After Date/Days (MC)** - This optional field can be defined 1 to 9999, yyyy/mm/dd, or NOLIMIT. The default is NOLIMIT. When this field is defined with a number, the data set is eligible for expiration that number of days after the creation date. When this field is defined with a date, the data set becomes eligible for expiration on that date. When this field is defined NOLIMIT, the data set can be kept indefinitely.
- **Rolled-off GDS Action (MC)** - This optional field can be defined MIGRATE, EXPIRE, or blank. The default is blank. When this field is defined MIGRATE, CA Disk deletes the rolled-off data set unless the expiration attributes have not expired. When this field is defined EXPIRE, CA Disk deletes the rolled-off data set unless the explicit expiration date in the format-1 DSCB is not met. When this field is defined with a blank, CA Disk deletes the rolled-off data set unless expiration attributes have not expired.

EXPIRE Command Syntax

The EXPIRE command syntax is:

EXPIRE DATE=

The EXPIRE command parameter is DATE=. This optional parameter indicates the date upon which to base expire processing. Specify a date constant with any acceptable CA Disk date format. For a list of valid dates, see the section Date Formats in the chapter "General Information."

This example:

```
SCAN REALVOLS
  SELECT SGNAME=/
  EXPIRE
```

accomplishes the following:

1. Select the volumes contained in all Storage Groups.
2. Select the data sets meeting the appropriate. Management Class attributes.
3. Expire the data sets.

EXPIRE Command Example 1

Expire (in simulate mode) the expired data set LABS.KSV.SOURCE.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SET MODE=SIMULATE  
FIND DSN=LABS.KSV.SOURCE  
EXPIRE
```

EXPIRE Command Example 2

Expire (in simulate mode) all data sets whose expiration dates are less than today's date and have not been used in one year, which reside on the DMSK01 volume.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SET MODE=SIMULATE  
SCAN REALVOLS  
SELECT VOL=DMSK01,CRITERIA=(EXPDT,LT,TODAY,AND,USEDT,LE,TODAY-365)  
EXPIRE
```

EXPIRE Command Example 3

Expire (in simulate mode) all VSAM clusters with data set names beginning with LABS.KSV, having expiration dates of January 1, 1999 or before.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SET MODE=SIMULATE  
SCAN CAT=/  
SELECT DSN=LABS.KSV./,CRITERIA=(DSORG,EQ,VSAM)  
EXPIRE DATE=1999.001
```

Sample Backup/Archive/Delete Report

The Sample Backup/Archive/Delete report provides the user with a wide range of information for each data set. The following illustration represents a sample BACKUP/ARCHIVE/DELETE report.

DATA SET NAME / NEWNAME	VOLUME	DEVTYPE	DSORG	RECFM	BLKSZ	LRECL	OPTCD	KEVL	ARC	SCR	CAT	REASON	BYTES	TRACKS
***** THIS IS A SIMULATION RUN														
NDMLCS1.TY	SSL003	3300	AM	KSDS	4,096	1,930	00	0	A	N	N	BACKUP	92.73K	2
NDMLCS1.TCQ	SSL003	3300	AM	RRDS	1,536	1,529	00	0	A	N	N	BACKUP	231.82K	5
NDMLCS1.TCX	SSL003	3300	AM	RRDS	1,024	1,014	00	0	A	N	N	BACKUP	46.36K	1
HSMHJB.SYSS.HSMPOX	SSL004	3300	PS	UB	22,532	1,024	00	0	A	N	N	BACKUP	2.08M	46
HSMHJB.SYSS.HSMPOY	SSL004	3300	PS	UB	22,532	1,024	00	0	A	N	N	BACKUP	2.08M	63
SYS4.SIUC.DFHSM.HSMPOX	SYCSY4	3300	PS	UB	4,100	1,024	00	0	A	N	N	BACKUP	95.08M	2,100
SYS4.SIUC.DFHSM.HSMPOY	SYCSY4	3300	PS	UB	4,100	1,024	00	0	A	N	N	BACKUP	99.83M	2,205
SMPEDS.MUS6.JIMM201.F2	SY8CAT	3390	PO	UB	8,000	1,024	00	0	A	N	N	BACKUP	3.46M	64
SYS1.AISTASN1	SY8DLB	3390	PO	FB	8,192	1,024	00	0	A	N	N	BACKUP	498.02K	9
SYS1.SISTASN1	SY8RSA	3390	PO	FB	1,024	1,024	00	0	A	N	N	BACKUP	719.37K	13
SMPEDS.MUS6.HUT4401.F5	SY8SNP	3390	PO	FB	8,192	1,024	00	0	A	N	N	BACKUP	2.11M	39
SYS4.FILEID.R80.FASMP.RFMTDEF	SY9CAT	3300	PO	UB	27,998	1,576	00	0	A	N	N	BACKUP	46.36K	1
SYS4.SIUC.DFHSM.HSMPOX	SY9PG1	3300	PS	UB	4,100	1,024	00	0	A	N	N	BACKUP	13.58M	300
SYS4.SIUC.DFHSM.HSMPOY	SY9PG1	3300	PS	UB	4,100	1,024	00	0	A	N	N	BACKUP	13.58M	300
SYS3.SA155.DS1KPNL	SY9PG1	3300	AM	KSDS	7,168	1,024	00	9	A	N	N	BACKUP	741.81K	16
SYS1.SISTASN1	SY9RSA	3390	PO	FB	1,024	1,024	00	0	A	N	N	BACKUP	719.37K	13
SYS1.SISTASN1	SY9RSB	3390	PO	FB	1,024	1,024	00	0	A	N	N	BACKUP	719.37K	13
SYS4.SIUC.DFHSM.OCD5	SY9SY1	3300	AM	KSDS	12,288	1,000	00	44	A	N	N	BACKUP	34.09M	753
SYS4.DM11.TB	SY9SY1	3300	AM	KSDS	4,096	1,024	00	128	A	N	N	BACKUP	1.40M	31

The column headings displayed in the previous figure are defined as follows:

dataset NAME / NEWNAME

DSNAME of the data set selected for processing. If the NEWNAME parameter is specified on the command, its new name would be displayed on a new line. For additional information, see NEWNAME=.

VOLUME

Volume on which the selected data set was found.

DEVTYPE

Device type in which the volume selected for processing resides. For a list of possible values, see the section Specifications in the chapter "MOVE/COPY."

DSORG

Data set organization of the data set. For a list of possible values, see the section DSORG – Data Set Organization in the chapter "Basic System – DMS PROC"

RECFM

Record format of the data set. For a list of possible values, see the section RECFM – Record Format in the chapter "Basic System – DMS PROC."

BLKSZ

For non-VSAM, the DCB block size of the data set. For VSAM, it is the CISIZE of the cluster.

LRECL

For non-VSAM, the DCB logical record length of the data set. For VSAM, it is the logical record size from the clusters data component.

OPTCD

This is a 1 byte character field at offset Hex 55 of the Format-1 DSCB. They describe certain characteristics of the data set depending on its DSORG. These bit settings have the following meanings:

KEYL

This is the length of the key in the data portion of the cluster, as specified by the user at DEFINE time.

ARC

Indicates whether or not the data set was written to tape media. Possible values in this column are:

A

Data set was written to tape media (archived).

N

Data set was not written to tape (not archived).

Data set was in use or an error occurred at the time it was selected for processing.

SCR

Indicates whether or not the data set was removed from DASD. Possible values in this column are:

N

Data set was not removed from DASD (no scratch).

S

Data set was removed (scratched).

Data set was in use or an error occurred at the time it was selected for processing.

CAT

Indicates what (if any) catalog action was taken against the data set. Possible values in this column are:

N

The catalog entry was not changed (no catalog action).

R

The volume associated with the catalog entry was changed to the CA Disk pseudo-volser (recatalog).

U

The catalog entry was removed from the catalog (uncatalog).

Data set was in use or an error occurred at the time it was selected for processing.

REASON

Indicates which DSCL action command processed the data set. Possible values in this column are:

ARCHIVE

This data set was processed by ARCHIVE.

BACKUP

This data set was processed by BACKUP.

DELETE

This data set was processed by DELETE.

EXPIRE

This data set was processed by EXPIRE.

Data set was in-use at the time it was selected for processing

KBYTES

Indicates the size of the data set in kilobyte format.

TRACKS

Indicates the size of the data set in track format.

DSORG	Binary	Hex	Meaning
BDAM	1000 0000	80	Write Validity check

DSORG	Binary	Hex	Meaning
	0100 0000	40	Track Overflow
	0010 0000	20	Extended Search
	0001 0000	10	Feedback
	0000 1000	08	Actual Addressing
	0000 0100	04	Dynamic buffering
	0000 0010	02	Read exclusive
	0000 0001	01	Relative Block Addressing
ISAM	1000 0000	80	Write Validity check
	0100 0000	40	Accumulate track index ent.
	0010 0000	20	Master Indices
	0001 0000	10	Independent overflow area
	0000 1000	08	Cylinder Overflow area
	0000 0100	04	Reserved
	0000 0010	02	Delete Option
	0000 0001	01	Reorganization criteria
BPAM	1000 0000	80	Write Validity check
BSAM	0100 0000	40	Allow data check (DASD printer)
QSAM	0010 0000	20	Chained scheduling
	0001 0000	10	DOS/OS interchange feature
	0000 1000	08	Treat EOF as EOVS (tape)
	0000 0100	04	Search direct
	0000 0010	02	User label totaling
	0000 0001	01	Each record contains a table reference character
VSAM	1000 0000	80	Data set is cataloged in an ICF catalog
	0100 0000	40	Data set is an ICF catalog
	0011 1111	—	Reserved

Explicit ARCHIVE

The CA Disk archive function allows the user to archive (and subsequently to restore) data sets of any of the following organization types:

AM

VSAM

DA

Direct Access

IS

Indexed Sequential

PO

Partitioned Organization

PS

Physical Sequential

Device Independence and Ease of Use

Data sets of all organization types can be relocated to both new device types and new track locations. This includes VSAM and ISAM data sets, traditionally regarded as unmovable. All that is needed is the data set name and volume. CA Disk, maintains all data set attributes, space requirements, device conversion calculations, and location of data sets on the archive volumes. This permits a single fixed set of JCL to be used for all archives (and restores), without need of looking up any of the previously mentioned items, or modifying any of the JCL (typically DD statements for needed volumes).

Data set Compression And Reorganization

When a data set is being read from disk, CA Disk processes the data blocks within it *sequentially* as dictated by the organization type. Thus only active and used data blocks are read. This results in PDS, ISAM and VSAM data sets being copied to the archive tape in compressed format. Any subsequent restore or volume recovery results not only in the consolidation of several extents into one (defragmentation), but also in the compression/reorganization of the data within the single extent (eliminating the additional need to compress and reorganize these same data sets).

PDS Member Lists

A subset of a partitioned data set can be designated by specifying a list of one or more of its members or member alias names. Associated member and alias names are always maintained, regardless of which are specified in the list. It should be noted that standard default processing at successful conclusion of the archive is to uncatalog and delete the PDS, and to index the archived members under the PDS name. Overrides to the defaults can be desirable in such a case.

PDS Anomalies

CA Disk does extensive testing for errors and inconsistencies in partitioned data sets, and provides installation options for processing them even when logical errors have been detected. Standard default processing is to issue diagnostic messages for the errors and then bypass them. Consult with your local CA Disk installation and support staff to determine what processing options are in effect if any of your partitioned data sets are diagnosed to have anomalies.

Deferred Archives (Queued Archive Requests)

Requests for data set archival can be deferred over a period of time until a batch DISPOSE run can be performed to process all requests deferred since the previous batch run. Use of this facility is desirable, in that a single archive tape mount is required instead of individual mounts for each data set.

A deferred request is made by using the DARCHIVE command name, instead of ARCHIVE, or by adding the DEFER parameter to the ARCHIVE command.

Deferred archive requests are queued and kept in the ARCHCMDS subfile of the CA Disk FILES. After a queued request has been processed, its completion status is maintained for a default period of five days, after which time it is deleted automatically during the next DISPOSE run. Through the ISPF panels, the saved status can also be displayed online. Otherwise, the output from the DISPOSE run must be inspected to determine the outcome of processing for the request. Sysparm DARCKEEP can be used to alter the default period for which the status is maintained in the ARCHCMDS file.

Default Delete, Catalog, and Change Bit Resetting Actions

The ARCHIVE command defaults to deleting and uncataloging the named data set. The *change bit* in the VTOC entry for the data set is also turned off by default if the data set is not deleted. All three of these default actions can be changed by the use of sysparms EXPASCOP, ARCATOPT and EXPARSOP. Consult your local CA Disk installation and support personnel to determine what defaults are being used.

Each default can be overridden by specifying the appropriate parameter on the ARCHIVE command.

Archive Report Sequencing Options

The archive report can be produced in three sequences:

1. Data set name sequence
2. Data set name sequence within volume
3. Exact processing sequence (unsorted)

By default, all three reports are produced. See sysparm ARCHSORTudv in the *Systems Guide* if you want to change the defaults. Any combination of the three report sequences can be specified.

Explicit ARCHIVE Condition Codes

The following return codes are set at the completion of the ARCHIVE job step:

0

All ARCHIVE/DARCHIVE requests have completed successfully.

1

All ARCHIVE/DARCHIVE requests have completed successfully. Produce sorted reports.

4

An error occurred during the archive of one or more data sets.

5

An error occurred during the archive of one or more data sets. Produce sorted reports.

16

One or more of the input commands contained an error and was not processed.

17

One or more of the input commands contained an error and was not processed. Produce sorted reports.

Explicit ARCHIVE Restrictions

The following paragraphs detail the Explicit ARCHIVE Restrictions.

ISAM

Present support is restricted to the archival of data sets contained on a single volume. Separate index, prime and overflow areas can exist but they must be contained on the same volume. During restore processing, the index and overflow areas are suballocated from prime (that is, not separate) unless the data set is preallocated.

Unmovable

SYSCTLG Archival

An OS CVOL catalog can be archived under any name desired, but it must be given the name of SYSCTLG when it is restored. Connect pointers or alias entry pointers from the master catalog to the CVOL catalog are not altered. If a CVOL catalog is to be moved to a new volume, the user must ensure that the appropriate connects are done in order to use the moved catalog. Due to the embedded TTR pointers in a SYSCTLG, it must be restored to a volume of the same device type (all variations of the 3380 are considered the same type).

Deferred Archive

A subset of a PDS (specified by a member list) is not supported.

Multiple Archive Requests and Deferred Archive Requests

Requests to archive one or more non-VSAM data sets or VSAM clusters immediately or to have them placed into a queue to be processed at a later time (deferred requests) are submitted with the following JCL. The addition of a STEPCAT DD statement can be needed for processing incorrectly cataloged VSAM clusters.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC ARCHIVE  
MODESET, DARCHIVE,  
        AND/OR  
ARCHIVE COMMANDS ENTERED HERE  
MODESET SIMULATE  
DARCHIVE  
        SIMULATE,DSNAME=,VOLUME=,NEWNAME=,RETPD=,EXPDT=,RECATALOG,  
        UNCATALOG,NOCATALOG,SCRATCH,NOSCRATCH,RESET,NORESET,PASSWORD=  
        (PASSWORD= not applicable to VSAM)  
ARCHIVE
```

same parameters as above with the addition of MEMBER= (not applicable to VSAM),DEFER

Explicit ARCHIVE MODESET Command Parameters

The following is a description of the commands and parameters used to invoke the explicit archive function.

The MODESET command has been provided to allow explicit archival to be run in a simulated mode and to provide the option of turning off Storage Administration FACILITY Class checking. If used, it must be the first command in the SYSIN input stream and will apply to all the following DARCHIVE or ARCHIVE commands.

The MODESET command syntax is:

MODESET SIMULATE, NOADMIN

Inclusion of the SIMULATE parameter will allow all explicit archive functions to be tested except for actual data set disposition and copy. The effect of the archive processing will be reflected in the CA Disk report, which is automatically generated.

Inclusion of the NOADMIN parameter will prevent all explicit archive functions from making use of the Storage Administration FACILITY Class Profiles. All functions performed in this step will only make use of the standard security access for the user submitting the job.

No tape allocation is required in the simulate mode.

Explicit ARCHIVE and DARCHIVE Syntax

Explicit ARCHIVE and DARCHIVE command syntax is:

DARCHIVE

SIMULATE, DSNAME=, VOLUME=, NEWNAME=, RETPD=, EXPDT=, RECATALOG,
UNCATALOG, NOCATALOG,
SCRATCH, NOSCRATCH, RESET, NORESET, PASSWORD=

ARCHIVE

Same parameters as above with the addition of MEMBER=, DEFER

SIMULATE

This simple parameter can be placed on each DARCHIVE or ARCHIVE command in lieu of using the MODESET SIMULATE command. For a given input stream, simulate and non-simulate modes should not be intermixed.

DSNAME=

The full explicit name of the non-VSAM data set or VSAM cluster to be archived (relative GDG version number is permitted, but if multiple generations are processed in the same job step, they must be specified in the descending order of their values, that is, -2, -1, 0). The catalog is searched to find VSAM clusters. The catalog is also searched to find non-VSAM data sets unless you also supply the volume parameter.

VOLUME=

This optional parameter specifies the volume serial number of the device containing the data set to be archived. If the data set is not found on this device, the request is rejected.

NEWNAME=

An alternative data set name to be assigned to the archived data set (a relative GDG name is also permitted). The archived version is indexed under this name, which can be up to 44 characters in length.

RETPD=

A one- to five-digit number to specify a retention period to be assigned to data sets. If this parameter is omitted, a default value is obtained from sysparm ARCRETPD. For more information, see the section Year 2000 Considerations in the chapter "General Information."

EXPDT=

An expiration date in an accepted CA Disk format can alternatively be specified to designate the period of time the data set will remain archived. (If the value given is less than the current date, the current date will be used as the EXPDT.) For more information, see the section Year 2000 Considerations in the chapter "General Information."

MEMBER=

To archive a subset of a partitioned data set, this parameter is used to specify one or more member and/or group names to constitute the archived version of the data set. (A maximum of 25 entries is supported.)

This parameter is valid only when an immediate archive is being done (not a deferred archive).

DEFER

This parameter can be specified on the ARCHIVE command to indicate that the request is to be queued for later processing. It can be used in lieu of changing the ARCHIVE command to DARCHIVE.

PASSWORD=

The password parameter provides the password value required to access a protected VSAM cluster. If the cluster is to be archived but not deleted, the read-level or higher access is required. If the cluster is to be deleted, the master-level password is required.

Catalog Actions

Only one of the following three catalog parameters (RECATALOG, UNCATALOG, or NOCATALOG) should be specified on any given command. However, if multiple parameters are entered, RECATALOG supersedes UNCATALOG and UNCATALOG supersedes NOCATALOG

RECATALOG causes the data set to be recataloged to the pseudo-volume after it has been archived.

The default value is UNCATALOG. The presence of this parameter causes non-VSAM data sets to be uncataloged after they have been archived successfully.

For VSAM clusters, this UNCATALOG parameter has no effect, use the SCRATCH parameter instead.

NOCATALOG causes non-VSAM data sets to be left cataloged after they have been archived successfully. If you specify this NOCATALOG parameter, you probably want to specify NOSCRATCH as well.

For VSAM clusters, this NOCATALOG parameter has no effect, use the NOSCRATCH parameter instead.

Delete Actions

Only one of the scratch parameters should be specified on any given command. However, if both are entered, SCRATCH supersedes NOSCRATCH

The default value is SCRATCH. This parameter can be specified to cause the data set to be deleted (and for VSAM, also uncataloged) after it is archived successfully.

This parameter can be specified to prevent the deleting (and for VSAM, any catalog action) of the data set after it is archived successfully.

Reset Change Bit Actions

Only one of the reset parameters should be specified on any given command. However, if both are entered RESET supersedes NORESET

The default value is RESET. This parameter will cause the *change bit* to be reset (to zero, or off) for any data set that is archived successfully.

If a data set is backed up, normal processing will reset (to zero) the change bit DS1IND02 in the format-1 DSCB. Specification of this parameter will cause processing to bypass updating the change bit.

Processing Queued ARCHIVE Requests

To dispose of the deferred archive requests (that is, actually do the archiving) the JCL for immediate archival is used, with the exception that the module executed is ADSDM276. The following JCL will process (dispose of) all deferred archive requests placed in the ARCHCMDS queue:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC ARCHIVE,PARM.ARCHIVE=ADSDM276
```

Listing of Deferred Archive Request (LISTREQ)

The status of the deferred archive requests can be listed using the following JCL:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC LISTREQ  
LISTREQ DSN=,ARCHIVE,ALL
```

DSN= is a list of one or more data set names or prefixes for which deferred requests are to be deleted. An acceptable abbreviation for this keyword is DSN. A maximum of 20 data set names or prefixes are allowed.

DSN=/ results in all DARCHIVE requests being deleted.

The ARCHIVE parameter results in queued DARCHIVE request records being selected. It can be abbreviated as ARC.

This parameter **MUST** be specified to list deferred archive requests. The default value for LISTREQ will display deferred restore requests.

The ALL parameter results in all the DARCHIVE request records being selected.

Erasing of Deferred ARCHIVE Requests

The DERASE command can be used with the following JCL to delete requests from the DARCHIVE queue:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DERASE  
DERASE DSN=,ARCHIVE
```

DSN= is a list of one or more data set names or prefixes for which deferred requests are to be deleted. An acceptable abbreviation for this keyword is DSN. A maximum of 20 data set names or prefixes are allowed.

DSN=/ results in all DARCHIVE requests being deleted.

The ARCHIVE parameter indicates that the deferred requests to be deleted are archive transactions. It can be abbreviated ARC.

Chapter 7: RESTORE/RECOVER

This section contains the following topics:

[About Restoring and Recovering](#) (see page 301)

[General Information](#) (see page 302)

[Restore and Recover of VSAM Clusters](#) (see page 311)

[RESTORE Command Syntax](#) (see page 327)

[Implicit Recovery and Volume Defragmentation](#) (see page 346)

[RECOVER Command](#) (see page 348)

[Volume Defragmentation](#) (see page 358)

[Sample Reports](#) (see page 358)

[SMS and RESTORE/RECOVER](#) (see page 360)

About Restoring and Recovering

The CA Disk restore function allows you restore DA, PO, POE, PO2, PS, PSE, and ISAM data sets and OS catalogs without the need for preallocation. You can also restore VSAM clusters and ICF catalogs. In most cases, you can even restore to a device type other than that from which the data set was backed up or archived.

- Physical Sequential (PS)
- Direct Access (DA)
- Indexed Sequential (IS)
- Partitioned Organization (PO)
- Access Method—VSAM (AM)

An individual data set or cluster is restored via the RESTORE command. A group of data sets and clusters, an entire volume, or a group of volumes, however, is restored via the RECOVER command.

Commands Available	JCL Proc Executed
DERASE	DERASE
LISTREQ	LISTREQ
RECOVER	FMS
RESTORE	RESTORE
(dispose)	RESTORE with overridden EXEC parm

General Information

The restore function begins by reading in all of the restore commands and sequencing them by file number within each archive tape (or archive data set on disk), the optimum performance arrangement. All data sets being restored from the archives on disk are done before those archived to tape. Two different variations of the restore function are available. The first form instructs CA Disk to go immediately to the index of archived data sets, find the named data set and proceed with the restore. The second form instructs CA Disk to begin by doing a catalog locate for the named data set. If the data set is cataloged to the CA Disk pseudo-volume (default is ARCIVE), the same process as for the first form is continued. Otherwise a return code is set and the next restore request is processed. Coding the SYSCTLG parameter on the command statement indicates this second form of the RESTORE command. The forms can be intermixed in the request stream.

Using the Bestfit facility for RESTORE/RECOVER of SMS Data sets

There are occasions where the DASD environment for a Restore/Recover has become so fragmented that there is not enough space to satisfy the request for storage because z/OS is unable to acquire the space requested in five extents or less on one volume, even though the space may be available in multiple extents or across volumes. This causes the Restore to fail and requires human intervention to resolve the *space shortage*. Traditionally CA Disk has attempted to work around this by requesting a smaller amount of storage, either the original primary amount or the sum of the first three extents, and relying on secondary extents to fill the amount needed. However this sometimes failed due to various other environmental causes. CA Disk now uses a facility provided by z/OS that was introduced called Space Constraint Relief. This allows CA Disk to request that z/OS allocate the data set and relax some of the constraints that it works under. In particular, z/OS will allocate the data set with more than five extents – up to the maximum extents the data set type supports – and can use multiple volumes to satisfy the request. The Space Constraint Relief is available under CA Disk only for SMS data sets, and the multiple volume support of Space Constraint Relief is available for SMS non-VSAM data sets regardless of number of volumes at the time of Backup/Archive and for SMS VSAM data sets that were contained on multiple volumes at the time of Backup/Archive.

A SYSPARM controls the multiple volume support of the Space Constraint Relief. The SYSPARM is SMSSCRVL, and it is a number from 0 to 59 which specifies the number of volumes that data sets can span if the data set is unable to be allocated using the number of volumes it originally occupied. The Bestfit facility is automatically invoked when normal Dynamic Allocation of a data set being restored, fails. An attempt is made to satisfy the request using the original number of volumes and, if that fails, an attempt is made to satisfy the request using the number of volumes specified by SMSSCRVL if it is greater than the original number of volumes.

Multiple Files Data set for Restore

Multiple FILES are supported for Restore, Deferred Restore, and Recover processing. If you want to make them all available to these functions, you should use the MFILES DD statement instead of the FILES DD statement. Add each data set name to the MFILES concatenation; up to 256 data sets are supported. If using a concatenation of all Files Data Sets (FDS) or a mixture of FDS and Files Database (FDB), then the order of the concatenation determines the order of search for the data set to be restored. The first one that contains an index record for the archived data set will be used, even if another contains a more recent version of the data set. Therefore, it is very important that they be specified an index record for the archived data set will be used, even if another searched as logically joined so the most current version of the data set is found regardless of the concatenation order.

Note: The MFILES DD statement is not supported in Dispose processing. If you have more than one FILES, you must execute DISPOSE separately for each one.

We do not recommend concatenating data sets. If you concatenate data sets, you will get errors.

Restore Condition Codes

The following return codes are set at the completion of the restore job step:

0

All data sets requested to be restored were restored successfully.

2

Disposition processing (requested/default catalog actions) failed for one or more data sets.

16

One or more Attempted restores failed.

Non-VSAM Restore Volume Selection

The philosophy behind the restore allocation technique is to maximize system catalog integrity and reduce the creation of disconnected data sets. Therefore, the sequence for selecting a target volume for the restored data set is as follows:

1. VOL= parameter specified.

Allocation uses the specified volume. If the allocation fails to the specified volume, the restore is terminated. (Mutually exclusive of POOL parameter).

2. POOL= parameter specified.

A pool name from the specified parmlib member DASDPOOL is used. If allocation fails for all volumes within the specified pool, the restore is terminated. (Mutually exclusive of VOL parameter).

3. No directional parameters specified. (VOL= parameter or POOL= parameter omitted).

Allocation is directed to the original source volume. If allocation to the original source volume fails, an allocation attempt is made to the default pool. If a default pool is not specified and the allocation fails, the restore is terminated.

Note: See sysparms RESRCAT, RESCHCAT, and RECCHCAT, in the *Systems Guide*. Under certain conditions, the allocation results can vary.

Multi-Volume Restore

A multi-volume data set can be restored three different ways:

1. Sysparm PRIALLOC set to 'Y'.

The multi-volume data set will be allocated to the original volumes from which it was archived. If allocation fails, the restore is terminated.

Note: CA Disk pooling is bypassed for multi-volume data sets when PRIALLOC is set to Y. Pooling products (such as CA Allocate) can redirect these data sets.

2. Sysparm PRIALLOC set to 'N'.

This setting consolidates the original total allocation to the new primary allocation amount. The multi-volume data set allocation is directed to a single volume, and if it does not fit on the first volume, the other volumes will be used as overflow volumes. If allocation fails, the restore is terminated. For example, if a 3-volume data set is restored to a single volume, the first volume will contain a format-1 entry and a catalog entry with the 2 remaining volumes represented only by the catalog entry. (All volumes must be online)

3. Sysparm PRIALLOC set to 'N' and VOL= or POOL= parameter specified.

This setting directs the allocation to a volume using the consolidated primary allocation amount. If allocation fails, the restore is terminated. For example, if a 3-volume data set is restored to the specified volume, that volume will contain both a format-1 and a catalog entry.

Deferred Restores (Queued Restore Requests)

Requests for restoring of data sets can be deferred over a period of time until a batch DISPOSE run is made. Use of this facility is desirable in that the number of tape mounts will be minimized. This is a consequence of the sequencing of perhaps several restore requests by archive tape volume serial and file number, and restoring all within a single mount and scan of the needed tapes.

A deferred request is made by adding the DEFER parameter to the RESTORE command (or using DRESTORE instead of RESTORE as the command name).

Deferred restore requests are queued and kept in the RESTCMDS subfile of the CA Disk files data set. After a queued request has been processed, its completion status is maintained for a default period of five days, after which it is automatically deleted during the next *dispose run*. Through the ISPF panels, the saved status can also be displayed online. Otherwise the output from the dispose run must be inspected to determine the outcome of processing for the request. Sysparm DRESKEEP can be used to alter the default period for which the status is maintained in the RESTCMDS file.

Normally, DRESTORE requires that the requestor has sufficient authority to restore the data set, determined by sysparm SECURDRS. However, if the SYSCTLG parameter is specified and sysparm ARSECURE has a value of 'N' or 'R', the requestor will only require INPUT authority. When the DISPOSE job is run, the SYSCTLG parameter will only allow the restore to complete if the data set is cataloged to the pseudo volser, and the USERID associated with DISPOSE has sufficient authority to create or update the data set.

For both TSO and ISPF DRESTORE commands, if the data set is cataloged to the pseudo volser, the SYSCTLG parameter will automatically be generated for the request, allowing these functions to operate exactly like DMSAR.

Restore of SYSCTLGS

Any time an OS CVOL catalog is restored, it must be given the name SYSCTLG in order to do proper formatting. Current support also requires that SYSCTLG be restored to the same device type from which it was archived (3390 and 3390-1 are regarded as the same device type). Connect or alias entry pointers from the master catalog are not altered. If a SYSCTLG is restored to a different volume, the user must ensure that the appropriate disconnects and connects are done to enable its use.

Restore Grace Periods

An installation option exists that assigns a grace period to any restored data set, during which time it is exempt from processing by archive (this process does not affect BACKUP). The grace period applies, as a default, no matter what volume the data set resides on, even if it is moved after restore.

For details, or to change the default processing, see the sysparm descriptions for RESRETPDnnnnn and RETEXCLDn in the *Systems Guide*.

Restore To Preallocated Data Sets

By default, CA Disk will not write over a preallocated data set. This can be changed, however, via sysparms PREALLOC and VSPREDEF. The SCRATCH parameter can also be specified on the RESTORE command, which will cause a preallocated data set to be scratched and then reallocated (providing the DSORG of the target data set matches the DSORG of the data set being restored, otherwise the restore will fail). Sysparm RESRCAT is for non-VSAM data sets only. It is used to determine when and from which volume CA Disk will scratch the preallocated data set. If the sysparm RESRCAT is set to Y (recommended in a DFSMS or pooling environment), and the SCRATCH parameter is specified, CA Disk will scratch the preallocated data set from the cataloged volume, before attempting any allocation. If the sysparm RESRCAT is set to N and the SCRATCH parameter is specified, CA Disk will only scratch the data set if it resides on the same volume to which CA Disk has attempted its allocation. For more information, see the sysparm description for RESRCATn in the *Systems Guide*.

If preallocated data sets are allowed, SCRATCH is not specified, and a partitioned data set is being restored, the ERASE parameter permits an existing PDS to be overlaid by the restored copy. If ERASE is not specified, the members being restored are merged into the PDS at the end of its current member data. However, if some of the members being restored already exist in the PDS, they will not replace the duplicate members unless the REPLACE parameter is specified.

PDS Member Restore

Specifying a member list can restore a subset of a PDS. These members can be restored directly into an existing PDS as described previous, or into a new data set of their own.

Any alias for a given member can be specified for selective restoring. The alias name will be used to restore the true member, and then all alias names associated with it will be updated.

PDS disconnected aliases archived by PDS anomaly processing will not be restored or recovered to pre-allocated PDSEs.

Restore to Unlike Device Types (Block Size Concern)

Most data sets can be restored to device types different from their original ones, as long as the track size of the new device is larger than any of the data set block sizes. Migrations to *larger devices* therefore rarely present any problems. But the reverse often does; for example, moving from 3380s to 3350s. If this is the case, you should include sysparm REBLKBIT with a value of Y. This will allow CA Disk to reblock data sets to larger or smaller devices. You should note, however, that some data sets cannot be reblocked; for example, a RECFM=U data set.

Restore Space Allocation

When a data set is archived, the actual space used is calculated and recorded. At restore time, the space allocated is either the original size of the data set or the used space quantity whichever is larger. For example, a restore will not reduce the primary space quantity, but will adjust it up for data sets that have gone into extents. The primary quantity is adjusted appropriately for restoring to different device types.

Restore of ISAM Data Sets

ISAM data sets that have separate index, prime and overflow will be archived successfully, but Restore will restore them as one prime allocation with the index and overflow areas suballocated in the prime space.

Also, CA Disk lets ISAM allocate routines determine the amount of index space needed. This cannot be sufficient to allow future updates necessary.

To change the index size or other allocation attributes, or to allocate with separate prime and overflow areas, preallocate the ISAM data set and restore into it.

Restore Considerations for GDG Data Sets

When restoring GDG data sets, the following principles will be followed in regards to catalog action:

NEWNAME:

When this parameter is specified, RESTORE treats the target DSNAME as a new data set creation. The target data set will always be cataloged and placed in the sphere, whether the sphere is full or not.

For non-SMS Data Sets without NEWNAME:

- If a catalog entry exists, the entry is left intact.
- If no catalog entry exists, the data set will be cataloged unless doing so would force another entry out of the catalog (that is, the sphere is full).

For SMS Data Sets without NEWNAME:

- If a catalog entry exists and it has a valid status, the status will be left intact.
- If a catalog entry exists but a valid status cannot be detected, or if a catalog entry does not currently exist, the status of the entry will be set to a status as close as possible to what it was at the time of the backup, unless doing so would force another entry out of the catalog. Therefore, the following will apply:
 - Data sets with a status of ACTIVE when backed up will be placed back in ACTIVE status, unless doing so would force another entry out of the catalog.
 - Data sets with a status of DEFERRED or ROLLED-OFF when backed up, but no longer have a valid status or catalog entry when the restore is initiated, will be placed in DEFERRED status whether or not the sphere is full.
 - If the original catalog status cannot be determined it will be placed in ACTIVE status.
 - If the sphere is full (that is, the GDG's limit has been reached), the entry will be placed in DEFERRED status, regardless of what its status was at the time of the backup.

If CA Disk cannot set a data set's catalog entry to the status CA Disk feels is correct, message 3729 will be issued to indicate the status the data set was left in, and the reason for placing the data set in that status.

For more information, see User Exit GDGCATEX —GDG Data set Catalog Action in the *Systems Guide*.

PSE Considerations

Extended sequential data sets are simply multi-volume sequential data sets that are managed by DFSMS. This combination resulted in some special considerations that each user should be aware of.

SYSPARMS

To restore PSE data sets, sysparm SMSALLOC should be set to a value of 'Y', or the DSNNAME must be eligible for PSE allocation.

PS/PSE Conversions

Data set conversions can be accomplished by preallocating the target data set prior to the CA Disk action being performed.

To restore an Archived PSE to a PS, the target PS must be either pre-allocated, or its DFSMS DC must be changed. Specifically, the data set NAME TYPE field must be changed to NON-EXTENDED.

Note: Later phase will allow conversion of an Archived PS to PSE.

Concurrent Copy Considerations

Since DFSMSdss was used to create the DSNINDEX record, DFSMSdss must also be used during Restore allocation. For this reason, all allocation-related RESTORE parameters are not supported.

The following list of RESTORE parameters is the ONLY parameters available when you use Concurrent Copy to create DSNINDEX records:

DSNAME=, NEWNAME=, VERSION=, TIMEDATE=, ERASE, SCRATCH

DFSMSdss Data Mover Interface

This interface provides CA Disk users a way to restore HFS data set organizations backed up by the CA Disk DFSMSdss data mover interface. Those data sets restored using this interface are maintained in the archives as normal ARCHVOL and DSNINDEX records. CA Disk utilities, such as IXMAINT, are used as normal to manage them.

Note: If you attempt to use this interface and DFSMSdss is not active or at a sufficient operating level (MVS/SP 5.22 with DFSMS 1.2 or above), CA Disk will bypass the selected data sets along with an appropriate error message.

With this interface active, CA Disk automatically switches to the DFSMSdss data mover when HFS data set types are detected. CA Disk switches back to the normal CA Disk data mover for all other data set types.

Activation

The DFSMSdss Data Mover interface is controlled by sysparm USEDSSIO. For details, see its sysparm description under USEDSSIO in the *Systems Guide*.

Limitations

Since CA Disk is more flexible than DFSMSdss, not all of the existing CA Disk RESTORE parameters are supported by DFSMSdss. Whereas some of these limitations are not appropriate for HFS data sets anyway, the complete list is as follows:

CREATE	ERASE	MEMBER=	NEWNAME=	NOCATALOG
NOLOAD	PDSECONV	POOL=	RECATALOG	REPLACE

Restore and Recover of VSAM Clusters

VSAM clusters are reorganized as they are restored or recovered (unless they were archived in control interval image copy format). Often, this eliminates multiple extents and CA/CI splits for the clusters involved. Catalog entries are dynamically recreated for each cluster with the attributes, as they existed, when the Archive/Backup was created (unless the cluster was predefined). CA Disk adjusts space allocations for the device type to which the clusters are recovered.

When recovering suballocated (non-unique) VSAM clusters to a "new" volume, you must predefine the VSAM space. After this space is created, CA Disk dynamically defines the VSAM clusters into it. CA Disk automatically defines the self-containing spaces for unique clusters. When restoring suballocated clusters that came from a VSAM catalog into an ICF catalog, the attribute is automatically changed to UNIQUE, regardless of its prior value.

The FILES contains the volume serial of the first data component of a VSAM cluster. During Archive/Backup processing, CA Disk issues a message that shows this volume serial. The volume serial must be correctly specified for a RECOVER run that uses the VOL= parameter. CA Disk can only RECOVER clusters if the first data component volume is correctly specified.

Restrictions in Restore and Recover of VSAM Clusters

Take care when altering the control interval size of any ESDS cluster that is part of an IMS or CICS database. Changing the CI size can alter the way logical records fit in the data set when it is restored, causing internal RBA pointers to become invalid. If the cluster was archived in control interval image copy format, the CI size cannot be changed at restore time. Changing CI size causes a record length error when CA Disk tries to restore the data into the data set.

Volume recovery can restore multi-volume data sets when the volume recovered is the primary volume for the cluster. Otherwise, the cluster is bypassed.

Recover does not process alternate indexes that are archived as physical data sets because of possible synchronization problems with the base cluster. CA Disk redefines and rebuilds the cluster's alternate indexes when the base cluster is restored.

Base clusters and physical alternate indexes cannot be restored in the same job step. The base cluster(s) must be restored first, followed by another job or step to restore the alternate indexes. For more information, see the sysparm description for VSARCAIXn in the *Systems Guide*.

RACF profiles are maintained only for base clusters. Except for alternate indexes that were explicitly archived as physical data sets, if an alternate index is restored that had a RACF profile; the base cluster's profile will be used for the alternate index.

Special Processing Options

The following topics detail special processing options that can be used to tailor the VSAM restore and recover functions. Consult Sysparms and User Exits in the *Systems Guide* for further explanations.

Secondary Extent Consolidation (Defragmentation of VSAM Clusters)

For single volume data sets, CA Disk will automatically change the primary allocation for a DEFINE request based on original allocated extents with sysparm VSSPUSED specified with a value of Y. By specifying this parameter, CA Disk will sum all of the extents that were allocated to the data set at archival time. It will then determine the amount of this space that was actually used, and if this value is larger than the original primary amount, will use the new value as the primary allocation amount. For multi-volume data sets, CA Disk will always use the original allocation amounts. If the data set is being restored to an unlike device type, the allocation amount will be changed to reflect either the increased or decreased track capacity.

In addition to sysparm VSSPUSED, the VSALLOEX user exit is available to allow a user program to change allocation amounts based on information supplied to the exit.

If a new allocation amount is specified on the RESTORE command via the ALLOC=, DALLOC= or IALLOC= parameters, the value specified will be used directly without any conversions.

Default Catalog Processing

Normal CA Disk processing allows catalog management to determine which catalog to define the cluster in, based on the *Order of Catalog Use: Define* outlined in the IBM publication Access Method Services Reference. By specifying sysparm VSDEFCAT with a value of N, CA Disk will define the cluster in the same catalog from which it was archived.

Note: This sysparm has no effect when the catalog is explicitly named in the RESTORE or RECOVER command.

Speed Option

When VSAM clusters are defined via Access Method Services, the normal default value of RECOVERY is used unless SPEED is explicitly requested. CA Disk will automatically define all clusters with the SPEED option to improve restore performance, regardless of how they were originally defined. To use the option defined with the cluster, specify sysparm VSSPEEDO with a value of N.

Changing Data Sets Attributes

Either because of choice or necessity, there are times when the data set being restored needs to have some of its attributes modified. This can be particularly true when restoring to an unlike device type, where the cluster define can fail due to restrictions in buffer space and control interval size. By specifying sysparm VSATTRIB with a value of Y, CA Disk will ignore the original buffer space and control interval size values that were defined with the cluster and will allow Access Method Services to calculate new values for these parameters.

In addition, the user can code an exit named VSATTREX. This exit not only allows the user to change the buffer space and control interval size, but it also allows the changing of certain attributes. The most notable of these allows the user exit to define the cluster as either UNIQUE or SUBALLOCATED, regardless of its original value. Specifying them on the restore command can also explicitly change the attributes mentioned previous, along with many others.

Empty Clusters

When restoring empty clusters, the only processing that takes place is the DEFINE. Therefore, the first time the data set is accessed after the restore, it must be processed in LOAD mode. A dummy record will be inserted and then deleted into any cluster that had a HURBA (High Used Relative Byte Address) greater than zero but had no data records in it. This option is useful primarily in an IMS or CICS environment where the data set must be initially loaded in order to be used.

Deleting Unexpired Data Sets

By specifying sysparm VSMPURGE with a value of Y (the default value), CA Disk will allow the deletion of unexpired data sets when the SCRATCH parameter is specified on a restore or recover. This is equivalent to using the PURGE parameter on an IDCAMS delete request.

Last Use Date Maintenance

When a data set is restored, CA Disk also restores the creation date and last use date fields to their original values. This default action can be customized as follows:

- Adding the CREATE parameter to the RESTORE command changes the creation date to the current date for the data set being restored.
- Supplying the LUSEDATE sysparm to a restore job changes the last use date to the current date for all data sets being restored by the job.
- Supplying the ARESCRDT sysparm to an auto-restore job changes the creation date to the current date for the data set being restored.

Depending on the criteria used during archive, these parameters can alleviate some concerns for exempting data set from archive. However, the opposite is also true. By changing their original dates, data sets can reside on DASD longer than necessary.

Example A

1. *userid.data.set* was last used on January 31st.
2. On March 1st, *userid.data.set* was archived based on the criteria CRI=(USED,LE,TODAY-28).
3. On April 1st, *userid.data.set* is restored. By default, it's last use date in the format-1 DSCB is set to **xx031**.
4. If the same archive job used in step 2, previous is executed before the restored data set is used, (that is, OPEN or CLOSE), then *userid.data.set* will be archived again.

If this is a problem to you, supplying sysparm LUSEDATy to the restore job used in step 3, previous will solve it. Doing so will change the last use date to the date the restore job execute.

Example B

1. *userid.data.set* was last used on January 31st.
2. On March 1st, *userid.data.set* was archived based on the criteria `CRI=(USED,LE,TODAY-28)`.
3. On April 1st, *userid.data.set* is restored using `LUSEDATEy`. The last use date in the format-1 DSCB is set to **xx091**.
4. If the same archive job used in step 2, previous is executed before the restored data set is used, (that is, OPEN or CLOSE), then *userid.data.set* is not archived because it does not match the criteria.

However, now that the last use date has been changed, the data set can reside on DASD longer than you can like.

For more information, see the sysparm descriptions for `ARESCRDTn`, and `LUSEDATEn` in the *Systems Guide*.

For more information regarding exempting data sets from archive, see Restore Grace Periods.

Special Considerations for VSAM

When a cluster is restored, CA Disk modifies the last use date after the definition of the cluster occurs. The last use date will be changed to the current date if the `LUSEDATE` sysparm is specified as Y.

Note: This support is only valid for those clusters defined in ICF catalogs.

Overriding Erase Option When Deleting

When a cluster is defined with the `ERASE` option, it means the data component's extents must be overwritten with binary zeros when it is deleted. Occasionally it is necessary to override this option in order to delete the cluster. When CA Disk deletes a cluster, it uses the value with which the cluster was defined (`ERASE` or `NOERASE`). If sysparm `VSNOERAS` is specified with a value of Y and a cluster defined with the `ERASE` option fails deletion with a specific return code, CA Disk will retry the deletion overriding the `ERASE` option (effectively making it a `NOERASE` option). This is equivalent to issuing an `IDCAMS DELETE` command with the `NOERASE` parameter specified. Note that even with this sysparm specified, CA Disk will attempt to delete the cluster first with its default value. Only if the sysparm is specified will CA Disk retry the deletion. (The default value for this sysparm is N, which allows the deletion to fail and processing is bypassed for the cluster).

Using NOSCRATCH Option When Deleting

An ICF catalog is made up of two components:

- A basic catalog structure (or BCS)
- One or more VSAM volume data sets (or VVDS).

The normal mode of operation for a delete is to remove the cluster's definition from both components. Occasionally, however, this is not possible. If the BCS and VVDS are on different volumes and the volume with the VVDS is destroyed, the BCS will still have pointers to the VVDS that is now destroyed. When this occurs (assuming you do not have a copy of the VVDS that can be recovered), it is necessary to delete the cluster entries from the BCS with the *noscratch* option prior to restoring the VSAM clusters. This tells catalog management to remove the entry from the BCS without trying to delete it from the VVDS. This can be done with IDCAMS, or can automatically be invoked by CA Disk by specifying sysparm VSNOSCRO with a value of Y.

With this parameter specified as a Y (the default value is N), CA Disk will reattempt any delete that fails with a VVDS error (during a normal cluster recovery operation) with the *noscratch* option. Note that the *noscratch* option will only be attempted if the normal delete fails.

Processing Clusters in Use by Other Tasks

Normal CA Disk processing determines the disposition type to use for data set allocation based on the share options with which a cluster is defined. If a cluster is defined with SHAREOPTIONS (1 3), it is allocated with DISP=SHR; otherwise it is allocated with DISP=OLD. By specifying sysparm VSALOSHR with a value of Y, all allocations will be made with DISP=SHR. In addition, user exit USERVENQ can be invoked to determine the allocation desired on the data set level.

Restoring to Predefined and Reusable Clusters

CA Disk allows restoring to predefined and reusable clusters by specifying sysparm VSPREDEF with a value of Y. CA Disk in determining when to use a predefined or reusable cluster's definition, employs the following logic:

1. An attempt to define the entity is performed. If the define is successful, the cluster was not predefined and normal restore procedures are used.
2. If the define failed because the entity already exists, a check is made to determine if the SCRATCH parameter was specified on the RESTORE or RECOVER command. If it was specified, the cluster is scratched and redefined, regardless of the value specified in sysparm VSPREDEF.
3. If the define failed and SCRATCH was not specified, but VSPREDEF is specified with a Y, the cluster's attributes are retrieved from the current (predefined) definition. If it is defined as reusable, and is not an empty cluster, it is opened with the reusable option and the restore continues. Otherwise, the restore for the data set is bypassed.

Note: When restoring data sets using the predefined or reusable option, CA Disk has to assume that the definition that has been found is indeed correct. Since it cannot be, CA Disk can assume no responsibility for the validity of the restored data. This option should only be used with the utmost care. Also note that empty clusters will not be restored over an existing data component when VSPREDEF is specified without the SCRATCH parameter.

Optional Cluster Data Report at Restore Time

By specifying sysparm VSRESCLD with a value of Y, CA Disk will produce a Cluster Data Report (CLD) for each cluster successfully restored. The report lines are inserted after each cluster's normal one-line entry on the *restored data sets* report. The lines that are printed are determined by the values specified for sysparm RPTCLSEL. This report allows you to review the actual attributes of the restored cluster without requiring a separate report to be run.

Determination of Data and Index Component New Names

When clusters are restored using the NEWNAME parameter, CA Disk must determine the new name to use for the data and index components. If AMS originally generated the name, CA Disk will allow VSAM to generate the new name for the component. However, if the user specified the component name at define time, CA Disk must generate a new name based on changes detected between the original base cluster name and the new name specified for the base cluster. In addition, the VSNEWNEX user exit is available to either specify a new component name or to instruct AMS to generate a new name. You can also assign new names on the RESTORE command via the DNAME and INAME parameters.

Determination of Alternate Index and Path New Names

When clusters with paths and/or alternate indexes associated with it are restored with the NEWNAME parameter, CA Disk must generate new names for these components. The names that it will generate are based on a comparison of the original base cluster name and its new name, along with a comparison of the original base cluster name and the original component name. CA Disk will be able to generate a much *cleaner* new name for its associated components if the new name is specified for a cluster either:

1. Has only the first two node names changed, with the remainder of the new name the same:
 - Original name = LABS.TJP.PMRFILE
 - New base name = LABS.TJP2.PMRFILE
2. Has only a suffix added to the original base cluster name:
 - Original name = LABS.TJP.PMRFILE
 - New base name = LABS.TJP.PMRFILE.TESTFILE

If the new name generated by CA Disk is unsatisfactory, you can assign your own new name. See the section RESTORE Examples in the chapter "Examples" for details.

Restoring Alternate Indexes

Alternate indexes are automatically redefined and rebuilt (by invoking the BLDINDEX facility of IDCAMS) when CA Disk restores a base cluster that had alternate index associations when it was archived. You can bypass the alternate index define and/or rebuild in 1 of 3 ways:

- By specifying NOAIXDEF and NOBLDINX on the Restore command
- By specifying AIX=NO on the DSCL Recover command
- By specifying AIX=DEFINEONLY on the DSCL Recover command

In addition, sysparms VSAIXDEF and VSBLDINX can be specified with a value of N to force CA Disk to bypass alternate index processing as a default. For example:

Archive a base cluster and all of its alternate indexes:

```
FIND DSN=clustername  
ARCHIVE DISP=RECAT
```

Restore a base cluster and rebuild all of its alternate indexes automatically:

```
RESTORE DSN=clustername
```

Restore a base cluster and rebuild two of its alternate indexes explicitly:

```
RESTORE DSN=clustername,NOAIXDEF  
RESTORE DSN=clustername,AIXNAME=alternateindex1name  
RESTORE DSN=clustername,AIXNAME=alternateindex2name
```

Recover under DSCL example:

```
RECOVER AIX=YES
```

Volume Selection for Alternate Indexes

Also, CA Disk must decide the volumes to which the alternate indexes are to be restored. If the base cluster is being restored to its original volume, the alternate indexes will automatically be restored to their original volumes, even if they were different from the base's. If the base cluster is restored to new volume(s), CA Disk uses the following logic for placement of the alternate indexes:

- If the alternate index originally resided on the same volumes as the base cluster, the new volumes will be used for the alternate index as well.
- The alternate index will go back to its original volume(s).

Restoring Alternate Indexes which were Archived Separately

In addition to the methods discussed previous, alternate indexes can also be archived and restored separately from their base clusters. For example:

Archive an alternate index by itself (CA Disk will retain the association information from the catalog in the FILES):

```
FIND DSN=alternateindexname  
ARCHIVE DISP=RECAT
```

Restore an alternate index by itself and re-establish association with the base cluster:

```
RESTORE DSN=alternateindexname
```

Note: In order to restore an alternate index as discussed previous, the associated base cluster must be preallocated and cataloged in the same catalog as the alternate index.

There are three critical precautions you must follow if you want to separately archive and restore alternate indexes:

- In archive, since the base cluster is not being archived at the same time as the alternate index, it is imperative that no updates to either file occur after the first archival begins. CA Disk cannot ensure this, so you must control this manually in order to keep the files synchronized. This becomes complicated if the data sets are selected implicitly by DSCL, because CA Disk processes the data sets in ascending data set name sequence without regard to the associations between the data sets (that is, the base cluster/alternate index associations). Finally, if you are going to archive the alternate indexes separately, the best thing to do is explicitly archive the base cluster and all alternate indexes in a single archive job. For example:

```
FIND DSN=alternateindex1name  
ARCHIVE DISP=RECAT  
FIND DSN=alternateindex2name  
ARCHIVE DISP=RECAT  
FIND DSN=alternateindex3name  
ARCHIVE DISP=RECAT  
FIND DSN=clustername  
ARCHIVE DISP=RECAT
```

- In restore, since the base cluster is not being restored at the same time as the alternate index, it is imperative that no updates to either file occur until after the last restore completes. For example, if the base cluster is restored first and the alternate index second, if the base cluster is updated prior to the completion of the restore of the alternate index, the two files will be unsynchronized. CA Disk cannot detect this type of error and no error will be indicated. After a base cluster and alternate index become unsynchronized, any application that accesses the base cluster through the alternate index can return erroneous information without the user being aware of any problem.

- Another problem can develop if you attempt to separately restore an alternate index that was defined over an ESDS cluster. If you restore the base cluster with a different CISIZE than its defined original, your separately restored alternate index will probably be invalid. This occurs because the alternate index contains pointers to the base cluster based on the relative byte address (RBA). When you restore the base cluster with a different CISIZE, the relative addresses of the records in the base cluster will probably change after the first control interval, thereby making all other offsets in the alternate index invalid. Although you can restore this alternate index, it cannot be used to access the data in the base cluster.

Volume Selection for Multi-volume Clusters

The DSCL RECOVER command has a VOLUMES= parameter on the SELECT command that assist in identifying volumes from which data sets are to be recovered. If the first volume portion of any multi-volume cluster existed on one of the named volumes, all portions will be recovered to their appropriate volumes. If the TOVOL= parameter is specified, however, RECOVER interprets this to mean that all of the multi-volume portions are to be moved to the single volume specified, which cannot have space to contain it. If it does fit, the extra space that it used can prevent other clusters from being recovered.

The RESTORE command works similarly, but provides greater flexibility. If only the DSN= parameter is specified, all portions of the cluster will be restored to their appropriate volumes, whether it is a single or multi-volume cluster. A VOL= parameter is available that acts the same as the TOVOL= parameter described for RECOVER previous; that is, all portions will be directed to the specified volume. However, parameters MVOL=(..list..), DVOL=(..list..) and IVOL=(..list..) are also available to supply multiple target volumes whenever a multi-volume cluster must be moved. The MVOL= list is sufficient if the data and index components can share the same volumes. The DVOL= and IVOL= lists are provided to allow these two components to be separated, and will override the MVOL= list if both are specified.

ICF Catalog Recovery

The volume recovery (not the restore) function supports the recovery of ICF catalogs. Although ICF catalogs are VSAM, the CA Disk recover function regards them as neither VSAM nor non-VSAM, but special data sets with special recovery requirements. The CATALOGS= parameter on the SELECT command determines whether ICF catalogs are included in the recovery. When included in the recovery, CA Disk always recovers the ICF catalogs first — before any of the data sets. This allows the catalogs to be updated correctly for those data sets that are subsequently included in the recovery process.

There are three parameters — CATALOGS=catname, DEFALIAS= YES/no, and EXPORTF=NO/yes — on the SELECT command that apply specifically to ICF catalog recovery. The CATALOGS= parameter provides a higher degree of control in deciding what catalogs are to be restored. It defaults to NO and must be specifically selected to be included.

As explained in that section, CA Disk dynamically links to IBM's IDCAMS IMPORT function to recover a catalog, and will then optionally redefine the alias entries in the master catalog. A few particulars regarding the IMPORT command should be noted. (See the IBM IDCAMS manual for more specific information.) RACF profiles, if any exist, are not affected by the catalog import. If the *old* catalog is still on the volume, it will be overlaid. A *new* catalog can be preallocated if desired (perhaps to increase the size), and the preallocated space will be used. If the catalog does not exist, IMPORT will define it. You cannot give the catalog a new name, however, as the IMPORT command does not support it. Another unexplained but observed phenomenon is that if the catalog was allocated in tracks, IMPORT will increase the allocation values if it has to redefine it. If track allocations were not used, IMPORT will reuse the original space values.

FILES Updates During RESTORE/RECOVER

When a restore/recover occurs the files data set archive record for each data set is updated. This record is called the DSNINDEX record and is described in Subfile Record Format —(DSNINDEX) in the *Systems Guide*. Three fields are updated in the DSNINDEX record for each restore. The first two are the date and time of the restore. The third field is a count of the number of restores that have been done from that DSNINDEX record. The restore date is used for 3 functions:

- The MERGE function uses this date in determining if a data set should be merged onto a different ARCHVOL
- The IXMAINT function uses this date to determine if a DSNINDEX record should be removed from the FILES
- The RECOVER function uses this date in conjunction with its RESDT criteria parameter

The restore time and count are currently for informational purposes only.

If the update of the DSNINDEX records during RESTORE/RECOVER is not desired because your data is not SMS managed or to expedite the recovery of multiple volumes, such as in a disaster recovery situation, see the sysparm description for SMSUPDATy in the *Systems Guide* for options to limit this update process. Note that this sysparm also controls accessibility to the Recover restart capabilities.

Use of Primary or Copy Archive Volumes

If a primary archive volume becomes unusable, the RESTORE/RECOVER functions can utilize the duplicate copy by disabling the primary volume. See the chapter "FILES Maintenance" for more information.

Default processing of the RESTORE/RECOVER functions will call for the primary archive volume (unless it has been disabled). If it is necessary to always use the duplicate volumes, such as in a disaster recovery situation, specify sysparm RESCOPYT with a value of Y.

Restrictions

The following is a list of restrictions that pertain to the restore/recover functions only. For other restrictions that apply to the CA Disk product, see the section General Restrictions in the chapter "General Information."

- When recovering a volume using the VTOC parameter, only the most recent copy of the VTOC that exists in the archives will be used as a screening list.
- Any PDS member that has a user defined TTRN in its directory will not be recovered or restored to a PDSE. If this is attempted, the member is bypassed, an informational message is issued, any possible scratch action against the member is disabled, and processing continues with the next member.
- NEWNAME not allowed if source allocation is PDS and target allocation is PDSE.

RESTORE JCL

Use the following JCL to restore one or more data sets or clusters immediately, or to have them placed into a queue to be processed at a later time (deferred requests).

```
//jobname JOB (acct,info),etc.
```

```
//STEP EXEC RESTORE
PASSWORD CATALOG=,MASTER=
        RESTORE DSNAME=,NEWNAME=,VOLUME=,VERSION=,TIMEDATE=,SCRATCH,ERASE,
        CREATE,DEFER,
```

Plus the following parameters honored only when the DEFER parameter is not specified:

```
NOLOAD,DATACLAS=,MGMTCLAS=,STORCLAS=
```

Plus the following parameters honored for non-VSAM data sets only:

```
POOL=,NOCATALOG,RECATALOG,SYSCTLG,
```

Plus the following parameters honored for non-VSAM data sets and only when the DEFER parameter is not specified:

```
MEMBER=,REPLACE,DIRBLKS=,PDSECONV,
```

Plus the following parameters honored for VSAM clusters only:

```
PASSWORD=,
```

Plus the following parameters honored for VSAM clusters and only when the DEFER parameter is not specified:

CATALOG=,RELATE=,RELPSWD=,

- Alternate index parameters:

AIXNAME=,AIXDEF,NOAIXDEF,BLDINX,NOBLDINX,

- Cluster level parameters:

ALLOC=,FREESPACE=,RECSZ=,BUFFERSPACE=,CISZ=,MASTERPW=,CONTROLPW=,
UPDATEPW=,READPW=,ATTEMPTS=,OWNER=,MVOLSERS=,FOR=,TO=,SHROPTIONS=,
KEYS=,NOERASE,IMBED,NOIMBED,ORDER,UNORDER,REPLICATE,NOREPLICATE,
REUSE,NOREUSE,SPAN,NOSPAN,SPEED,RECOVERY,UNIQUE,SUBALLOC,WRITECHK,
NOWRITECHK,

- Data component level parameters:

DNAME=,DALLOC=,DCISZ=,DMASTERPW=,DCONTROLPW=,DUPDATEPW=,DREADPW=,
DATTEMPTS=,DOWNER=,DSHROPTIONS=,DVOLSERS=,DORDER,DUNORDER,
DWRITECHK,DNOWRITECHK,

- Index component level parameters:

INAME=,IALLOC=,ICISZ=,IMASTERPW=,ICONTROLPW=,IUPDATEPW=,IREADPW=,
IATTEMPTS=,IOWNER=,ISHROPTIONS=,IVOLSERS=,IORDER,IUNORDER,IWRITECHK,
INOWRITECHK

RESTORE Password Command

If you are restoring non-VSAM data sets only, or if you do not use password protection of your VSAM clusters, you can omit this PASSWORD command.

If the catalog in which VSAM clusters are to be defined is update-protected, the catalog's master password must be supplied via this PASSWORD command. In addition, if the cluster being restored was password-protected and its current password does not match that which is on the archive medium, this command must be supplied with the proper password so that the scratch of the existing data set will not fail.

The PASSWORD command syntax is:

PASSWORD CATALOG=,MASTER=

CATALOG=

Specify the catalog name to which the clusters are being restored.

MASTER=

Specify the catalog's master-level password.

RESTORE Command Syntax

The RESTORE command syntax is:

CREATE

Specify this parameter if the creation date for the data set being restored is to be set equal to the current date. Default processing leaves the original creation date unaltered.

DSNAME=

Enter the full explicit name of the non-VSAM data set, VSAM cluster, physical alternate index, or uncataloged GDG entry to be restored.

Note: If the GDG name is currently cataloged, relative GDG names (for example, -2) are permitted. However, uncataloged versions are not locatable by version number and will be skipped even if the DSNINDEX record exists. It is recommended that if you uncatalog GDGs (for example, DISP=UNCAT) during archive, that you specify the fully qualified GDG entry name (for example, A.B.C.Gnnnnvnn suffix during RESTORE).

To restore alternate indexes based on the catalog information archived with a base cluster, enter the base cluster's name, not the alternate index names.

To restore an OS CVOL catalog, it must be given the name SYSCTLG in order to do proper formatting. Current support also requires that SYSCTLG be restored to the same device type from which it was archived.

NEWNAME=

The data set name to be assigned to the restored data set or cluster. A maximum of one entry is supported.

For GDGs, a relative GDG name can be specified.

For VSAM, this name applies only to the primary base cluster or alternate index being restored. If the data (and for a KSDS, the index also) component was originally assigned a name by Access Method Services, CA Disk will let AMS generate a new name for the component(s) again (unless sysparms VSAPPEND, VSDATSUF, or VSINXSUF are specified).

If the component name(s) were originally specified by the user, CA Disk will assign new names to the data and index components based on the differences detected between the original base cluster and the new name specified on the command. The data and index names can also be explicitly specified on the DNAME and INAME parameters.

If a base cluster is being restored (AIXNAME parameter is not specified) that had alternate index associations, the new name specified is the one to be given the base cluster. CA Disk will generate names for its alternate index(es) and path(s).

Note: Default action causes CA Disk to use the Base Cluster or AIX new name, depending on the path entry alignment, and appends *.PATH* to the PATH as the LLQ. If you require that only the HLQ be changed and the LLQ left unchanged, use sysparm NEWPATHN. For details, see NEWPATHNy in the *Systems Guide*.

If an alternate index is being redefined from a catalog definition found with the base cluster (AIXNAME parameter specified), the new name will be given to the alternate index being restored.

VOLUME=

The volume serial number of the device to which the data set is to be restored. The restore attempt will be terminated if the allocation cannot be on the specified volume. This parameter is mutually exclusive with the POOL parameter, and a maximum of one entry is supported.

VERSION=

To restore other than the most recent version of the data set contained in the CA Disk archives (either an archived copy or a backup copy), specify from -1 to -99 to get the successively older version of the data set.

TIMEDATE=

This parameter can be used to identify a specific version of the data set within the archives, rather than the relative version as supplied on the VERSION= parameter. Its use would be appropriate if a possibility exists that another version of the same data set will be placed in the archives before the restore can be completed.

The format for this parameter is *hhmmdate* where *hhmm* is the 24 hour time and *date* is any recognized CA Disk date format, described in the section Date Formats in the chapter "General Information." Correct values can be obtained by listing the CA Disk index entries for the data set via either batch or TSO.

SCRATCH

Use this parameter to scratch a non-VSAM data set when it exists on the target volume (providing the DSORG of the target data set matches the DSORG of the data set being restored) with the same name that the data set being restored is to have. If sysparm RESRCAT has a value of Y, CA Disk will attempt to scratch the target data set from its cataloged volume prior to restoration. With sysparm RESRCAT specified as N, CA Disk will scratch the target data set only if it resides on the same volume to which CA Disk attempted its allocation.

For VSAM, if a cluster already exists in the catalog with the same name that the cluster being restored is to have, CA Disk will delete the target data set name, then reallocate the target data set name and restore the data.

ERASE

If a partitioned data set is being restored to a preallocated data set, this parameter causes the existing contents to be erased and overlaid with the restored version.

If ERASE is not specified when restoring to a preallocated PDS, sufficient space must be left to load the members following the existing data.

If a VSAM cluster is being restored, this parameter indicates the cluster's data component is to be overwritten with binary zeros when its catalog entry is deleted.

DEFER

This parameter can be specified on the RESTORE command to indicate that the request is to be queued for later processing. It can be used in lieu of changing the RESTORE command to DRESTORE.

OUTDY=

The volume serial number of the device to which the CA Disk DSS data mover backed up data set is to be restored. This parameter is mutually exclusive with the parameters VOL= and TOVOL.

Notes: For the parameter:

Only for DSS restore

Full volume serial numbers; no / or *

Can accept up to 59 volume serial numbers

Specified volumes must be online

Parameters Honored Without the Defer Parameter

NOLOAD

This parameter causes CA Disk to allocate or define the data set, cluster, or catalog being restored, but not load any of its records. This can be useful if you want to restore a cluster or catalog's definition, and then issue an IDCAMS ALTER before restoring data to it (as a preallocated cluster).

Note: This parameter cannot be used with DEFER.

DATACLAS=

For a VSAM or non-VSAM SMS-managed data set, provide a new Data Class name to be assigned to the data set during allocation. The Data Class name must have been previously defined to SMS.

Note: This parameter is only valid when the system parameter SMSALLOC is specified with a value of Y.

Note: This parameter cannot be used with DEFER.

MGMTCLAS=

For a VSAM or non-VSAM SMS-managed data set, provide a new Management Class name to be assigned to the data set during allocation. The Management Class name must have been previously defined to SMS.

This parameter is only valid when the system parameter SMSALLOC is specified with a value of Y.

Note: This parameter cannot be used with DEFER.

STORCLAS=

For a VSAM or non-VSAM SMS-managed data set, provide a new Storage Class name to be assigned to the data set during allocation. Non SMS-managed data sets can become SMS managed by specifying this parameter. The Storage Class name must have been previously defined to SMS.

This parameter is only valid when the system parameter SMSALLOC is specified with a value of Y.

Note: This parameter cannot be used with DEFER.

Parameters for Non-VSAM Data Sets

POOL=

The name of the DASD pool to be used to find a volume on which to restore the data set. This is for non-VSAM data sets only (mutually exclusive with the VOL= parameter).

NOCATALOG

Suppress catalog action that would normally take place upon data set restoration.

RECATALOG

Default processing attempts to catalog all data sets, but will not change an existing catalog entry. Specify this parameter only when you want to force the data set to be cataloged to the target volume. See OFFLDASDn in the *Systems Guide* for instructions on how to do a delete noscratch for offline volumes.

SYSCTLG

Code this parameter if the RESTORE request is to be conditioned on the named data set being cataloged to the pseudo-volume (implying that the data set does not exist on DASD but only in the CA Disk archives).

PDSECONV

This optional parameter is required only when attempting to restore an archived PDSE to a PDS. When specifying this parameter, the value of SMSALLOC sysparm should be N and VOL= parameter specified.

During PDSE to PDS conversion, the number of directory blocks allocated will be as few as possible, based on the number of members in the source PDSE.

Parameters for Non-VSAM Without DEFER

MEMBER=

A list of one or more member names or patterns to be restored from the archived data set can be specified. See parameters REPLACE, ERASE and SCRATCH also. A maximum of 25 entries is supported.

If you specify the MEMBER= parameter without the ERASE parameter, CA Disk will allow concurrent restore processing and ISPF editing of a PDS, using the same enqueue convention as that used by ISPF edit or the linkage editor. See ENQ/DEQ/Reserve Usage in the *Systems Guide* for more information.

Note: This parameter cannot be used with DEFER.

REPLACE

This optional parameter permits members in an existing PDS to be replaced by duplicate named members being restored. Default processing merges in only those members that do not have conflicting names.

Note: This parameter cannot be used with DEFER.

DIRBLKS= **(field1,field2)**

This optional parameter permits the size of the directory to be adjusted. Field1 is a numeric value interpreted according to the value of field2. Field2 can be either PCT (for percent) or ABS (for absolute), and defaults to PCT if not specified.

(nn,PCT) causes the directory to be adjusted such that, as close as possible, the new directory will be nn percent full.

(nn,ABS) causes exactly nn directory blocks to be allocated, unless they are not enough to hold all of the entries. In this case, the minimum number that is sufficient will be used.

Note: This parameter is ignored when members are being merged into an existing PDS, and it cannot be used with the DEFER parameter.

Parameter for VSAM Clusters

The PASSWORD= optional parameter is required only when attempting to restore a cluster that was archived with assigned passwords. The password specified must match the cluster's master password as found on the archive medium. Optionally, you can supply the CA Disk master password.

Note: This parameter is required whenever a data set is being restored that had passwords assigned, regardless of whether the receiving catalog is protected or not.

Parameters for VSAM Without DEFER

Specify one or more of the following parameters only if you want to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, ALLOC= and DALLOC=), the value specified at the data or index level will take precedence over the cluster value, just as IDCAMS does when you issue a DEFINE. See the IBM publication Access Method Services Reference for a more complete description of the following parameters.

CATALOG=

This optional parameter applies only to the restore of a VSAM cluster. It specifies the name of a VSAM catalog in which the restored cluster is to be defined (overriding the catalog selected by normal catalog management functions). If this parameter is used, you should also specify the same catalog in a STEPCAT DD statement. If you don't, VSAM OPEN errors are likely to occur. This is caused by VSAM OPEN routines always following the standard catalog management rules to locate the proper catalog, regardless of the catalog specified for the define processing.

Note: For non-ICF catalogs, a named catalog must own the volume to which the cluster is being restored. Use caution when adding a STEPCAT, since all new catalog entries (non-VSAM as well as VSAM) will be placed in the STEPCAT catalog.

RELATE=

This parameter must be specified when you are restoring an alternate index (either a physical AIX or one from the catalog information archived with the base cluster) to a base cluster that was restored with a new name. Give the new name that the base cluster was restored with in this parameter.

RELPSWD=

This parameter must be specified when an alternate index is being restored and its base cluster was defined with passwords different from those on the archive tape. Specify the update or higher password for the base cluster that the alternate index is being related to. If a physical alternate index is being restored, this parameter must always be specified if the base cluster is password-protected.

Parameters for VSAM AIXS Without DEFER

Specify one or more of the following parameters only if you want to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, ALLOC= and DALLOC=), the value specified at the data or index level will take precedence over the cluster value, just as IDCAMS does when you issue a DEFINE. See the IBM publication Access Method Services Reference for a more complete description of the following parameters.

AIXNAME=

Specify this parameter when you want to restore a specific alternate index from the catalog information archived with the base cluster. The name given on this parameter must be the alternate index name that was present when the base cluster was archived. The DSNNAME parameter must be specified with the base cluster name that contains the definition. The base cluster is NOT restored with this command — only the specific alternate index is. If you are attempting to restore a physical alternate index, it cannot be specified with this parameter (it must be on its own RESTORE command and specified in the DSNNAME parameter). This parameter can be used when you want to restore an alternate index with new attributes. Normally if a base cluster is being restored with its alternate indexes, any parameters defining new attributes are assigned only to the base cluster. The alternate index is restored with the same attributes it was archived with. If you want to change the alternate index you can specify the following:

```
RESTORE DSN=base.cluster.name,CISZ=4096,  
        NOAIXDEF                                don't define AIX(S)  
RESTORE DSN=base.cluster.name,  
        AIXNAME=aix.name,                        alternate index name  
        BUFFER=32000,                            new bufferspace  
        NEW=new.aix.name                        new name for AIX
```

AIXDEF

This parameter causes CA Disk to redefine all of the alternate indexes associated with the base cluster when the cluster is restored. This is the default option unless sysparm VSAIXDEF is specified with a value of N.

NOAIXDEF

This parameter causes CA Disk to restore only the base cluster without redefining any of its alternate indexes.

BLDINX

This parameter causes CA Disk to invoke IDCAMS to rebuild any alternate indexes (BLDINDEX) that CA Disk can define for a base cluster. This is the default option unless sysparm VSBLDINX is specified with a value of N. This parameter is ignored if a physical alternate index is being restored.

NOBLDINX

This parameter causes CA Disk to bypass the call to IDCAMS to rebuild the alternate index. This parameter is ignored if a physical alternate index is being restored.

Parameters for VSAM Clusters Without DEFER

Specify one or more of the following parameters only if you want to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, ALLOC= and DALLOC=), the value specified at the data or index level will take precedence over the cluster value, just as IDCAMS does when you issue a DEFINE. See the IBM publication Access Method Services Reference for a more complete description of the following parameters.

ALLOC=

This parameter allocates space for the cluster by type, primary amount and secondary amount. The format is ALLOC=(TYP,PA,SA)

Where:

TYP CYL (cylinders), TRK (tracks), or REC (records)

PA Primary allocation amount — must be from 1 to 99999999

SA Secondary allocation amount — must be from 0 to 99999999

FREESPACE=

Specify the percent of space to be left free in control intervals and control areas.

Valid values are from 0 through 100, inclusively. The format is

FREESPACE=(CI-percent, CA-percent)

RECSZ=

Specify the average and maximum record size in bytes. The format is

RECSZ=(AVE,MAX).

Where:

AVE Average record size.

MAX Maximum record size. If left blank, the average record size is assumed to also be the maximum.

BUFFERSPACE=

Specify the amount of buffer space to be provided in bytes. The largest amount is

99999. If a value of 0 is specified, Access Method Services will calculate a new

buffer space based on data and index control interval size.

CISZ=

Specify the size of the control interval. The largest CI size is 32768. If this parameter is specified and BUFFERSPACE is not, CA Disk will zero out the buffer space value passed to catalog management so that a new buffer space amount can be calculated based on the new control interval size. If a value of 0 is specified, Access Method Services will calculate a new control interval size.

Caution should be used when restoring an ESDS cluster that belongs to IMS or CICS — changing the CI size on these clusters can destroy internal RBA pointers and cause corruption of the database.

If the cluster was archived in control interval image copy format, changing the control interval size will result in a record length insert error when restoring the cluster. This parameter should not be used in that case.

MASTERPW=

Specify the new master level password to be assigned to the cluster or alternate index.

CONTROLPW=

Specify the new control interval level password to be assigned to the cluster or alternate index.

UPDATEPW=

Specify the new update level password to be assigned to the cluster or alternate index.

READPW=

Specify the new read level password to be assigned to the cluster or alternate index.

ATTEMPTS=

Specify the number of attempts (0 to 7) for entering a password.

OWNER=

Specify the owner ID. It is 1 to 8 alphanumeric or special characters.

MVOLSERS=

Specify a list of from one to twenty volumes on which the cluster is to be allocated. Non-MVS accounts are limited to a total of five volumes.

FOR=

Specify the retention period of the cluster in days. The valid range is from 0 to 9999.

TO=

Specify the expiration date in Julian format (yyddd).

SHROPTIONS=

Specify how the cluster can be shared among users. The format is SHROPTIONS=(CR,CS).

Where:

CR Cross-region options are 1 through 4

CS Cross-system options are 3 through 4

KEYS=

Specify the key length and offset in the record for a key-sequenced data set. The format is KEYS=(LN,OF).

Where:

LN Length of the key

OF Offset of the key into the record

NOERASE

This parameter indicates the cluster's data component is not to be overwritten with binary zeros when its catalog entry is deleted.

IMBED

This parameter indicates the sequence set of the cluster is to be written with the data component.

NOIMBED

This parameter indicates the sequence set of the cluster is not to be written with the data component.

ORDER

This parameter indicates the volume(s) are to be used in the order listed.

UNORDER

This parameter indicates the volume(s) are to be used in any order.

REPLICATE

This parameter indicates each index record is to be written around a track as many times as it will fit.

NOREPLICATE

This parameter indicates each index record is not to be written around a track as many times as it will fit.

REUSE

This parameter indicates the cluster can be opened again with the high-used RBA set to zero.

NOREUSE

This parameter indicates the cluster cannot be opened again with the high-used RBA set to zero.

SPAN

This parameter indicates the cluster can have records longer than the control interval.

NOSPAN

This parameter indicates the cluster cannot have records longer than the control interval.

SPEED

This parameter indicates the data component is not to be preformatted before records are loaded.

RECOVERY

This parameter indicates the data component is to be preformatted before records are loaded.

UNIQUE

This parameter indicates the cluster's space allocation is to come from the volume's available space, not from a VSAM data space.

SUBALLOC

This parameter indicates the cluster's space allocation is to come from a VSAM data space on the volume.

WRITECHK

This parameter indicates a check is to be issued after every write.

NOWRITECHK

This parameter indicates a check is not to be issued after every write.

Parameters for VSAM Data Components Without DEFER

Specify one or more of the following parameters only if you want to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, ALLOC= and DALLOC=), the value specified at the data or index level will take precedence over the cluster value, just as IDCAMS does when you issue a DEFINE. See the IBM publication Access Method Services Reference for a more complete description of the following parameters.

DNAME=

This parameter specifies a new data component name. You can specify either an explicit data set name, or a symbolic name that starts with a "\$". A "\$" instructs CA Disk to use the cluster name as a base name, and to append to it the string following the "\$". A second "\$" causes the string ".DATA" to be appended to the cluster name.

For example with a cluster name of VSAM.CLUSTER:

- If DNAME=VSAM.CLSTR.DATA.COMP, the data component name will be VSAM.CLSTR.DATA.COMP.
- If DNAME=\$\$, the data component name will be VSAM.CLUSTER.DATA.
- If DNAME=\$.DATA.COMP, the data component name will be VSAM.CLUSTER.DATA.COMP.
- If DNAME=\$DATA.COMP, the data component name will be VSAM.CLUSTER.DATA.COMP.

DALLOC=

This parameter allocates space for the data component by type, primary amount and secondary amount. The format is DALLOC=(TYP,PA,SA).

Where:

TYP CYL (cylinders), TRK (tracks), or REC (records)

PA Primary allocation amount — must be from 1 to 99999999

SA Secondary allocation amount — must be from 0 to 99999999

Note: CA Disk will specify the values you request, but IBM's catalog management can change it and give absolutely no warning that it has done so. Combinations of attributes that it does not like usually cause this.

DCISZ=

Specify the size of the control interval in the data component. The largest CI size is 32768. If this parameter is specified and BUFFERSPACE is not, CA Disk will zero out the buffer space value passed to catalog management so that a new buffer space amount can be calculated, based on the new control interval size. If a value of 0 is specified, Access Method Services will calculate a new control interval size.

Caution should be used when restoring an ESDS cluster that belongs to IMS or CICS—changing the CI size on these clusters can destroy internal RBA pointers and cause corruption of the database.

DMASTERPW=

Specify the new master level password of the data component.

DCONTROLPW=

Specify the new control interval level password of the data component.

DUPDATEPW=

Specify the new update level password of the data component.

DREADPW=

Specify the new read level password of the data component.

DATTEMPTS=

Specify the number of attempts (0 to 7) for entering a password for the data component.

DOWNER=

Specify the new owner ID of the data component. It is 1 to 8 alphanumeric or special characters.

DSHROPTIONS=

Specify how the data component can be shared among users. The format is DSHROPTIONS=(CR,CS).

Where:

CR Cross-region options are 1 through 4

CS Cross-system options are 3 through 4

DVOLERS=

Specify a list of from one to twenty volumes on which the data component is to be allocated. Non-MVS accounts are limited to a total of five volumes.

DORDER

This parameter indicates the volume(s) are to be used in the order listed.

DUNORDER

This parameter indicates the volume(s) are to be used in any order.

DWRITECHK

This parameter indicates a check is to be issued after every write.

DNOWRITECHK

This parameter indicates a check is not to be issued after every write.

Parameters for VSAM Index Components Without DEFER

Specify one or more of the following parameters only if you want to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, ALLOC= and DALLOC=), the value specified at the data or index level will take precedence over the cluster value, just as IDCAMS does when you issue a DEFINE. See the IBM publication Access Method Services Reference for a more complete description of the following parameters.

INAME=

This parameter specifies a new index component name. You can specify either an explicit data set name or a symbolic name that starts with a "\$". A "\$" instructs CA Disk to use the cluster name as a base name and to append to it the string following the "\$". A second "\$" causes the string ".INDEX" to be appended to the cluster name.

For example with a cluster name of VSAM.CLUSTER:

- If INAME=VSAM.CLSTR.INDEX.COMP, the index component name will be VSAM.CLSTR.INDEX.COMP.
- If INAME=\$\$, the index component name will be VSAM.CLUSTER.INDEX.
- If INAME=\$.INDEX.COMP, the index component name will be VSAM.CLUSTER.INDEX.COMP.
- If INAME=\$INDEX.COMP, the index component name will be VSAM.CLUSTER.INDEX.COMP.

IALLOC=

This parameter allocates space for the index component by type, primary amount and secondary amount. The format is IALLOC=(TYP,PA,SA).

Where:

TYP CYL (cylinders), TRK (tracks), or REC (records)

PA Primary allocation amount — must be from 1 to 99999999

SA Secondary allocation amount — must be from 0 to 99999999

Note: CA Disk will specify the values you request, but IBM's catalog management can change it and give absolutely no warning that it has done so. Combinations of attributes that it does not like usually cause this.

ICISZ=

Specify the size of the control interval in the index component. The largest CI size is 32768. If this parameter is specified and BUFFERSPACE is not, CA Disk will zero out the buffer space value passed to catalog management so that a new buffer space amount can be calculated based on the new control interval size. If a value of 0 is specified, Access Method Services will calculate a new control interval size.

IMASTERPW=

Specify the new master level password of the index component.

ICONTROLPW=

Specify the new control interval level password of the index component.

IUPDATEPW=

Specify the new update level password of the index component.

IREADPW=

Specify the new read level password of the index component.

IATTEMPTS=

Specify the number of attempts (0 to 7) for entering a password for the index component.

IOWNER=

Specify the new owner ID of the index component. It is 1 to 8 alphanumeric or special characters.

ISHROPTIONS=

Specify how the index component can be shared among users. The format is ISHROPTIONS=(CR,CS).

Where:

CR Cross-region options are 1 through 4

CS Cross-system options are 3 through 4

IVOLSERS=

Specify a list of from one to twenty volumes on which the index component is to be allocated. Non-MVS accounts are limited to a total of five volumes.

IORDER

This parameter indicates the volume(s) are to be used in the order listed.

IUNORDER

This parameter indicates the volume(s) are to be used in any order.

IWRITECHK

This parameter indicates a check is to be issued after every write.

INOWRITECHK

This parameter indicates a check is not to be issued after every write.

JCL for Processing Queued Restore Requests

To dispose of the deferred restore requests — that is, to actually restore the data sets — the same JCL is used as for immediate restores, with the exception that the module executed is ADSDM279. (Since the input consists of the queued requests, no //SYSIN stream is needed.)

```
//jobname JOB (acct,info),etc.
//DISPOSE EXEC RESTORE,PARM.RESTORE=ADSDM279
```

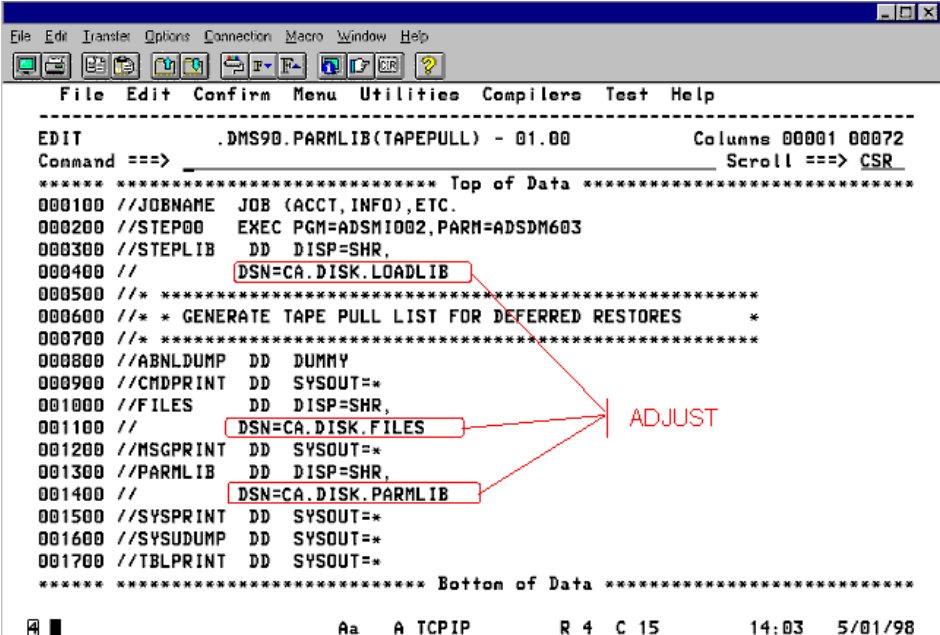
The completion status for each request processed is maintained within the RESTCMDS subfile, indicating if the request is pending or complete and, if completes, the time of completion.

Note: The MFILES DD statement is not supported in dispose processing. If you have more than one FILES, you must execute DISPOSE separately for each one.

Tape Pull List for Queued Restore Requests

If you use the deferred restore capability, you can also want to generate a list of tapes to be pulled out of the tape library prior to running the job that actually restores the data sets. A utility to create this list is provided.

The following JCL can be used. (It is also supplied as member PULLJCL in the CA Disk installation library.) The only alteration needed is to supply the proper data set names as indicated in the sample JCL.



```

File Edit Transfer Options Connection Macro Window Help
File Edit Confirm Menu Utilities Compilers Test Help
-----
EDIT          .DMS90.PARMLIB(TAPEPULL) - 01.00          Columns 00001 00072
Command ==>          Scroll ==> CSR
***** ***** Top of Data *****
000100 //JOBNAME  JOB (ACCT,INFO),ETC.
000200 //STEP00  EXEC PGM=ADSM1002,PARM=ADSDM603
000300 //STEPLIB  DD DISP=SHR,
000400 //          DSN=CA.DISK.LOADLIB
000500 //* *****
000600 //* * GENERATE TAPE PULL LIST FOR DEFERRED RESTORES *
000700 //* *****
000800 //ABNLDUMP  DD DUMMY
000900 //CMDPRINT  DD SYSOUT=*
001000 //FILES    DD DISP=SHR,
001100 //          DSN=CA.DISK.FILES
001200 //MSGPRINT  DD SYSOUT=*
001300 //PARMLIB   DD DISP=SHR,
001400 //          DSN=CA.DISK.PARMLIB
001500 //SYSPRINT  DD SYSOUT=*
001600 //SYSUDUMP   DD SYSOUT=*
001700 //TBLPRINT  DD SYSOUT=*
***** ***** Bottom of Data *****
Aa  A TCPIP          R 4 C 15          14:03  5/01/98

```


To generate the list, run this JCL prior to running the dispose job as described previous. The only condition this utility does not provide for is the insertion of a DRESTORE request after the running of this job but before the dispose job. In this case, the dispose job can call for a mount of an archive tape that was not in the list.

Condition Codes for Tape Pull List

The following condition codes are set at the completion of the pull list job:

0

One or more tapes or cartridges are needed.

4

No tapes or cartridges are needed.

Listing of Deferred Restore Requests

The status of the deferred restore requests can be listed using the following JCL:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC LISTREQ  
        LISTREQ DSN=,RESTORE,ALL
```

DSNAME=

A list of one or more data set names or prefixes for which request entries are to be listed. If this parameter is not specified, the ALL parameter as defined below must be specified. (DSN=/ is equivalent to the ALL parameter.)

RESTORE

The presence of this parameter results in queued DRESTORE request records being selected. It can be abbreviated as RES.

ALL

The presence of this parameter results in all the DRESTORE request records being selected.

Erase a Deferred Restore Request from the Queue

The DERASE command can be used with the following JCL to delete requests from the DRESTORE queue:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DERASE  
        DERASE DSN=,RESTORE
```

DSNAME=

A list of one or more data set names or prefixes for which deferred requests are to be deleted. An acceptable abbreviation for this keyword is DSN. A maximum of 20 data set names or prefixes are allowed.

DSN=/ results in all DRESTORE requests being deleted.

RESTORE

The presence of this parameter indicates that the deferred requests to be deleted are restore transactions. It can be abbreviated as RES.

Implicit Recovery and Volume Defragmentation

A group of data sets, data sets on an entire volume, or a group of volumes, however, are restored through the RECOVER command. Recover processing makes use of all of the applicable RESTORE parameters, plus several additional ones needed to provide greater flexibility unique to the recovery process.

When recovery is being applied to a new volume, or to one or more volumes with sufficient space, the standard benefits provided by individual data set restores are expanded to the volume level:

- Multiple data set extents are combined, resulting not only in defragmented data sets, but defragmented volumes as well.
- PDS data sets are compressed.
- ISAM and VSAM clusters are reorganized.
- Different device types can be used.
- CATALOG entries can be recreated (including VSAM).

See the RESTORE command to restore an individual data set or cluster.

Features Unique to RECOVER Command

The following additional features are unique to RECOVER:

- SIMULATE mode — to see just what data sets will be recovered, based upon the supplied command and parameters.
- VTOC-based recovery — limiting data sets to those listed in the most recent backup copy of the volume's VTOC.
- Date and Time recovery - telling recovery to use only those backups that were taken within a specified time period.
- Backups-only selection — skipping the restores for those data sets that were scratched when placed in the CA Disk archives.
- Non-VSAM and VSAM-only separation — the ability to process these two distinctly different types separately if desired. Default processing will recover both types.

Note: If an OS CVOL or a VSAM ICF catalog resides on a volume being recovered and are selected for processing, they will be restored first. This permits any catalog updates done for the remaining recovered data sets to be properly entered. See below for detailed information regarding the recovery of ICF catalogs.

- Restart capability
- Implicit selection based upon fields unique to the DSNINDEX record

VSAM Cluster Recovery Only

The following notes pertain to VSAM cluster recovery only:

- Clusters can be moved freely between ICF and non-ICF catalogs
- When recovering suballocated (non-unique) VSAM clusters to a new volume, the user must predefine the VSAM space as desired. After this space is created, CA Disk will dynamically define (redefine) the VSAM clusters into it. Unique clusters will have their self-containing spaces defined automatically by CA Disk.
- Predefined (preallocated) VSAM clusters are permitted only under certain conditions. See the previous discussion for detailed information.
- TOCAT= specification — used to specify the name of a VSAM catalog in which the restored clusters are to be defined. For VSAM catalogs only, the named catalog must own the volume to which the clusters are being restored.

Note: This parameter is normally not needed, assuming that the master catalog contains the correct alias entries to cause it to select the proper catalog. If it does not, a STEPCAT should be provided, in which case this parameter still isn't needed.

- CATALOGS=, DEFALIAS= and EXPORTF= parameters on the SELECT command pertain solely to the recovery of ICF catalogs.

RECOVER Command

The CA Disk recover function (executed under the FMS PROC discussed in the chapter "Basic System – FMS PROC"), allows you to restore data sets and OS catalogs without the need for preallocation. You can also restore VSAM clusters and ICF catalogs. In most cases, you can even restore to a device type other than that from which the data set was backed up or archived. The following is a list of the data sets supported:

- Physical Sequential (PS and PSE)
- Direct Access (DA)
- Indexed Sequential (IS)
- Partitioned Organization (PO, POE and PO2)
- Access Method—VSAM (AM)

Command Available: RECOVER

JCL Proc Executed: FMS

RECOVER Command General Information

The recover function begins by filtering all of the candidate data sets and sequencing them by file number within each archive tape (or archive data set on disk), the optimum performance arrangement. All data sets being restored from the archives on disk are done before those archived to tape.

Selection Criteria

All applicable Selection Criteria are documented in the section Criteria Statement Operands in the chapter "Basic System – FMS PROC," and can be used to restrict Recover to any combination of selection criteria. See that section before attempting a recover.

Multiple FILES for Recover

Multiple FILES are supported for Restore, Deferred Restore, and Recover processing. If you want to make them all available to these functions, you should use the MFILES DD statement instead of the FILES DD statement. Add each data set name to the MFILES concatenation; up to 256 data sets are supported. If using a concatenation of all Files Data Sets (FDS) or a mixture of FDS and Files Database (FDB), then the order of the concatenation determines the order of search for the data set to be restored. The first one that contains an index record for the archived data set will be used, even if another contains a more recent version of the data set. Therefore, it is very important that they be specified in the proper order. When using a concatenation of all FDBs, they are searched as logically joined so the most current version of the data set is found regardless of the concatenation order.

Note: The MFILES DD statement is not supported in Dispose processing. If you have more than one FILES, you must execute DISPOSE separately for each one.

We do not recommend concatenating data sets. If you concatenate data sets, you will get errors.

RECOVER Command Condition Codes

For condition codes received from the FMS PROC see the section Condition Codes in the chapter "Basic System – FMS PROC."

If recovering for disaster recovery where original volume(s) are offline, RECOVER will automatically do a delete noscratch prior to data set allocation — no special sysparm is required. However, for batch RESTORE there is. For details see sysparm OFFLDASDn in the *Systems Guide*.

More information:

[Example 16—Recovery with GDG and GDGLIMIT](#) (see page 669)

RECOVER Command Syntax

The syntax of the RECOVER command is:

```
RECOVER  
TOVOLUME=,POOL=,DISP=,NEWHLQ=,SCRATCH,CREATE,PDSECONV,ERASE,NLOAD,AIX=,TOCAT=,  
EXPORTF=,DEFALIAS=,DATACLAS=MGMTCLAS=,STORCLAS=,REPORT,CATGDGBASE,CATTAPEDS,  
CATALIAS,GDGLIMIT=
```

TOVOLUME=

By default, the recovery of data set(s) is to the volume(s) from which the data set(s) were backed up or archived. This parameter allows for an alternate target volume to be used. A maximum of one explicit entry is supported.

POOL=

This parameter can be used to specify the name of a DASD Pool (a list of target volumes) that RECOVER can select from. This parameter is limited to only non-VSAM data sets, and is mutually exclusive with the TOVOLUME= parameter. For information on how to setup DASD Pool support, see the sysparm description of DFLTPOOLxxxxxxx and the PARMLIB description of DASDPOOL in DASD Pool List in the *Systems Guide*.

DISP=

This parameter indicates to CA Disk what should be the final catalog disposition of the restored non-VSAM data set(s). VSAM data sets will always be cataloged after restore, regardless of the value specified for this parameter. Three values are supported:

CATALOG

This is the default value, and will cause all recovered data sets to be cataloged. If a recovered data set is already cataloged, the existing catalog entry is left unaltered.

NOCATALOG

This non-VSAM only value prevents RECOVER from cataloging any data sets to the target volume.

RECATALOG

This non-VSAM only value forces RECOVER to catalog the data sets to the target volume. This value will force the recatalog of existing data sets by the same name to the target volume as well. For more information, see the sysparm description of RESRCATn in the *Systems Guide*.

NEWHLQ2=

This parameter forces a rename of all recovered data sets. Each data set recovered will have High Level Qualifier 2 (node 2) changed to the value of this parameter. This parameter can be used in conjunction with NEWHLQ (node 1). See MSG 4485 for errors in using this parameter; the data set may be ineligible or too long. Please be aware that when selecting multiple data sets, the outcome of a FMS Recover using NEWHLQ and/or NEWHLQ2 can be adversely affected. When using other than the defaults settings for sysparms like PREALLOC, RECCHCAT, RESRCAT or VSPREDEF, as well as a command parm like SCR, special care must be taken in order to ensure that the new data set names do not duplicate each other during the recover operation. Executing the FMS Recover in simulate mode and review of the report output can help identify any duplication.

SCRATCH

Indicates that if the current data set being processed already exists on the target volume, scratch it prior to recovery. For more information, see the sysparm description of RESRCATn in the *Systems Guide*.

CREATE

This parameter causes CA Disk to set the creation date in the Format-1 DSCB for each recovered data set to the current date. Default processing leaves the original creation date unaltered.

PDSECONV

This parameter causes CA Disk to recover a PDSE as a PDS. To do so, you must specify PDSECONV and TOVOLUME= on the RECOVER command, and set sysparm SMSALLOC to 'N' to not supply the original DFSMS class names during allocation. For more details, see the sysparm description for SMSALLOCn in the *Systems Guide*.

ERASE

When recovering a PO data set to a preallocated PO, the presence of this parameter erases the contents of the target data set with the restored version. If ERASE is not specified when recovering to an existing PO, sufficient space must be available for RECOVER to *merge* the restored members into the data set.

For more information, see the sysparm description for PREALLOCn in the *Systems Guide*.

NOLOAD

This parameter causes CA Disk to allocate or define the data sets, clusters, or catalogs being recovered onto the target volume, but not load any of their records.

Note: This parameter has no effect on catalog recovery if any of the CAT options (See CATALIAS for more information) is also specified. This means that the command *RECOVER NOLOAD, CATALIAS* will recover all ALIAS entries.

AIX=

This parameter indicates how alternate indexes are to be processed for the clusters being recovered. Three values are supported:

YES

This is the default value, and will cause all clusters being recovered to have their associated alternate indexes defined and rebuilt.

DEFINEONLY

Causes all clusters to have any associated alternate indexes defined, but not rebuilt. This parameter can be abbreviated DEF.

NODEFINE

Causes all clusters that have associated alternate indexes to not be defined or rebuilt. This parameter can be abbreviated NO.

TOCAT=

Specifies the catalog where the recovered cluster should be defined. It applies to VSAM only, and overrides the catalog selected by normal catalog management facilities.

EXPORTF=

Indicates whether or not CA Disk should recover ICF catalogs. Two values are supported:

NO

This is the default value, and recovers (IMPORT) ICF catalogs directly (that is, recreates the catalog).

YES

This option causes CA Disk to create a sequential file in *Export-Temp* format, allowing you to execute the IMPORT function as a separate job. For more information, see the sysparm descriptions for CATBKNAME, CATBKSPC, and CATBKUNT, in the *Systems Guide*.

Note: Using EXPORTF is not the only way to recover a catalog. A catalog can be recovered with any RECOVER command, providing the command doesn't specify NOLOAD, CATGDGBASE, CATTAPEDS, or CATALIAS.

DEFALIAS=

Indicates whether CA Disk should redefine alias entries into the master catalog. Two values are supported:

YES

This is the default value, and causes alias entries for the catalog being recovered to be redefined into the master catalog.

NO

Prevents the alias entries from being redefined.

Note: This parameter is ignored when EXPORTF=YES is specified.

DATACLAS=

For VSAM or non-VSAM SMS-managed data sets, provide a new Data Class name to be assigned to the data sets during allocation. The Data Class name must have been previously defined to SMS.

This parameter is only valid when the system parameter SMSALLOC is specified with a value of "Y".

Note: This parameter cannot be used with DEFER.

For more information pertaining to the RESTORE command, see DATACLAS=.

MGMTCLAS=

For VSAM or non-VSAM SMS-managed data sets, provide a new Management Class name to be assigned to the data sets during allocation. The Management Class name must have been previously defined to SMS.

This parameter is only valid when the system parameter SMSALLOC is specified with a value of Y.

Note: This parameter cannot be used with DEFER.

For more information pertaining to the RESTORE command, see MGMTCLAS=.

STORCLAS=

For VSAM or non-VSAM SMS-managed data sets, provide a new Storage Class name to be assigned to the data sets during allocation. Non SMS-managed data sets can become SMS-managed by specifying this parameter. The Storage Class name must have been previously defined to SMS.

This parameter is only valid when the system parameter SMSALLOC is specified with a value of Y.

Note: This parameter cannot be used with DEFER.

For more information pertaining to the RESTORE command, see STORCLAS=.

GDGLIMIT=

Specify the number of generations of GDGs to be recovered. This parameter is not required for CA Disk to recover generation data sets. Cataloged rolled-off generations are not counted as part of the limit since they are no longer part of the sphere and they are always recovered.

Note: This parameter only applies to FMS RECOVER

REPORT

The purpose of this parameter is to assist you in knowing in advance what resources are required if the application selected needs to be recovered.

Note: This parameter is only valid when **simulating** an FMS RECOVER.

OUTDY=

The volume serial number of the device to which the CA Disk DSS data mover backed up data set is to be restored. This parameter is mutually exclusive with VOL= or TOVOL=parameters.

Notes: for the parameter

- Only for DSS restore
- Full volume serial numbers; no "/" or "*"
- Can accept up to 59 volsers
- Specified volumes must be on-line

When specified, three reports are generated:

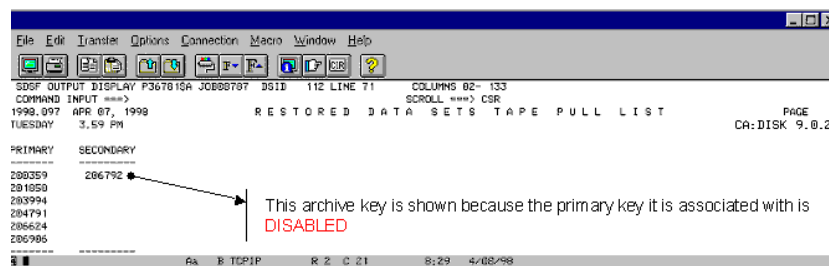
Disk Archive List

This report generates a list of Archive disk keys that would be required if you decide to execute RECOVER in LIVE mode. A sample of this report is illustrated as follows:

VOLSER	ARCHIVE KEY	ARCHIVE DATA SET NAME
QATEST	092251	QDHSTEST.QAR900.ARCH.IMSC.D1998097.T014015
QATEST	092252	QDHSTEST.QAR900.ARCH.IMSC.D1998097.T018725
QATEST	092255	QDHSTEST.QAR900.ARCH.IMSC.D1998097.T018620
QATEST	092258	QDHSTEST.QAR900.ARCH.IMSC.D1998097.T004136
QATEST	092259	QDHSTEST.QAR900.ARCH.IMSC.D1998097.T004037
QATEST	092260	QDHSTEST.QAR900.ARCH.IMSC.D1998097.T002222
QATEST	092261	QDHSTEST.QAR900.ARCH.IMSC.D1998096.T212715
QATEST	092262	QDHSTEST.QAR900.ARCH.IMSC.D1998096.T211548
QATEST	092263	QDHSTEST.QAR900.ARCH.IMSC.D1998096.T211513
QATEST	092264	QDHSTEST.QAR900.ARCH.IMSC.D1998096.T211432
QATEST	092271	QDHSTEST.QAR900.ARCH.IMSC.D1998096.T211038

Tape Pull List

This report generates a list of Archive tape and cartridge keys. The content is controlled by sysparm RESCOPYT, described in RESCOPYTn in the *Systems Guide*. With RESCOPYT set to 'N', all enabled primary keys associated with the data sets scheduled for recovery is displayed. Also listed are any **secondary keys** for disabled primaries.



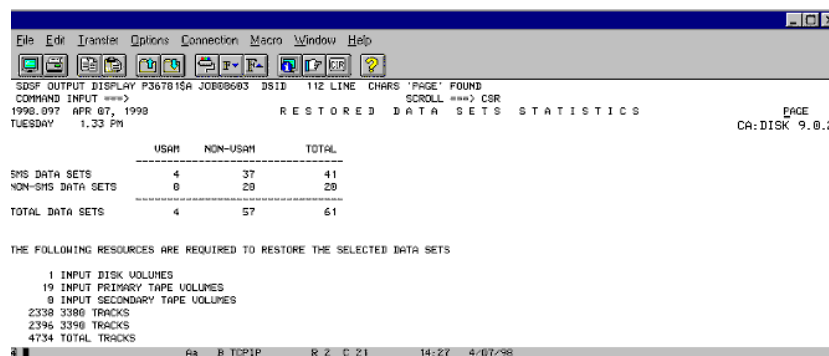
With RESCOPYT set to Y, only **enabled secondary keys** will be displayed.

Important! Using this setting can result in a subset of your data being recovered. For example, data that would normally be recovered using primary keys (RESCOPYTN) will not be recovered if the primary key does not have a secondary. Instead, you will receive message 0097 for each data set as shown in the following section:

The two data sets in the previous illustration failed because RESCOPYT was set to Y, and the primary keys they reside on **do not have a secondary key**.

Statistics

This report generates a summary of all data sets that will be recovered, along with the resources that will be required if you decide to execute RECOVER in LIVE mode.

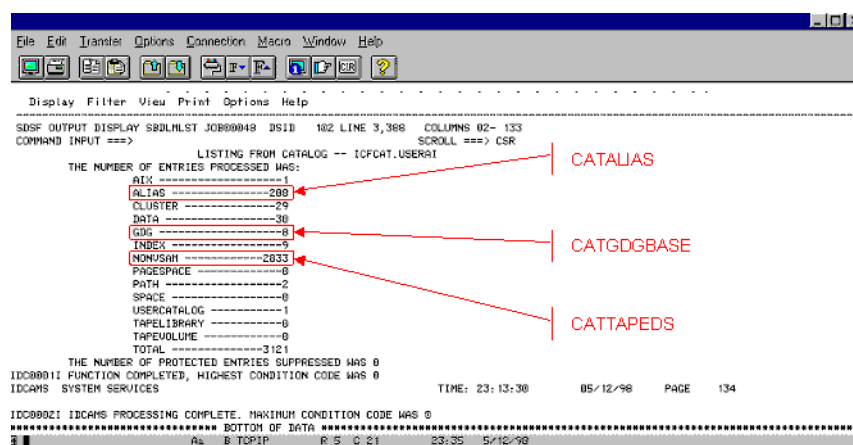


Note: The values specified for sysparms ALLOCPRI and PRIALLOC affect some of the information displayed in this report. Specifically, the total number of tracks will either increase or decrease, depending on the value specified for these sysparms.

CATALIAS

The purpose of this parameter is to assist you in recovering portions of ICF catalogs in disaster recovery situations. It recovers every alias defined in the catalog(s) that are specified on the FMS SELECT statement. For example, based on the information displayed in LISTCAT OF ICFCAT.USERAI, the following command will create a catalog at your disaster recovery site called ICFCAT.USERAI containing 208 alias entries:

```
SCAN DSNINDEX
SELECT CATALOGS=ICFCAT.USERAI
RECOVER TOVOL=vvvvvv,CATALIAS
```



The previous illustration is a sample LISTCAT OF ICFCAT.USERAI panel. All other entries (for example, AIX, CLUSTER, DATA, and so on) will not be recovered. If your intention is to recover these entries as well, you must recover the entire catalog. For details, see EXPORTF=.

CATGDGBASE

The purpose of this parameter is to assist you in recovering portions of ICF catalogs in disaster recovery situations. It recovers every GDG base catalog entry defined in the catalog(s) that are specified on the FMS SELECT statement. For example, based on the information displayed in LISTCAT OF ICFCAT.USERAI, the following command will create a catalog at your disaster recovery site called ICFCAT.USERAI containing 8 GDG bases:

```
SCAN DSNINDEX
SELECT CATALOGS=ICFCAT.USERAI
RECOVER TOVOL=vvvvvv,CATGDGBASE
```

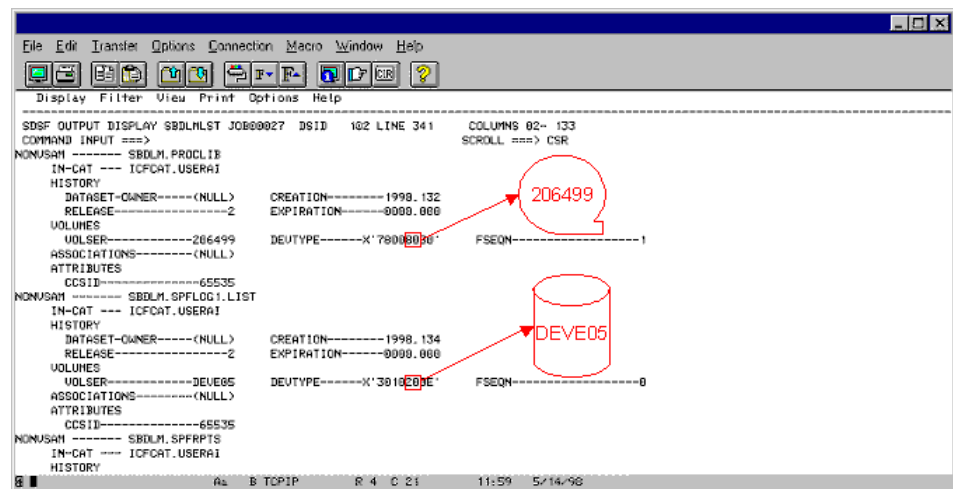
All other entries (for example, AIX, CLUSTER, DATA, and so on) will not be recovered. If your intention is to recover these entries as well, you must recover the entire catalog. For details, see EXPORTF=.

CATTAPEDS

The purpose of this parameter is to assist you in recovering portions of ICF catalogs in disaster recovery situations. It recovers every tape data set entry defined in the catalog(s) that are specified on the FMS SELECT statement.

Note: Tape data set entries are simply non-VSAM data sets that reside on tape.

When this parameter is specified, CA Disk searches through the non-VSAM candidate entries looking for 80 in the DEVTYPE field. For example, the LISTCAT shown as follows attempts to distinguish non-VSAM tape data set entries from their disk counter parts:



Only non-VSAM entries that contain '80' in their DEVTYPE field will be recovered. For example, based on the previous information, the following command will create a catalog at your disaster recovery site called ICFCAT.USERAI containing one tape data set entry:

```
SCAN DSNINDEX
SELECT CATALOGS=ICFCAT.USERAI
RECOVER TOVOL=vvvvvv,CATTAPEDS
```

Note: In the previous LISTCAT, SPFLOG1.LIST would not be recovered because of the 20 in its DEVTYPE field.

All other entries (including all remaining non-VSAM entries) will not be recovered. If your intention is to recover these entries as well, you must recover the entire catalog. For details, see EXPORTF=.

RECOVER Command Examples

Various CA Disk executions have been developed and grouped together in a separate chapter for easy reference. Several recovery scenarios executing the FMS PROC can be found in the section RECOVER Examples – FMS PROC in the chapter "Examples."

Volume Defragmentation

A common, though infrequent, task at many locations is to *defrag* a volume, which at minimum usually means to consolidate the used tracks on the pack, such that free space is in just one or two extents. To CA Disk, it means the following:

1. Consolidate used tracks to obtain large contiguous free space extents.
2. Combine multiple extent data sets into a single extent.
3. Compress PDS data sets.
4. Relocate and reorganize ISAM data sets.
5. Reorganize VSAM clusters.
6. Compress the VTOC.

A recommended approach to doing this with CA Disk is as follows:

1. Locate and format a new *mini* FDS.
2. Use it in a DSCL run that archives and scratches all data sets on the pack. Follow this with a RECOVER execution to restore all of the data sets on the volume.

The advantage of using a separate FILES for this special function is that it keeps these reorganization index entries separate from the Archive/Backup entries, and speeds the time needed to process the FILES, since it contains fewer entries.

When completed, ARCHVOLS used in the temporary FILES can be deleted from your tape management system by using the following IXMAINT command:

```
PURGE DSN=/,VOL=/
```

For more information, see the user exit description of TMSCTLEX —Tape Management System Control Interface Exit in the *Systems Guide*.

Sample Reports

This section shows sample reports for the RESTORE command and the RECOVER command.

RESTORE Command Report

The following is a sample RESTORE Report:

DATA SET NAME / NEWNAME	SOURCE DEVICE	ARCHIVED	TARGET DEVICE	DS	REC	BLKSZ	LRECL	TYPE	ALLO	---TRACKS---	MFILES
	VOLUME	TIME	VOLUME	CAT	ORG	FM			ALLO	IDLE	CONCAT
QDMTEST.SAMS.DISK0630.PS1	QA1801 3380	2239 1997382	QA1801 3380	Y	PS	FB	9640	80 TRK	1	0	1
QDMTEST.SAMS.DISK0630.PS2	QA1801 3380	2239 1997382	QA1801 3380	Y	PS	FB	9640	80 TRK	1	0	1
QDMTEST.SAMS.DISK0630.PS3	QA1801 3380	2239 1997382	QA1801 3380	Y	PS	FB	9640	80 TRK	1	0	1
QDMTEST.SAMS.DISK0630.PS4	QA1801 3380	2239 1997382	QA1801 3380	Y	PS	FB	9640	80 TRK	1	0	1
QDMTEST.SAMS.DISK0630.PS5	QA1801 3380	2239 1997382	QA1801 3380	Y	PS	FB	9640	80 TRK	1	0	1
TOTAL DATA SETS	5		TOTAL ALLOCATED TRACKS	5							
			TOTAL IDLE TRACKS								

RECOVER Command Report

In a simulation RECOVER, the column TRACKS ALLOC is an integer retrieved from the DSNINDEX record. This integer is either the data sets primary allocation size, or total allocation size, depending on the value specified for ALLOCPRI when the DSNINDEX record was created at BACKUP/ARCHIVE time.

- ALLOCPRI set to Y — the primary allocation size
- ALLOCPRI set to N — the total allocation size

For more information, see the sysparm descriptions for ALLOCPRI and PRIALLOCN in the *Systems Guide*.

DATA SET NAME / NEWNAME	SOURCE DEVICE	ARCHIVED	TARGET DEVICE	DS	REC	BLKSZ	LRECL	TYPE	ALLO	---TRACKS---	MFILES
	VOLUME	TIME	VOLUME	CAT	ORG	FM			ALLO	IDLE	CONCAT
***** THIS IS A SIMULATION RUN											
LABIUP4.DRK.TESTDSN	SMS002 3380	1832 97297	SMS002 3380	Y	PS			BLK	1	0	1
LABIUP4.DRK.TESTGDC.G0002U00	SMS002 3380	1818 97301	SMS002 3380	Y	PS			BLK	1	0	1
LABIUP4.DRK.TESTGDC.G0003U00	SMS003 3380	1816 97301	SMS003 3380	Y	PO			BLK	9	0	1
LABIUP4.ISPREB1.REGRESS.DSETR065	SMS001 3380	1818 97301	SMS001 3380	Y	PS			BLK	0	0	1
LABIUP4.KID2.TEST.PS	SMS002 3380	1811 97301	SMS002 3380	Y	PS			BLK	1	0	1
LABIUP4.TEST.SMS	SMS002 3380	0953 97182	SMS002 3380	Y	PS			BLK	2	0	1
QAFROD.DIST.LOAD	SMS901 3390	1756 97045	SMS901 3390	Y	PO			BLK	2	0	1
TOTAL DATA SETS	7		TOTAL ALLOCATED TRACKS	16							
			TOTAL IDLE TRACKS								

SMS and RESTORE/RECOVER

A key standard in an SMS environment is that all data sets must be cataloged. Therefore, whenever the Restore/Recover function is used, its catalog checking feature must be activated. This is accomplished by using the following system parameters:

RECCHCATn

The default for this system parameter is *N*. In an SMS environment it must be specified as a *Y*. This system parameter ensures that if the data set being recovered is cataloged, that it goes back to its cataloged volume. For more information, see the sysparm description for RECCHCATn in the *Systems Guide*.

RESCHCATy

The default for this system parameter is *Y*. In an SMS environment it must remain as a *Y*. This system parameter ensures that if the data set being restored is cataloged, it goes back to its cataloged volume. For more information, see the sysparm description for RESCHCATy in the *Systems Guide*.

RESCRCATn

The default for this system parameter is *N*. In an SMS environment it should be specified as a *Y*. It controls whether or not the original non-VSAM data set should be scratched if the SCRATCH parameter is present on the Restore command, and the target allocation is to a different volume. For more information, see the sysparm description for RESCRCATn in the *Systems Guide*.

CA Disk can supply the SMS construct names (Management Class, Data Class, and Storage Class) at creation time. Use the following system parameters to control this:

SMSALLOcn

The default for this system parameter is *N* which means not to supply the existing class names stored in the DSNINDEX record. If the value is changed to *Y* then at allocation time, CA Disk will supply the Management Class, Data Class and Storage Class names. The names supplied will be the ones the data set was archived or backed up with.

In most cases, except where the ACS constructs specifically review these fields during allocation, this sysparm should remain at its default value.

SMSACSDFn

When this system parameter is specified with the default value of *N*, the default class names and application ID ACS routine variables are not provided during VSAM define.

When this system parameter is specified with a value of *Y*, ACS routine variables are provided.

RESIXRPD Concern

Using this sysparm can cause adverse affects on SMS managed data sets. In situations where you don't want to keep multiple archive DSNINDEX records (that is, 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 further 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 (that is, MERGE will ignore its Management Class attributes when determining whether or not to merge the data set based upon level 1 days).

Chapter 8: VBACKUP/VRECOVER

This section contains the following topics:

[About VBACKUP and VRECOVER](#) (see page 363)

[General Information](#) (see page 363)

[VBACKUP Function](#) (see page 365)

[VRECOVER Function](#) (see page 370)

[SMS and VBACKUP](#) (see page 375)

About VBACKUP and VRECOVER

Typical CA Disk processing operates at the data set level; that is, each data set is processed as an individual logical unit, with the data being read and written in logical order. CA Disk can also process at the volume level for backup and recover processing; that is, a DASD volume is read and written starting at cylinder 0 track 0, and continuing through the end of the used cylinders.

These functions provide additional versatility to CA Disk in an online environment, and are documented in this section.

Commands available: VBACKUP and VRECOVER

JCL Proc executed DMS and RECOVER

General Information

Typical CA Disk processing operates at the data set level; that is, each data set is processed as an individual logical unit, with the data being read and written in logical order.

Volume-Level Backup

The volume-level backup function (VBACKUP) is invoked through DSCL. It will produce a backup tape or tapes containing an entire DASD volume or a group of DASD volumes, starting at cylinder 0, track 0 and continuing through all used cylinders.

Graceful Shutdown

VBACKUP jobs can be shutdown gracefully by using sysparm DSCLRPLY. A graceful shutdown will allow the jobs to complete all work in progress and produce the normal reports for error conditions encountered and work successfully completed before the jobs terminate.

With DSCLRPLY active, a message is issued to the operator at the beginning of the job and remains on the console for the duration of the job. If the operator replies END to this message, it indicates processing is to be stopped early. For details, see the section DSCL Graceful Shutdown in the chapter "Basic System – DMS PROC."

Volume-Level Recovery

The volume-level recovery function is invoked under the RECOVER procedure but through the VRECOVER command. When specified at recover time, this function uses the volume-level backup tape or tapes to overlay the entire DASD volume in extent track order, from cylinder 0 to the end of the used cylinders.

VBACKUP Function

The CA Disk volume-level backup function is invoked through DSCL. The volume-level backup function will create one backup tape (or a set of backup tapes) for each volume. Each tape or set of tapes will contain all of the data, starting at cylinder 0, track 0, and continuing through all used cylinders.

Because this is a full-volume backup function, change-bit actions are not observed, and DSNINDEX records are not written for the data sets on the volume.

CA Disk will request a new backup tape automatically if the backup processing switches to a new DASD volume. That is, if the volume-level backup includes two or more DASD volumes, the contents of the volumes are not mixed on the backup tapes.

The volume-level backup will look like one large data set to CA Disk. It will have one DSNINDEX record with an internally generated data set name of

DMS05.Vxxxxxx

where xxxxxx is the volume serial. The DSNINDEX record points to the ARCHVOLS records that identify the tape volumes used for the backup.

CA Disk generates data set names for the volume-level backup tapes in the same manner as it generates names for the data set-level backup tapes created by ARCHIVE/BACKUP; that is, sysparms ARCONAME and ARCCNAME provide the base-level name.

The DSNINDEX records for volume-level backups are flagged to indicate an extent track image backup with an x'01' in the second LISTD flag. The ARCHVOLS records for volume-level backups are flagged with an x'10' in the second LISTV flag.

If you have instructed CA Disk to use data compression during backup processing (sysparm DCDATACP specified with a value of Y), volume-level backup processing will either use the CA Disk-supplied techniques 0 or 2. The following illustration is a sample VBACKUP Command Report.

The screenshot shows a window titled 'VBACKUP Command Report' with a menu bar (File, Edit, Transfer, Options, Connection, Macro, Window, Help) and a toolbar. The main display area contains the following text:

```

SDSF OUTPUT DISPLAY ISPDLM1A JOB24368 DSD 116 LINE 1,567 COLUMNS 02- 133
COMMAND INPUT ===> SCROLL ===> CSR
1997.307 NOV 03, 1997 DATASET EXTENT REPORT FOR VOLUME SSL007 PAGE 20
MONDAY 12:55 PM CA-DISK 9.0.2
  
```

DATA SET NAME	DSO	BEG CCHH	END CCHH
VAM.VDSR530.PI043374.CNTL	PO	0018000E	00190000
VAM.VDSR530.PI043374.LKED	PO	00930000	00930004
VAM.VDSR530.PI043374.LOADLIB	PO	004C0005	004C000C
VAM.VDSR530.PI043374.MACLIB	PO	00150003	00150004
VAM.VDSR530.PI043374.OBJ	PO	0066000C	00670000
VAM.VDSR530.PI043374.SOURCE	PO	01250009	0126000A
		0047000A	0047000E
VAM.VDSR530.PI131378.CNTL	PO	004B0003	004B0005
VAM.VDSR530.PI133228.MACLIB	PO	0074000D	0074000D
VAM.VDSR530.PI135195.CNTL	PO	02A10009	02A1000A
VAM.VDSR530.PI162283.CNTL	PO	00DE000B	00DE0009
VAM.VDSR530.PI162636.LKED	PO	00ED000B	00ED0008
VAM.VDSR530.PI162636.SOURCE	PO	0084000B	00850005
VAM.VDSR530.PI162976.CNTL	PO	00DF0009	00E0000A
VAM.VDSR530.PI162978.CNTL	PO	00590006	00590006
VAM.VDSR530.PI163355.SOURCE	PO	005B000C	005B000D
VAM.VDSR530.PI163403.CNTL	PO	02160001	0216000D
VAM.VDSR530.PI210031.CNTL	PO	018C0005	018C0008
VAM.VDSR530.PI42962.CNTL	PO	059C0003	059C0005
		0A5B000B	0A5B000A

At the bottom of the window, there is a status bar showing: Aa B TCP/IP R 2 C 21 12:57 11/03/97

Offline VBACKUP

There are circumstances under which a volume must be offline to the system performing the backup. Since the volume serial number is not known to the system the unit address of the volume(s) must be specified in the UNIT= parameter of a SELECT statement. In addition, a SET statement must specify OFFLINE=INCLUDE so that offline units will be considered during the VBACKUP process. The DSNENQ parameter is ignored for an offline VBACKUP, but the RESERVE parameter is honored.

Track Overflow Data Sets

Because VBACKUP uses the EXCP access method, it is not able to back up track overflow data sets. If CA Disk encounters a track overflow data set on the volume you are backing up during VBACKUP processing, it will bypass the data set, issue a message and continue processing.

If the track overflow data set is updated following the most recent VBACKUP run, it can be restored during VRECOVER if the UPDATE parameter is specified. However, if the data set has not been updated since the last VBACKUP run, it will not be recovered during VRECOVER processing and must be restored explicitly from a previous data set-level backup copy.

VBACKUP Security Considerations

CA Disk performs all volume-level security checking prior to backing up each DASD volume. No data set-level security checking is performed. If there is not sufficient authority to backup a volume, the VBACKUP job will end and the volume will not be backed up. CA Disk will only check for volume-level authority to the volume if one of the three security interfaces is activated and sysparm SECURVOL defaults to a value of Y. Use of the VBACKUP command itself can be restricted by CA Disk. For more information on volume security, protecting CA Disk commands, and security authorization levels for VBACKUP, see the section Security Processing in the Systems Guide.

Data Integrity Considerations

If CA Disk encounters any physical I/O errors during the course of the volume-level backup, the CCHH for the track is saved. At the end of the backup processing, CA Disk writes a record to the archive tape that contains the CCHH for the tracks in error. The number of I/O errors allowed before the volume is bypassed is determined by the sysparm IOERRLIM. For detail information, see IOERRLIMnnnnn in the Systems Guide.

This record of I/O errors is prefixed with a record type of I. The DSNINDEX record Flag2 field is set to x'08', indicating a data set has been backed up with I/O errors.

ARCHIVE Management Considerations

The tapes CA Disk creates for volume-level backup are exempt from MERGE, EXTRACT, and XCOPY processing. They are handled in the same manner as normal CA Disk data set-level backup tapes for the IXMAINT, REBUILD, TAPECOPY, and REPARCH functions.

VBACKUP Command and Parameters

The VBACKUP command syntax is:

VBACKUP DSNENQ=,RESERVE=,RETPD=,EXPDT=,RPT=,LSTAR,PACK

DSNENQ=

This parameter defaults to SHR, which will issue an enqueue for every data set on the volume. If CA Disk cannot obtain an enqueue, it will continue processing, but will issue a message to identify the data set, and set the Flag1 field in the DSNINDEX record to X'01' to indicate that the volume was backed up without complete enqueue integrity. This parameter is ignored for Offline VBACKUPs and the setting of the Flag1 field in the DSNINDEX record will depend on the value of the RESERVE= parameter.

If this parameter is set to NONE, CA Disk will not issue enqueues for the data sets, but will set the Flag1 field in the DSNINDEX record to X'01' to indicate the lack of integrity.

RESERVE=

This parameter defaults to YES, which causes CA Disk to issue a hardware reserve and VTOC enqueue. If this parameter is set to NO, CA Disk will not issue the hardware reserve and the VTOC enqueue, and the DSNINDEX record Flag1 field is flagged as X'01' to indicate the lack of integrity.

RETPD=

This parameter and the EXPDT parameter are mutually exclusive and have the same meaning as the identical parameters used for data set-level processing.

Specify a one- to five-digit number to indicate a retention period to be assigned to the volume-level archive data sets. If this parameter is omitted, a default retention period value is obtained from sysparm RETRETPD.

Note: The expiration date for the ARCHVOL is determined by the sysparm DYNEXPDT, described in DYNEXPDT E99365 in the Systems Guide.

For more information, see the section Year 2000 Considerations in the chapter "General Information."

EXPDT=

This parameter and the RETPD parameter are mutually exclusive and have the same meaning as the identical parameters used for data set-level processing.

Specify an expiration date in an accepted CA Disk format to designate the period of time the data set will remain archived. If this parameter is omitted, a default retention period value is obtained from sysparm RETRETPD.

Note: The expiration date for the ARCHVOL is determined by the sysparm DYNEXPDT, described in DYNEXPDT E99365 in the Systems Guide.

For more information, see the section Year 2000 Considerations in the chapter "General Information."

RPT=

Specify this parameter with a value of MAP to produce a report for each DASD volume processed. This report will include information such as data set names and extents. If you specify a value of NONE, which is the default value, no report will be produced.

LSTAR

If this parameter is supplied, the volume-level backup processing will include only the used tracks in physical sequential and partitioned data sets, as defined by the LSTAR value in each data set's format-1 DSCB.

If this parameter is not supplied, the volume-level backup processing will include all allocated tracks on the DASD volume.

PACK

By default, VBACKUP keeps DASD volumes separated onto their own ARCHVOLS. For example, the following command will create 3 ARCHVOLS:

```
SCAN REALVOLS
  SELECT VOLUMES=(DASD01,DASD02,DASD03)
  VBACKUP RETPD=180
```

Specifying this parameter results in multiple DASD volumes being combined onto the same ARCHVOL. For example, by adding PACK to the previous command, all 3 volumes will be backed up onto the same ARCHVOL.

Note: This parameter is ignored when sysparm ARCHPACK, described in ARCHPACKy in the Systems Guide, is specified with a value of N.

VRECOVER Function

The CA Disk volume-level recovery function will overlay the entire volume in extent track order, or a specified range of absolute tracks on the volume. It is invoked from RECOVER processing through the VRECOVER command.

VRECOVER uses the volume-level backup data set identified in the DSNINDEX as DMSOS.Vvolser. CA Disk generates this name automatically.

A volume-level recovery must use a target volume that is the same device type as that from which the volume-level backup was taken. However, a single-density 3380 can be recovered to a double- or triple-density 3380, and a double-density 3380 can be recovered to a triple-density 3380. In such cases, the VTOC will be updated to invoke system routines to recalculate the free space.

If the data on the volume-level backup is compressed, it will be decompressed during VRECOVER processing using the CA Disk-supplied techniques.

Note: See Track Overflow Datasets for a description of alternate processing required to restore these data sets.

If you use the UPDATE parameter on the VRECOVER command and there is a catalog on the volume being recovered, specify the sysparm CATBKDEL with a value of Y. This allows the most current version of the catalog to be restored.

Security Considerations

CA Disk performs all volume-level security checking prior to recovering each DASD volume. No data set-level security checking is performed. If there is not sufficient authority to a volume, the VRECOVER job will end and the volume will not be recovered. CA Disk will only check for volume-level authority to the volume if one of the three security interfaces is activated and sysparm SECURVOL defaults to a value of Y.

If SECURVOL is specified with a value of N, even though the volume can be protected by your security system, CA Disk will still recover to that volume. This applies to both the target volume and the volume name specified for the IDVOL= parameter of the VRECOVER command. Use of the VBACKUP and CA Disk can restrict VRECOVER commands themselves. For more information on volume security, protecting CA Disk commands, and security authorization levels for VRECOVER, see Security Processing in the Systems Guide.

VRECOVER Command Syntax

The VRECOVER command syntax is:

VRECOVERSIM, VOLUMES=, TOVOLUME=, VERSION=, TIMEDATE=, UPDATE, DSNENQ=, RESERVE=,
GOAHEAD=, IDVOL=, CYL0TRK0=, SECERASE, ABSTRKS=, RPT=

SIM

This parameter causes processing to appear to take place; however, no data sets will be altered. Messages will be issued as if normal volume-level recover processing were taking place.

NOADMIN

Specify NOADMIN to bypass the use of Storage Administration FACILITY Class Profiles for this command. This VRECOVER command will only make use of the standard security access for the user submitting the job.

VOLUMES=

Specify for this required parameter a list of one or more volumes to be recovered. (A maximum of 50 entries is supported). If you specify more than one volume for this parameter, you cannot specify the TOVOLUME=, IDVOL= or ABSTRKS= parameters on the same VRECOVER command.

TOVOLUME=

The volume to which the volume-level recover is directed. If this parameter is not specified, CA Disk will use the volume(s) specified in the VOLUMES= parameter. If the VOLUMES= parameter specifies more than one volume and this parameter is also specified, the command will be rejected.

VERSION=

The VERSION= parameter is mutually exclusive with the TIMEDATE= parameter, and has the same meaning as defined for the RESTORE command.

TIMEDATE=

The TIMEDATE= parameter is mutually exclusive with the VERSION= parameter, and has the same meaning as defined for the RESTORE command.

UPDATE

This optional parameter indicates that any data set changed after the volume-level backup being used for this VRECOVER was taken will subsequently be updated (recovered) from its data set-level backup; that is, recover the data set from the incremental backup copy after the volume-level recovery is complete. CA Disk internally generating and executing the data set-level RECOVER command accomplishes this. The parameters that are generated for the RECOVER command are:

```
RECOVER VOL=,TOV=,DATE=,TIME=,SCRATCH,NOC,(SIM)
```

The parameter values generated for VOL= and TOV= are the same as those specified on the VRECOVER command. CA Disk obtains the DATE= and TIME= values from the DSNINDEX record of the volume-level backup being used. If the VRECOVER command is submitted with the SIMULATE parameter specified, the RECOVER command will be generated in SIMULATE mode as well.

By generating this RECOVER command, CA Disk allows you to first recover the entire volume from a volume-level backup, then follow immediately with updates from incremental backups. CA Disk will select the data sets needing to be restored based on the DATE and TIME values generated on the RECOVER command. To be effective, your installation should schedule volume-level backup runs to follow incremental backups. This is because the incremental backup turns off the change bit for updated data sets, thus reducing the number of data sets needing to be restored after the volume-level recover is complete.

DSNENQ=

The DSNENQ= parameter has a default value of EXC, which causes CA Disk to issue an exclusive enqueue for every data set on the volume. If you specify a value of NONE, no enqueues are issued for the data sets. A value of SHR causes shared enqueues to be issued.

RESERVE=

The RESERVE= parameter defaults to YES, and issues a hardware reserve and VTOC enqueue. If you specify a value of NO, no hardware reserve or VTOC enqueue are issued.

Note: Using the defaults for the DSNENQ= and RESERVE= parameters will prevent any updates to the volume during VRECOVER processing.

GOAHEAD=

This parameter controls processing only when an enqueue cannot be obtained or if the hardware reserve fails. The default value is OPERATOR, which causes a message to be issued on the console notifying the operator of the enqueue or reserve that failed, and giving the operator the options to LIST, CONTINUE or CANCEL. If the operator responds LIST, CA Disk will display at the operator's console a list of the data sets for which the enqueue or reserve failed.

If you specify a value of CONTINUE on this parameter (or if the operator replies CONTINUE), CA Disk will continue processing. If you specify a value of CANCEL (or if the operator replies CANCEL), CA Disk will issue a message and terminate VRECOVER processing.

IDVOL=

The IDVOL= parameter controls the volume name CA Disk assigns to the target volume. The default value (not specifying this parameter) causes CA Disk to retain the original name of the volume. If you specify a different volume name for this parameter, CA Disk will rename the target volume. (This is sometimes referred to as "clipping" the volume.)

This parameter applies only to a full volume-level recovery; that is, it is mutually exclusive with the ABSTRKS= parameter described in the following section.

Note: The volume name specified must not exist in the system at the time the process is initiated

CYL0TRK0=

The CYL0TRK0= parameter controls the recovery of the data records, if present, on cylinder 0, track 0. This track is a reserved track on each volume containing a pointer to the VTOC, the volume serial, and other optional data. It often includes IPL text.

When a volume-level recovery is performed, the pointer to the VTOC is always updated to reflect its location on the recovered volume. The default value of YES for this parameter causes all data records on this track to be recovered from the backups. If you specify a value of NO, only the VTOC pointer is updated (and the volume serial, if it is being renamed).

This parameter applies only to a full volume-level recovery; that is, it is mutually exclusive with the ABSTRKS= parameter described in the following section.

SECERASE

If you specify this simple parameter, CA Disk will zero all tracks on the target volume not recovered.

This parameter applies only to a full volume-level recovery; that is, it is mutually exclusive with the ABSTRKS= parameter described as follows:

ABSTRKS=

ABSTRKS= This parameter can be specified to overlay only a range of absolute tracks rather than the full volume. ABSTRKS= is mutually exclusive with the UPDATE, IDVOL=, CYLOTROK0=, and SECERASE parameters. If the VOLUMES= parameter specifies more than one volume and this parameter is also specified, the command will be rejected.

When this parameter is specified, the DSNENQ processing is also bypassed.

Specify this parameter in the following format:

ABS=(cccchhhh-cccchhhh, . . .)

Or

ABS=(cccCCCC:H-cccCCCC:H,...)

Each pair indicates a beginning and ending range of tracks in absolute address format. The range of tracks will be recovered without regard to any data set checking.

The format of cccchhhh is the format that the hardware writes the CCHH field in. For non-Extended Address Volumes (EAV), the cccc is the cylinder number (where the cylinder number is less than or equal to hexadecimal FFFF) and the hhhh portion is the head number in the range of hexadecimal 0000 to 000E. The format of cccCCCC:H is the format used to specify the logical cylinder number for both EAV and non-EAV devices. For non-EAV devices the range of value for the cccCCCC cylinder number will be from hexadecimal 0000000 to 000FFFF (from 0 to 65,535). For EAV devices, the range will be hexadecimal 0000000 to FFFFFFFF (from 0 to 268,435,455). A value larger than the device size will be rejected with a message numbered 3240. The H portion of the address is the head number and may range from hexadecimal 0 to E (0 to 14). Each address pair must use the same format but individual pairs may be of either format.

If cylinder 0 track 0 is one of the absolute tracks to be recovered, the TOVOLUME= volume serial is placed in the volume serial field before the remainder of the track is written.

RPT=

This parameter can be specified with a value of MAP to cause CA Disk to produce a VTOC MAP report at the end of the volume-level recovery. If the default value NONE is specified or the Simulate parameter is present, no report is produced.

VRECOVER Condition Codes

The following return codes are set at the completion of the VRECOVER job step:

0

A requested volume-level recovery completed successfully.

2

ICKDEF failed to rebuild the VTOC index.

4

No data sets selected; 1 or more data sets were not restored – PREALLOC =N

8

Resource not obtained; for example; enqueue, security, and dispose action.

12

Unrecoverable error in 1 or more data sets.

SMS and VBACKUP

VBACKUP uses SMS Storage Group information to determine eligibility of volumes for Volume Backup.

The SMS field that CA Disk supports is as follows:

Auto Dump (SG) is a required field that can be defined Y or N. The default value is N. When this field is defined Y, CA Disk will VBACKUP all the volumes in the Storage Group. When this field is defined N, CA Disk will not.

This example:

```
SCAN REALVOLS
      SELECT SGNAME=/  
      VBACKUP
```

Does the following:

- Select the volumes contained in all Storage Groups.
- Select the data sets meeting the appropriate Storage Group attribute.
- VBACKUP the volumes on which the data sets reside.

Chapter 9: XCOPY

This section contains the following topics:

- [About XCOPY](#) (see page 378)
- [XCOPY Limitations](#) (see page 378)
- [Basic Operation](#) (see page 380)
- [XCOPY Condition Codes](#) (see page 381)
- [Create a Secondary FILES](#) (see page 381)
- [Identify Data Sets for Disaster Recovery](#) (see page 382)
- [Determine Expiration Criteria](#) (see page 382)
- [Running XCOPY](#) (see page 382)
- [XCOPY Commands and Parameters](#) (see page 385)
- [Sample JCL](#) (see page 392)
- [Sample OUTPUT](#) (see page 393)
- [User Exits](#) (see page 393)
- [Record Formats](#) (see page 394)

About XCOPY

The CA Disk Disaster Recovery Extract Utility (XCOPY) creates ARCHVOLS from CA Disk archive and backup tapes. With its unique design, XCOPY creates ARCHVOLS in a way that will not impact your normal production backup window. These newly created ARCHVOLS are identical in file format to their archive and backup counterparts; they just reside in separate FILES.

XCOPY provides its own set of flexible commands and parameters that allow Storage Administrators to copy only a subset of archive and backup data onto the new ARCHVOLS. Selecting a special character and specifying a pattern of a data set name can select by fully qualifying their data set names, or data sets.

CA Disk data compression, activated via the DCDATACP sysparm, has no effect on XCOPY. Data is processed identically as the input volume, whether it was compressed or not.

Note: Output ARCHVOLS can be compressed using IDRC even if the input ARCHVOL was not compressed.

Finally, XCOPY has an internal mechanism that gives the Storage Administrator the ability to maintain different DSNINDEX and ARCHVOL expiration dates from those on their primary archive and backup copies.

XCOPY is an ideal complement to a Data Center's disaster recovery procedure. While running XCOPY, you will benefit in lower CPU usage compared to that of your normal archive and backup production jobs. XCOPY is designed to copy data from an archive and backup ARCHVOL onto a Disaster Recovery ARCHVOL, eliminating all data set and volume selection overhead.

XCOPY Limitations

This feature does not support archive and backup ARCHVOLS created prior to or equal to CA Disk release 7.8.8.

The Merge flag is set for all XCOPY-created ARCHVOLS. Merge can exclude these XCOPY-created tapes by using the NOMERGEIN parameter.

Not all sysparms can be used with XCOPY. The following table shows sysparms used by the XCOPY feature.

ACF2FORC	ACF2SUPP		
ARCDSKBZ	ARCDSNEX	ARCHPACK	ARCSEPC0
ARCTNAME	ARCVOLID	CARTCALC	CARTFEET

CARTPRCT	DIAGAUTH	DIAGENQF	DIAGENQV
DIAGEXCP	DIAGNOSE	DIAGSTAE	DSCBCVEX
DSCBJBNM	DSCBLMOD	DSCBLUSD	DSCBOPCD
DSCBSVMD	DYN1UNIT	DYN2UNIT	DYN3UNIT
DYN4UNIT	DYN5UNIT	DYN6UNIT	DYN7UNIT
DYN8UNIT	DYN9UNIT	DYN10UNIT	DYN11UNIT
DYN12UNIT	DYN13UNIT	DYN14UNIT	DYN15UNIT
DYN16UNIT	DYN17UNIT	DYN18UNIT	DYN19UNIT
DYN20UNIT			
DYNALDSN	DYNCUNIT	DYNEXPDT	DYNTUNIT
DYNUKEEP	DYNVREQD	ENQVOLWT	FILESPEC
FILEUNIT	FILOGNAM	FLYCOUNT	FORCPPOOL
ICHEKAEX	ICHECKPEX	IDRCSTAT	IDRCUSED
MERC1NAM	MERC1TYP		
		MERP1NAM	MERP1TYP
MERT1NAM	MERT1TYP	MERT2NAM	MERT2TYP
MERT3NAM	MERT3TYP	MERT4NAM	MERT4TYP
MERT5NAM	MERT5TYP		
MERREPLY	PASSNEWN	PASSWORD	POINTABS
RACFALLO	RACFALWZ	RACFDAEX	RACFDPEX
RACFDVOL	RACFEAEX	RACFENTY	RACFEPEX
RACFMODL	RACFNEWN	RACFPROC	RACFSUPP
RACFTEST	RACFUSID	RACGDGVL	RACHKAEX
RACHKPEX	RADELAEX	RADELPEX	RADELSVC
RESCOPYS	RESCOPYT	RSUPPRES	SACSSUPP
SECURAEX	SECURCMD	SECURFIL	SECURLIM
SECURLOG	SECURPEX	SECURRPT	SECURVOL
SECUSUPP	SORTPADD	SPACEPRI	SPACERNR
SPACESEC	STAETRAP	TAOPPOEX	TAOPPREX
TAPCTLEX	TAPEFEET	TAPEFSCR	TAPEPOOL
TAPEPROT	TAPEVVER	TMSCTLEX	TOPSALWZ
TOPSFORC	TOPSSUPP	TPMGMTX	TPOOLSEQ

TSCHKAEX	TSCHKPEX	UCBALLOC	UNLOADEX
VSSTUML	XCPARCEX	XCPDSNEX	

Basic Operation

XCOPY uses archive and backup ARCHVOLS previously created by CA Disk as input to create subsets of the data. It was developed for Data Centers that were required to create subsets of their backup data to take offsite for disaster recovery. By creating a disaster recovery ARCHVOL from a primary archive and backup ARCHVOL, copying the data can be run outside the backup time window without having to quiesce systems relying on the data.

Automatic Selection

XCOPY is an intelligent process that can automatically select only those primary archive and backup ARCHVOLS that it has not yet previously processed. This feature is controlled by the FLAG and CHKFLAG parameters. After an archive and backup ARCHVOL has been processed, it is marked as *XCOPY processed* and can be automatically skipped during subsequent XCOPY runs.

This feature allows you to set up XCOPY to run regularly and automatically generate a list of input ARCHVOLS of disaster recovery data. It will find new backup data taken during normal backup processing, select data sets to be copied for backup data sets, and copy the data.

Secondary FILES

XCOPY can be optionally set up to record all backup information into a separate files data set. Input index records are selected and located through the FILES DD statement, while output index records are recorded in the FILES through the OFILES DD statement. The data set allocated to the OFILES DD statement has the same format as the data set allocated to the FILES DD statement, and can be used by any other CA Disk function that normally uses the FILES.

The advantage of having a secondary files data set for disaster recovery is that it simplifies the recovery process. This FILES and all corresponding ARCHVOLS can be taken to a new location and used for individual data set restoration, or for full system recovery. Finally, by having a FILES that only contains information about offsite data, CA Disk will never attempt to restore data from an ARCHVOL outside the set of offsite ARCHVOLS.

Expiration Date Support

You can optionally assign special expiration dates to the disaster recovery ARCHVOLS by supplying either the RETPD= or EXPDT= parameter on the XCOPY command. For details, see EXPDT=, RETPD=.

For more information, see the section Year 2000 Considerations in the chapter "General Information."

XCOPY Condition Codes

The following condition codes are returned from the stepname XCOPY during the XCOPY process:

- 0**
Successful execution.
- 1**
Successful execution (extra copy step needed).
- 4**
Warning messages issued.
- 5**
Warning messages issued (extra copy step needed).
- 8**
Error terminated processing.
- 9**
Error terminated processing (extra copy step needed).
- 16**
Critical error caused XCOPY to terminate.

Create a Secondary FILES

Using normal CA Disk procedures, create a secondary FILES for recording disaster recovery backup information. For detailed instructions to accomplish this task, see the *Systems Guide*.

Identify Data Sets for Disaster Recovery

Prepare a list of data set names and patterns to specifically identify those data sets to be included in the disaster recovery set of data. This list of data sets will be used as selection criteria for selecting data sets.

Determine Expiration Criteria

Using normal CA Disk index maintenance planning, determine when CA Disk should expire disaster recovery data.

Running XCOPY

XCOPY uses the XCOPY PROC, which contains the basic JCL to run the utility. In addition to executing the PROC, supply the following JCL items:

- Input FILES
- Secondary FILES
- System Parameter Specifications
- SET MODE=SIMULATE (optional)
- SELECT= command to select the archive and backup data sets to be copied
- XCOPY command

System Parameter Considerations

The following system parameters are used to define output device types and ARCHVOL naming conventions. Detailed descriptions of these system parameters can be found beginning on in the *Systems Guide*.

- MERP1NAM
- MERP1TYP
- MERC1NAM
- MERC1TYP
- MERT1NAM
- MERT1TYP

Simulate Mode Considerations

By using the SET MODE=SIMULATE command in an XCOPY job, CA Disk instructs XCOPY to execute without actually mounting any ARCHVOLS or copying any DSNINDEX records. The result is a report indicating which DSNINDEX and ARCHVOL records would have been selected in a live run.

Because output devices are not allocated during simulations, CA Disk cannot accurately predict the number of output ARCHVOLS there will be in a live run. If you only want to know which DSNINDEX and/or ARCHVOL records will be selected for input by XCOPY, then this is not a concern. However, certain steps need to be taken if your intent is to determine the approximate number of ARCHVOLS that will be required for output.

XCOPY has a facility to simulate specific device types to predict volume capacity, and when a volume switch would occur. This is accomplished by specifying the appropriate DYNnUNIT/DYnnUNIT sysparm with the CA Disk value for the device type to be simulated as indicated below:

The MERP1TYP (MERC1TYP and/or MERT1TYP, if desired) sysparm should specify one of the following Media Option types:

Media Option	Sysparm in Effect
3480	DYNCUNIT
DYN1	DYN1UNIT
DYN2	DYN2UNIT
DYN3	DYN3UNIT
DYN4	DYN4UNIT
DYN5	DYN5UNIT
DYN6	DYN6UNIT
DYN7	DYN7UNIT
DYN8	DYN8UNIT
DYN9	DYN9UNIT
DY10	DY10UNIT
DY11	DY11UNIT
DY12	DY12UNIT
DY13	DY13UNIT
DY14	DY14UNIT
DY15	DY15UNIT

Media Option	Sysparm in Effect
DY16	DY16UNIT
DY17	DY17UNIT
DY18	DY18UNIT
DY19	DY19UNIT
DY20	DY20UNIT

The DYNnUNIT/DYnnUNIT sysparm, related to the previous Media Options, should be specified with one of the following cartridge device types:

Value	Device Simulated	Capacity
01600	3420 at 1600 BPI	
06250	3420 at 6250 BPI	
34800	3480 Standard	208 MB
34801	3480 Enhanced	396 MB
34900	3490 Standard	396 MB
34901	3490/Enhanced	810 MB
34902	3490/E	1.6 GB
34903	3490/E	9.5 GB
34904	3490/E	20 GB
34905	3490/E	39 GB
34906	3490/E	60 GB
34907	3490/E	200 GB
34909	3490/E	2.0 GB
34990	Redwood Cartridge	9.5 GB
34991	Redwood Cartridge	24 GB
34992	Redwood Cartridge	47 GB
35900	3590 Magstar Cartridge	9.5 GB
35901	3590 Magstar Cartridge	19 GB
35902	3590/E	30 GB
35903	3590/E	39 GB
35904	3590/E	60 GB

Value	Device Simulated	Capacity
3590C	3590/E	75 GB
35907	3590/E	100 GB
35908	3590/E	120 GB
35905	3590/E	200 GB
3590D	3590/E	240 GB
35906	3590/E	300 GB
35909	3590/E	500 GB
3590A	3590/E	700 GB
3590B	3590/E	1000 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 3480/3490 devices use CARTFEET. For more information, see the sysparm descriptions of CARTCALC, CARTPRCT, and CARTFEET in the *Systems Guide*.

To specify a unit address using the DYNnUNIT or DYnnUNIT sysparm, it must be preceded by an EBCDIC slash(/).

For example, to direct the duplex copy of an XCOPY ARCHVOL to a 25 GB Redwood cartridge device, specify:

```
//SYSPARMS DD *
MERP1TYP3480
MERC1TYPDYN1
DYN1UNIT34991
```

XCOPY Commands and Parameters

The Disaster Recovery Extract Utility uses three commands to process the data:

- SET MODE=
- SELECT
- XCOPY

SET MODE= Command

Use this command to define environment values that will be in effect for the duration of the SYSIN. You can put the SET command anywhere in the command structure, but before the first action command.

SET MODE= Command Parameters

The following parameters are available to the SET MODE= command:

SIMULATE

Specifies that this is a simulation run. In simulate mode, CA Disk will produce the normal messages and reports as if processing had taken place in LIVE mode, but will not alter any data sets.

To switch the processing environment into LIVE mode, simply comment out the SET statement by placing an asterisk (*) in column 1, or change the value to LIVE.

LIVE

This parameter is optional and the default value for the SET MODE= command. By specifying LIVE, you are instructing CA Disk to execute the XCOPY command in live mode. Datasets selected by the SELECT DSN= command will be processed according to XCOPY parameters.

SET Mode= Command Syntax

```
//SYSIN DD *  
SET MODE=parameter
```

SET Mode= Command Example

```
//SYSIN DD *  
SET MODE=LIVE
```

SELECT Command

The SELECT command indicates the archive and backup data set names to be copied from input ARCHVOLS. Any number of SELECT commands is supported.

SELECT Command Parameters

The following parameter is available to the SELECT command:

DSN= is a list of data set names to select as input to the XCOPY command. This list can contain fully qualified data set names or pattern names. You can optionally enclose the list in a single SELECT command within parenthesis. CA Disk supports a maximum of 20 entries.

SELECT Command Syntax

```
//SYSIN DD *  
SELECT DSN=(value1,value2,value3)
```

SELECT Command Example

```
//SYSIN DD *  
SET MOD=SIMULATE  
  SELECT DSN=(PROD1./,PROD3./)  
  SELECT DSN=SYSA.*.LOADLIB  
  SELECT DSN=PRODACCT.MASTER.LOADLIB  
  SELECT DSN=CHKPOINT.UPDATE.LIB
```

XCOPY Command

The XCOPY command instructs CA Disk to create a copy of selected data sets from ARCHVOLS, and store the DSNINDEX records and their associated ARCHVOL records in the secondary FILES.

Only one XCOPY command can be supplied per execution.

XCOPY Command Parameters

The following parameters are available to the XCOPY command:

TYPES=

This optional parameter can be specified to limit processing to those archive data sets residing on specified device types. You can specify TAPE, DISK, 3480, 3490, DYN1 through DY20, or any combination. If this parameter is omitted, processing defaults to all device types.

DATE=

This optional parameter can specify that a date in an accepted CA Disk format be used instead of the current date in determining which archived data sets have expired.

TAPEPULL=

This is an optional parameter. Specify TAPEPULL=GEN to generate a hardcopy list of the archive tape volumes selected for merging; that is, the input tapes that must be pulled from the tape library.

Specify TAPEPULL=USE to limit processing to those input volumes listed when the TAPEPULL=GEN parameter was specified on a previous execution. (The only additional optional parameters that are valid when processing in this mode are COPY and EXCLUDE.)

A detailed description of the TAPEPULL process can be found in the section TAPEPULL Processing in the chapter "MERGE."

TDSNAMES=

dsnamelist

This optional parameter can specify 1 to 20 archive volume data set names or patterns for which processing is to be limited.

INCLUDE=

A list of one or more archive volumes or volume patterns (or keys to disk archive data sets) to be included in the XCOPY input without regard to their eligibility for other reasons. Their presence in this list overrides the EXCLUDE parameter.

Note: To limit processing to ONLY those ARCHVOLS specified in this parameter, all other ARCHVOLS must be excluded by specifying the parameter EXCLUDE=.

EXCLUDE=

A list of one or more archive volumes or volume patterns (or keys to disk archive data sets) to be excluded from XCOPY processing (unless in an INCLUDE list). A maximum of 50 entries is supported.

FCREDT=

This optional parameter can be specified to limit the scan of ARCHVOL records to those whose creation date is greater than or equal to (from or since this creation date) the date specified via this value. The date value must be in an accepted CA Disk format. This parameter and the FDAYSOLD= parameter are mutually exclusive. Do not specify both on the same XCOPY command or the command will be rejected.

FDAYSOLD=

This optional parameter can be specified to limit the scan of ARCHVOL records to those whose creation date is greater than or equal to the current date (or an alternate date specified in the DATE= parameter) minus the number of days specified on this parameter. The maximum value for this parameter is 9999. This parameter and the FCREDT= parameter are mutually exclusive. Do not specify both on the same XCOPY command or the command will be rejected.

CREDT=

This optional parameter can be specified to limit the scan of ARCHVOL records to those whose creation date is less than or equal to the date specified via this value. The date value must be in an accepted CA Disk format. This parameter and the DAYSOLD= parameter are mutually exclusive. Do not specify both on the same XCOPY command or the command will be rejected.

DAYSOLD=

This optional parameter can be specified to limit the scan of ARCHVOL records to those whose creation date is less than or equal to the current date (or an alternate date specified in the DATE= parm) minus the number of days specified on this parameter. The maximum value for this parameter is 9999. This parameter and the CREDT= parameter are mutually exclusive. Do not specify both on the same XCOPY command or the command will be rejected.

LISTALL

Normally report records and messages generated by this function are produced only for the volumes actually processed. If you want to print statistics and messages for volumes and data set names not processed, specify this parameter on the XCOPY command.

COPY

The primary output data sets created by XCOPY can have duplicate and/or tertiary copies created concurrently, often referred to as *triplexing*. (This is controlled by the dynamic allocation SYSPARMS or by providing the proper JCL directly.) This COPY parameter indicates whether or not an additional copy is to be produced in an extra job step following the primary XCOPY.

If duplexing is performed in the primary XCOPY, this parameter causes the keys of the duplicate copies to be written to a disk data set and passed to the subsequent step where they can be copied again, creating tertiary copies. If duplexing is not done, the key of the primary (instead of the duplicate) is passed to the copy step. This parameter merely causes the keys of the appropriate new archive volumes to be written (saved) in a disk data set. You must request procedure MERGCOPY directly after the XCOPY procedure if you want to actually execute the extra copy step. An ARCHVOLS record will be made for each new copy volume and properly chained from the primary (or duplicate) ARCHVOLS record.

FLAG

FLAG is provided to indicate that the original ARCHVOLS record in the input FILES is to be flagged as processed upon completion of processing. When used in conjunction with the CHKFLAG parameter, XCOPY will automatically bypass volumes already processed in prior XCOPY executions. This will automate COPY selection of volumes so the process can be run at any time to XCOPY volumes that have been newly created since the last XCOPY operation.

CHKFLAG

Will cause the program to bypass processing for any ARCHVOL previously flagged as having been processed by the FLAG parameter.

EXPDT=, RETPD=

EXPDT and RETPD have been added to provide the ability to set the expiration date for all data sets extracted by the XCOPY function. These parameters are mutually exclusive. Whether or not these parameters are specified, ARCHVOL expiration dates will be set according to the existing rules for CA Disk MERGE processing (that is, DYNEXPDT sysparm will be used). For details, see DYNEXPDT E99365 in the *Systems Guide*.

If these output ARCHVOLS are being sent offsite and you normally use the External Data Manager for your output tape support, you can consider updating the XCOPY executing program name so that your output tapes are not managed by EDM. You can do this by updating your JCL or PROC MI= statement to specify a program name of MI=000 instead of MI=002. This runs the XCOPY PROC under the alias name of ADSMI000 and allows the output to have a valid expiration date instead of *permanent retention* if you set DYNEXPDT accordingly. This would be recommended if IXMAINT processing on the output FILES is being done at the offsite location (or not at all), or if the output tapes are not in your local tape library management system.

For more information, see the section Year 2000 Considerations in the chapter "General Information."

DEVTYPE=

By default, XCOPY processes all device types. This optional parameter can be used to limit processing to only those DSNINDEX records residing on a particular device type. Possible values are: 3380, 3390, 3480, 3490, REDW (StorageTek's Redwood device) or 3590 (IBM's Magstar device).

MAXVOLS=

This optional parameter can specify a maximum of input volumes to be processed for this execution of XCOPY, which is 999,999 volumes. Default value for this parameter is 32,767.

Multi volume archvols are a single data set, so they only count as one input volume.

VERSION0

Selects only the latest copy of the data sets selected. All other versions are ignored.

XCOPY Command Syntax

```
//SYSIN DD *
```

```
XCOPY parameter1,parameter2,parameter3, etc...
```

XCOPY Command Example

```
//SYSIN DD *
SET MOD=LIVE
  SELECT DSN=(APP1./,APP2./)
  SELECT DSN=SYSA.*.LOADLIB
  SELECT DSN=PRODACCT.MASTER.LOADLIB
  SELECT DSN=CHKPOINT.UPDATE.LIB
    XCOPY FLAG,CHKFLAG,TDSN=(APP1.BKUP/,APP2.BKUP/)
```

Ending an XCOPY Job

You can optionally specify a system parameter that allows the computer operator to terminate XCOPY prematurely (a graceful shutdown). Message 4255 is issued to the operator at the beginning of the XCOPY process. If the operator replies to the message to indicate that processing is to be stopped, XCOPY will terminate when the current input volume has completed processing. Processing completes normally, and all reports and statistics are generated as normal. This option permits you to start XCOPY and run until the resources are needed for something else. Thus XCOPY can be run in otherwise unused machine time. Specify this option through sysparm MERREPLY.

Sample JCL

The XCOPY command shown below selects all archive volumes that have not already been processed by XCOPY, and creates a second set of archive and backup data sets. The new backup data sets are stored in an ARCHVOL created in the secondary FILES allocated to the OFILES DD statement.

```
//COPYTAPE EXEC XCOPY
//XCOPY.FILES DD DSN=DISK.PRIMARY.FILES,DISP=SHR
//XCOPY.OFILES DD DSN=DISK.OFFSITE.FILES,DISP=SHR
//XCOPY.SYSPARMS DD *
MERP1NAMDMS.MERGPRI1
MERP1TYP3480
MERC1NAMDMS.MERGCOP1
MERC1TYPDISK
//XCOPY.SYSIN DD *
SELECT DSN=(PROD1./,PROD3./)
SELECT DSN=SYSA./
SELECT DSN=PRODACCT.MASTER.LOADLIB
SELECT DSN=CHKPOINT.UPDATE.LIB
  XCOPY FLAG,CHKFLAG,TDSN=(APP1.BKUP/,APP2.BKUP/)
```


Sample OUTPUT

The XCOPY job output shown below copies fifteen (15) data sets from normal archive and backup ARCHVOLS to create 1 primary and 1 copy disaster recovery ARCHVOLS. The following illustration is a sample OUTPUT from the XCOPY Command:

```

SYSDIEN ISPF2 CA11 ----- OUTPUT, Output ----- 01/23/06 20:21:43
Command ===== Scroll ===== Lvl 4 Row 1-22 Col 1-132/132
MONMA10I JOB 55123 CONAME - SYSPRINT STEPNAM - SIXCOPY PROCSTEP - PRINT
2006.023 JAN 23, 2006 X COPY DATA SET STATUS PAGE 1
MONDAY 8.21 PM CA-DISK P11.5

DATA SET NAME ----- ARCDATE KEY FILE KILO- BLOCK ACTION ----- NEW --- OUT-
                           BYTES. COUNT EXPDATE TAKEN KEY FILE DD -----
MONMA10.ANZ.BIN 2006.023 576673 1 4 1 2006.028 COPIED P00001 1 1
MONMA10.ANZ.BIN 2006.023 576673 13 4 1 2006.028 COPIED P00001 13 1
MONMA10.ANZ.REPORT1 2006.023 576673 2 43393 1587 2006.028 COPIED P00001 3 1
MONMA10.ANZ.REPORT1 2006.023 576673 2 43393 1587 2006.028 COPIED P00001 2 1
MONMA10.ANZ.REPORT2 2006.023 576673 3 23 1 2006.028 COPIED P00001 5 1
MONMA10.ANZ.REPORT2 2006.023 576673 3 23 1 2006.028 COPIED P00001 4 1
MONMA10.ANZ.REPORT3 2006.023 576673 4 54 2 2006.028 COPIED P00001 7 1
MONMA10.ANZ.REPORT3 2006.023 576673 4 54 2 2006.028 COPIED P00001 6 1
MONMA10.ANZ.REPORT4 2006.023 576673 5 32 1 2006.028 COPIED P00001 9 1
MONMA10.ANZ.REPORT4 2006.023 576673 5 32 1 2006.028 COPIED P00001 8 1
MONMA10.ANZ.REPORT6 2006.023 576673 6 54 2 2006.028 COPIED P00001 10 1
MONMA10.ANZ.REPORT6 2006.023 576673 14 54 2 2006.028 COPIED P00001 14 1
MONMA10.A300.JCLLIB 2006.023 576673 7 6 1 2006.028 COPIED P00001 12 1
MONMA10.A300.JCLLIB 2006.023 576673 7 6 1 2006.028 COPIED P00001 11 1
2006.023 JAN 23, 2006 X COPY VOLUME STATUS PAGE 1

```

User Exits

Specifics regarding the use of user exits can be found in the User Exits section of the *Systems Guide*. Only a brief description is presented here.

MERGUSEX--Merge Sort Field User Exit

Valid for MERGE

The module name that is supplied for this exit receives control from the CA Disk MERGE function after an ARCHVOLS record is selected for processing. If a multivolume ARCHVOLS entry is selected, only the first ARCHVOLS key in the chain is passed to this exit. When using the TAPEPULL processing with GEN, the first 4 bytes of the user-sort field are sequence number from the TAPEPULL list.

This exit does not accept a return code. The MERGE process cannot be altered.

Use sysparm MERGUSEX to specify the module name to invoke for this exit.

The following table lists parameters that are passed to MERGUSEX:

Type	Size	Description
INPUT	CL6	Primary ARCHVOL key
INPUT	CL6	Copy ARCHVOL key if any
INPUT	PL2	Percent used on volume

IN/OUT

PL2

User Sort field:

Where the first 4 bytes will be the
TAPEPULL sequence number.

XCPARCEXmmmmmmmmmm

This module is called during XCOPY processing to allow selection processing based on criteria other than that available through SELECT and XCOPY parameters. For details, see XCPARCEX-XCOPY Archive Volume Selection Exit in the *Systems Guide*.

XCPDSNEXmmmmmmmmmm

This module is called during XCOPY processing to allow selection processing based on criteria other than that available through SELECT and XCOPY parameters. For details, see XCPARCEX-XCOPY Archive Volume Selection Exit in the *Systems Guide*.

Record Formats

There are three Record Formats related to XCOPY. For detailed information, see Report Record Layout-(RPTNDSNX), Report Record Layout-(RPTNVOLX), or Subfile Record Format – (ARCHVOLS) in the *Systems Guide*.

Chapter 10: FILES Maintenance

This section contains the following topics:

[About Files](#) (see page 395)

[General Information](#) (see page 396)

[LISTV Command](#) (see page 399)

[LISTD Command](#) (see page 402)

[LISTF Command](#) (see page 412)

[LISTS Command](#) (see page 415)

[IXMAINT Utility General Information](#) (see page 418)

[SMS and IXMAINT](#) (see page 430)

[Archive Index Rebuild Utility](#) (see page 436)

[IXUPDATE Utility](#) (see page 441)

[Making Additional Copies of Archive Tapes](#) (see page 449)

About Files

The functions in this section provide convenient means to manage the index entries for archived data sets and the volumes that contain them.

Commands Available	JCL Proc Executed
LISTV, LISTD, LISTF, LISTS	LISTD
DSNDELETE, VOLDELETE, PURGE	IXMAINT
REBUILD	REBUILD
DELETE, CHANGE, RESET, BUILDARC	IXUPDATE

General Information

The archival and backup functions assist you in managing your online storage, but the copies in the archives (and the index entries for them) must also be managed. This section describes the utilities and procedures that provide you with the ability to do so.

CA Disk combines the data sets being archived (or backed up) into a single output data set. This output data set is in sequential format, and therefore can be placed on a standard tape, a tape cartridge, or any disk device. Since archive and backup functions have traditionally directed their output to tape, the output data set has also corresponded one for one to an output volume. This has led us to think of them as archive volumes, with CA Disk keeping track of them in an archvols index. But keep in mind that each is really just a sequential data set that can be on either disk or tape.

Whenever a data set needs to be restored, rapid access into the archvols data set is desired, regardless of whether it is on disk or tape. To provide this ability, CA Disk also maintains an index to each of the archived data sets. The index entry contains the exact location of each archived data set within the single archvols data set.

In the following discussion, any reference to an archive volume refers to an archvols data set, which can reside on either disk or tape, and can consist of backup and/or archive copies of numerous data sets.

Each data set in the CA Disk archives has its own expiration date (assigned by the requestor or an installation default), and CA Disk is designed to manage the archives by examining this date for each data set. Since the expiration date for each data set is the controlling factor, CA Disk default processing assigns all of the archive volumes (tapes or disk data sets) the never expire date of 99365. When CA Disk determines that all of the data sets on an archive volume have expired, the archive volume will be deleted/expired as well.

When a data set's expiration date is less than or equal to the current date, the data set is considered expired. Consequently, an archvols data set can contain expired as well as unexpired data sets. When an archvols expiration date has passed, all of the data sets contained within it should also have expired. At that time, or shortly thereafter, all of the corresponding entries should be removed from the CA Disk archive index. If the archvols data set was on tape, the tape should be put back into the scratch pool. If it was a disk data set, it should be deleted.

Given this situation, the following items describe the most common tasks dealing with files data maintenance:

- The most common task can be to simply list the archive volumes and/or the names of the archived data sets.
- Perhaps the most important task is to delete index records for expired data sets from the DSNINDEX subfile. This process both frees up space in the subfile and releases archive volumes when all data sets they contain have expired.
- Combine two or more partially used archvols into more fully used archvols. A partially used archvols can consist of a tape that had very little data written to it, or it can be a tape or disk archvols data set that contains many expired entries. The unexpired entries are combined onto a new archvols data set that is more fully used, freeing up either disk space or tapes to be returned to the scratch pool. Detailed information regarding this facility is contained in the chapter "MERGE."
- Delete data sets from the archives regardless of their expiration status. Normally this is done on an as-requested basis to satisfy a specific need (that is, delete obsolete data sets from the archives).
- Delete archvols from the archives, regardless of their expiration status. This can be done when a tape volume becomes lost or unusable. At the time the tape volume is removed from the archives, all index records pertaining to data sets that existed on the tape must also be deleted.
- Make a copy of an archvols data set for backup purposes. CA Disk records the presence of a copy archvols such that it can be used when a primary archvols becomes unusable. This can be done automatically by certain functions, such as Archive, Merge, Xcopy, or it can be done at any time by specific request.
- Change the status of data sets by changing control information for them in the archive index. Users may want to change the expiration date of a non-SMS data set in the archive index. It may need to be either extended or reduced.
- Rebuild index entries from an existing archive data set (on either disk or tape), into the CA Disk archive index. This can be useful in transporting data to another CA Disk system, or to recover archive index data in the event it is lost.

Maintenance Functions

The following chart indicates the CA Disk function that can be used to perform the appropriate maintenance function:

Function	PROC/PGM	Command
List the archive index entries.	LISTD	LISTD, LISTV, LISTF, LISTS
Delete data set index entries that have expired.	IXMAINT	DSNDELETE, VOLDELETE

Function	PROC/PGM	Command
Delete all index entries pertaining to expired volumes.	IXMAINT	VOLDELETE
Delete data set index entries regardless of their expiration dates.	IXMAINT IXUPDATE	PURGE DELETE
Delete all index entries pertaining to a volume, regardless of their expiration dates.	IXMAINT	PURGE
Make a copy of an archive volume.	ADSDM227	—
Recreate archive index records from data on an existing archive volume.	REBUILD	REBUILD
Change the expiration date in the index record for a data set.	IXUPDATE	CHANGE
Change the expiration date in the index record for an archive volume.	IXUPDATE	RESET
Assign a copy volume to a primary volume.	IXUPDATE	RESET
Disable an unusable archive volume.	IXUPDATE	RESET
Build a new volume index record.	IXUPDATE	BUILDARC

LISTD Condition Codes

CA Disk index entries for archive volumes and archived data sets can be listed selectively or in aggregate. In either case, the following list contains the possible condition codes that can occur.

0

All specified entries were listed okay.

4

Nothing was selected for listing.

8

One or more unique entries specified were not found.

8+

Errors during the list process.

JCL - Index Listings

Index entries for each of the archive data sets as well as the containing archive volumes can be listed with the following JCL. Each archive volume entry is unique, but multiple entries can occur for each archive data set, one for each time the data set was archived/backed up. The most recent (current) version will be listed first with the oldest version last.

```
//INDEXES EXEC LISTD
//SYSIN DD *
```

LISTV Command

ARCHVOL records created for each archive data set can be listed out with the LISTV command. By default, this command will list out the entire ARCHVOLS subfile. However, using any combination of the 3 subcommands listed as follows will allow you to list out a selective number of entries.

LISTV Command Syntax

The LISTV command syntax is:

```
LISTV VOLUME=,DSNAME=,DATE=
```

VOLUME=

By default, all archvol entries are listed. Specify a list of one or more keys (or key prefixes) of the archvol entries you want to see. For tape volumes, the key is simply the tape volser. For disk archvols, CA Disk generates the key. The format of a key used for disk archiving is presented under the sysparm description for ARCVOLID@#\$+-%&[" in the *Systems Guide*.

DSNAME=

This optional parameter restricts the list of volumes displayed to only those with data set names matching the list of names or pattern names specified. For example, DSN=DMS.BACKUPS/ would list only ARCHVOL volumes starting with these names. An ARCHVOL volume with DSN=DMS.ARCHIVES would be skipped. (A maximum of 20 entries is supported.)

DATE=

This optional parameter restricts the list of volumes displayed to only those created on the specified date. The date value can be entered in any accepted CA Disk format.

LISTV Flags

The flag fields of the archive volume index record use individual bits. These bits have the following meanings:

Byte #	Binary	Hex	Meaning
First	1000 0000	80	Volume is disabled
	0100 0000	40	Volume is a copy of another volume
	0010 0000	20	Volume is password-protected
	0001 0000	10	Volume CLOSE failed
	0000 1000	08	Volume not closed
	0000 0100	04	Volume is full (next data set did not fit)
	0000 0010	02	Volume created by MERGE or XCOPY process
	0000 0001	01	Volume written at 6250 BPI
Second	1000 0000	80	Old DYN1 flag - special allocation needed
	0100 0000	40	Old DYN2 flag - special allocation needed
	0010 0000	20	Old DYN3 flag - special allocation needed
	0001 0000	10	Volume-level backup
	0000 1000	08	IDRC data compaction
	0000 0100	04	XCOPY has processed this ARCHVOL
	0000 0010	02	ARC#BYTE 6 byte count is present
	0000 0001	01	ARCDDEVT pseudo device type present
Third	1000 0000	80	Volume is a PRIMARY2 ARCHVOL
	0100 0000	40	Volume is a FAST ARCHVOL
	0010 0000	20	Volume is a tertiary of another volume
	0001 0000	10	Reserved
	0000 1000	08	Reserved
	0000 0100	04	Reserved
	0000 0010	02	Reserved
	0000 0001	01	Reserved

For example, *A204* means primary volume is disabled, is password-protected, was created by the MERGE command, and has been processed by XCOPY.

CA Disk keeps track of the number of feet on a tape and the number of tracks on disk that have been used by the archiving process. When the size of the next data set to be archived will not fit in the remainder of the archvols data set, the archvols data set is closed and a new one allocated. The data set being archived is then written as the first data set in the new archvols data set. If a data set being archived is too large for the space on a single output volume, multiple output volumes are used. In such a case, the entry for the first volume indicates the number of volumes (in the VOL CNT FIELD), and chains the additional volumes together using their key fields.

The sequence number (SEQ NBR) is incremented for each volume in the chain.

LISTV Sample Report

The following is a sample LISTV Report:

SYSVIEW ISPF2 CA11 ----- OUTPUT, Output -----										01/05/06 15:25:59									
Command ==>>>										Scroll *==>> CSR									
-----										Lvl 4 Row 308-329 Col 1-132/132									
DB5MG014 JOB 2086 DDname - SYSPRINT Stepname - LISTD Procstep - LISTD										-----									
2006.003 JAN 03, 2006										A R C H I V E V O L U M E S									
TUESDAY 1.32 PM										PAGE 4									
										CA Disk r11.5									
ARCHIV										WRITTEN									
DATA										BLOCK									
K/M/G/T ID VOL										FLAGS									
COPY										TRUE									
SEQ										DEVICE									
KEY										KEY									
DATA SET NAME										TIME									
DATE										EXPDATE									
SETS										COUNT									
BYTES										RC CNT 1 2 3									
-----										-----									
#99610 QDMSTEST.QATEST.ARC0.DMSC.										1325 2006.003 1999.365									
1										1 3.77K									
1 000300										SMD058									
1 3390DISK										D2006003.T132535									
#99611 QDMSTEST.QATEST.ARC0.DMSC.										1325 2006.003 2006.005									
1										1 4.76K									
1 000300										SMD058									
1 3390DISK										D2006003.T132509									
#99612 QDMSTEST.QATEST.ARC0.DMSC.										1325 2006.003 1999.365									
1										1 4.63K									
1 000300										SMD058									
1 3390DISK										D2006003.T132508									
#99613 QDMSTEST.QATEST.ARC0.DMSC.										1322 2006.003 1999.365									
2										2 25.84K									
1 000300										SMD058									
1 3390DISK																			

D2006003.T132231											
#99614	QDMSTEST.QATEST.ARC0.DMSC.	1322	2006.003	2006.005	1	1	7.01K	1	400300	SMD006	1 3390DISK
D2006003.T132229											
#99615	QDMSTEST.QATEST.ARC0.DMSC.	1322	2006.003	2006.005	1	1	7.01K	1	000300	#99614 SMD069	1 3390DISK
D2006003.T132229											
#99617	QDMSTEST.QATEST.ARC0.DMSC.	1322	2006.003	1999.365	2	2	25.85K	1	000300	SMD058	1 3390DISK
D2006003.T132225											
#99621	QDMSTEST.QATEST.ARC0.DMSC.	1322	2006.003	1999.365	1	29	791.27K	1	000300	SMD058	1 3390DISK
.											

LISTD Command

DSNINDEX records created for each archive data set can be listed out with the LISTD command. By default, this command will list out the first 90 bytes of the record. However, there are 3 ways in which you can list out a selective portion of the 256 byte records:

LISTF

This command, described in LISTF Command produces CA Software defined report that lists out fields pertaining to the Format-1 date information of the archived data set.

LISTS

This command, described in LISTS Command produces CA Software defined report that lists out fields pertaining to the DFSMS information of the archived data set.

FIELDS=

This parameter of the LISTD command, described in FIELDS= allows the user to produce a customized report listing only those fields the user wants included in the report.

These index records can also be listed out interactively. This method is particularly useful for Flag Field information. For details on how to use these panels, see the section Archived Data Set Index Selection in the chapter "ISPF."

LISTD Command Syntax

The syntax of the LISTD command is as follows:

LISTD DSNNAME=, VOLUME=, DATE=, DUPLICATES, IDRC, FIELDS=

DSNAME=

By default, all archived data sets are listed. Specify a list of one or more data set names or patterns to restrict the display to those entries with matching names. (A maximum of 20 entries is supported.)

VOLUME=

This optional parameter can be used to restrict the listing of data set index entries to only those that are contained on the specified archive volumes. The list can consist of tape volume serials, keys generated for the archvol data sets written to disk, or prefixes for either of them. The list must always be for the primary volumes, even if they are disabled. (A maximum of 20 entries is supported.)

DATE=

This optional parameter restricts the list of data set index entries displayed to only those created on the specified date. The date value can be entered in any accepted CA Disk format.

DUPLICATES

This optional parameter indicates that all versions of the specified data sets are to be listed. If this parameter is not specified, only the most recent copy of the data set will be displayed.

IDRC

This optional parameter replaces the FLGS column in the report, with the IDRC column. It is mutually exclusive with the FIELDS= parameter. Values listed in this column are only produced if the data set was archived or backed up while CARTCALC was set to Y. For more details, see the sysparm description for CARTCALCn in the *Systems Guide*.

Note: These compression statistics will only appear if the ARCHVOL was created using IDRC for the given data set.

FIELDS=

This optional parameter contains the list of DSNINDEX record field names you want to appear on the LISTD report. The default value is *blank* which deactivates the parameter, resulting in the LISTD report retaining its original format. When a list of field names is specified, they should be specified in the column sequence desired.

A maximum of 50 fully qualified field names is support, and their combined output length must be less than or equal to 132 characters.

LISTD Example Using FIELDS= Option of LISTD Command

For example, if you execute the following command:

```
//jobname JOB (acct,info),etc.
//LIST EXEC LISTD
//SYSIN DD *
LISTD FIELDS=(DSNAM,VOLNO,ARCDT,ATIME,FLAG3B,DSORG,FLGSC,FLGUC)
/*
```

The LISTD report would print out as follows:

2003.113 APR 23, 2003	DATA SET NAME INDEX							PAGE 1
WEDNESDAY 3.33 PM								CA Disk 9.0.2
	SOURCE	ARCHIVE	ARCH	FLAG 3				
DATA SET NAME	VOLUME	DATE	TIME	12345678	DSORG	SCR	UNCAT	
-----	-----	-----	----	-----	----	----	----	
USERA10.BASE90.PARMLIB	SMD006	2003.036	1528	X...X...	P0			
USERA10.CHANGE.LOG	SMD040	2003.036	1529	X...X...	PS			
USERA10.CLIST	TSU013	2003.036	1529	X...X...	P0	Y		
USERA10.DDIR	TSU004	2003.036	1529	X...X...	AM			
USERA10.DEMOED.ISPPROF	DEMS04	2003.036	1529	X...X...	P0			
USERA10.DISK.ANZ.FILES	TSU002	2003.036	1529	X...X...	PS			
USERA10.DISK.AUTOLINK	TSU019	2003.106	1351	X...X...	AM		Y	
USERA10.DISK.CSLVL2.FILES	TSU039	2003.036	1529	X...X...	PS			
USERA10.DISK.ESDS	SMD006	2003.106	1351	X...X...	AM			
USERA10.DISK.FILES	SMD056	2003.036	1529	X...X...	DA			
USERA10.DISK.JCL	SMD041	2003.036	1529	X...X...	P0			
USERA10.DISK.LOADLIB.OLD	SMD060	2003.036	1529	X...X...	P0	Y	Y	
USERA10.DISK.NODMS.MITABLE	SMD060	2003.036	1529	X...X...	PS			
USERA10.DISK.PANELS	SMD042	2003.036	1529	X...X...	P0			
USERA10.DISK.PARMLIB	SMD067	2003.036	1529	X...X...	P0			
USERA10.DISK.PRODSP.FILES	TSU001	2003.036	1529	X...X...	PS			
USERA10.DISK.R9B.FILES	SMD063	2002.315	1526	X...X...	PS			

USERA10.DISK.R9C.FILES

SMD007 2002.315 1526 X...X... DA

Every field contained in the DSNINDEX record is available for selection. What this means is that even though the LISTS and LISTF commands are still valid, their usefulness has been significantly reduced.

For additional information regarding the DSNINDEX record format, see Subfile Record Format —(DSNINDEX) in the *Systems Guide*.

Possible Values for LISTD FIELDS= Parameter

The following table is a list of field names available to you, in DSECT order of the DSNINDEX record:

DSNINDEX Field	FIELDS= Value	Field Description
DSNALSTN	ALSTN	Userid of last update
DSNARCDT	ARCDT	Date on which the data set was archived
	ATMDT	Combination of ATIME and ARCDT fields
DSNAREL#	AREL#	CA Disk release number at time of archive
DSNARKEY	ARKEY	Key to ARCHVOLS record
	FLAG1B	FLAG1 in binary format
	FLAGS	Flags 1, 2,3,4, and 6 in Hexadecimal
	FL126	Flags 1, 2, and 6 in Hexadecimal
DSNATIME	ATIME	Time on which the data set was archived
DSNBLKCO	BLKCO	block count
DSNBLKCT	BLKCT	Large block count (release 8.2 and above)
DSNBLKFG	BLKFG	FLAG3 BIT 0—Large Block Count
DSNBLKNO	BLKNO	First block number and PRV for 3480's
DSNBLKSI	BLKSI	Block Size Value
DSNBSAM	BSAM	FLAG3 BIT 2—BASM Backup
DSNBYTES	BYTES	Byte count
DSNBYTFG	BYTFG	FLAG3 BIT 4—Number Bytes Available
DSNDEVCL	DEVCL	Device class
DSNDRBLK	DRBLK	Directory block count
DSNDS1CD	DS1CD	Original CREDIT from format-1 DSCB

DSNINDEX Field	FIELDS= Value	Field Description
DSNDS1ED	DS1ED	EXPDT from format-1 DSCB
DSNDS1JN	DS1JN	JOBNAME or accounting code from format-1
DSNDS1MD	DS1MD	MODDT from format-1 DSCB
DSNDS1UD	DS1UD	USEDT from format-1 DSCB
DSNDSKTT	DSKTT	TT of DISK Control Record
DSNDSNAM	DSNAM	Fully qualified data set name
	DSNAMS	First 28 characters of data set name
DSNDSORG	DSORG	Data set organization from format-1 DSCB
DSNEXPDT	EXPDT	Expiration date for DSNINDEX record
DSNEXPNS	EXPNS	SMSFL BIT 5—Date Flag Altered
DSNFILNO	FILNO	File number on the ARCHVOL
DSNFLAG1	FLAG1	FLAG BYTE 1
DSNFLAG2	FLAG2	FLAG BYTE 2
DSNFLAG3	FLAG3	FLAG BYTE 3
DSNFLAG4	FLAG4	FLAG BYTE 4
	UNCAT	FLAG4 BIT 0—WAS NOT CATLGED
	LOADL	FLAG4 BIT 1—PDSE PROGRAM LIBRARY
	HFS	FLAG4 BIT2—HFS
	FAT	FLAG4 BIT3—VBACKUP FAT DASD
	DSS	FLAG4 BIT 5—BACKUPCC/DBACKUP IO path
	CC	FLAG4 BIT 6—Concurrent Copy
	FLAG5B	FLAG5 in binary format
DSNFLAG5	FLAG5	FLAG BYTE 5
		FLAG5 Bit 7—USEDSSIO dumped data set
	FLAG6B	FLAG6 in binary format
DSNFLAG6	FLAG6	FLAG BYTE 6
	FLGP2	FLAG6 Bit 0—PRIMARY2 RECORD
	RENM	FLAG6 Bit 0—DSNINDEX Renamed
	FAST	FLAG6 Bit 2—Fast DSNINDEX Record
DSNFLAG7	FLAG7	FLAG BYTE 7

DSNINDEX Field	FIELDS= Value	Field Description
	DEVP	FLAG7 Bit 0—Device Flag Present
	32BID	FLAG7 Bit 1—32 bit Blockid
DSNFLGAR	FLGAR	FLAG1 BIT 3—Explicit Archive
DSNFLGAX	FLGAX	FLAG2 BIT 2—VSAM AIX
DSNFLGBK	FLGBK	FLAG1 BIT 5—Incremental Backup
DSNFLGBT	FLGBT	FLAG2 BIT 6—RACF Bit
DSNFLGCB	FLGCB	FLAG2 BIT 3—Catalog Backup
DSNFLGEN	FLGEN	FLAG2 BIT 0—Generic Profile
DSNFLGET	FLGET	FLAG2 BIT 7—Track Image
	FLAG3B	FLAG3 in binary format
DSNFLGEX	FLGEX	FLAG1 BIT 4—Low level I/O
DSNFLGIO	FLGIO	FLAG2 BIT 4—Had I/O errors
DSNFLGPE	FLGPE	FLAG2 BIT 5—PDSE
DSNFLGRC	FLGRC	FLAG1 BIT 2—Recat to pseudovol
DSNFLGRF	FLGRF	FLAG2 BIT 1—Discrete Profile
DSNFLGSC	FLGSC	FLAG1 BIT 0—Scratched
DSNFLGUC	FLGUC	FLAG1 BIT 1—Uncataloged
DSNFLGUV	FLGUV	FLAG1 BIT 6—PS as DA
DSNFLGVV	FLGVV	FLAG1 BIT 7—From Virtual Vol
	FLAG2B	FLAG2 in binary format
DSNFNFCT	FNFCT	File Number
DSNGARCD	GARCD	GMT Archive Date
DSNGARCT	GARCT	GMT Archive Time
DSNLCOPY	LCOPY	Last Backup Copy Detected Date
DSNMCBYP	MCBYP	SMSFL BIT 4—Bypassed Mgmt Class
DSNMEMLA	MEMLA	FLAG3 BIT 3—Member Level Backup
DSNMMBRS	MMBRS	Member count
	DIRECT	Combination of DRBLK and MMBRS fields
DSNMSPFL	MSPFL	MSP flags

DSNINDEX Field	FIELDS= Value	Field Description
DSNPRTK	PRTK	First extent or total allocation (For further information, see the sysparm description for ALLOCPRI in the <i>Systems Guide</i>)
	ALLOC	Combination of SCALO and PRTK fields
DSNPRVC	PRVC	PRV for Copy
DSNRATIO	IDRC	IDRC Compression ratio
	MSPFLB	MSP Flags
DSNRELBK	RELBK	First relative block number
DSNRTCNT	RTCNT	Count of restores
DSNRDTE	RTDTE	Date of last restore
DSNRTIME	RTIME	Last restore time
DSNSCALO	SCALO	Type of allocation
DSNSEXPT	SEXPT	SMSFL BIT 0—ARCDT Altered
DSNSMSDC	SMSDC	SMS Data Class
DSNSMSFL	SMSFL	SMS flags
DSNSMSFL	SMSFLB	SMS Flags
DSNSMSMC	SMSMC	SMS Management Class
DSNSMSSC	SMSSC	SMS Storage Class
DSNSMSSG	SMSSG	SMS Storage Group
DSNSTRIP	STRIP	FLAG3 BIT 1—Extended Sequential
DSNUPDAT	UPDAT	SMSFL BIT 6—ARCDT
DSNUSERF	USERF	User Field
DSNVERS	VERS	Version Number
DSNVOLNO	VOLNO	Volser from which data set was archived

LISTD Flags

The flag fields of the data set index record use individual bits to reflect data set attributes.

Note: This Flag information can also be listed out interactively. For details on how to use these panels, see the section Archived Data Set Index Selection in the chapter "ISPF."

The flag field within the DSNINDEX record uses individual bits. These bits have the following meanings:

Byte #	Binary	Hex	Meaning
First	1000 0000	80	Data set was scratched at time of archival.
	0100 0000	40	Data set was uncataloged at time of archival.
	0010 0000	20	Data set was recataloged to CA Disk pseudo-volume.
	0001 0000	10	Archived due to EXPLICIT request. Note: If this bit is off, Archived by RETAIN
	0000 1000	08	Data set was archived via EXCP support.
	0000 0100	04	Archived with Incremental Backup
	0000 0010	02	Data set is DSORG=DA RECFM=U/V, or is DSORG=PS RECFM=U/V and was backed up as DA.
	0000 0001	01	Data set resided on Virtual Volume
Second	1000 0000	80	Data set was protected by RACF when backed up or archived, but not by a discrete profile. Data set can be protected by a generic profile, or the data set's high-level-qualifier is the same as the userid of the task that backed up or archived the data set. Release 7.0 and up only.
	0100 0000	40	CA Disk has a copy of the RACF discrete profile that protected the data set when it was backed up or archived.
	0010 0000	20	A VSAM alternate index that was processed separately from the base cluster.
	0001 0000	10	Data set is an ICF catalog backup. Release 7.0 and up only.
	0000 1000	08	Data set was backed up with I/O errors. Set by Release 7.8 and up only.
	0000 0100	04	DSORG is PDSE (Partitioned Extended)

Byte #	Binary	Hex	Meaning
	0000 0010	02	Data set was RACF-indicated when archived or backed up. Release 7.0 and up only.
	0000 0001	01	Extent track image backup. Release 7.8 and up only.
Third	1000 0000	80	New DSNBLKCT field has been updated
	0100 0000	40	DSORG is PSE (Extended Sequential
	0010 0000	20	PSE data set archived with BSAM I/O
	0001 0000	10	Member level Archive
	0000 1000	08	DSN number byte is present
Fourth	1000 0000	80	Data set was not cataloged at Backup time
	0100 0000	40	Data set is a PDSE Program Library
	0010 0000	20	DSN is HFS data set
	0001 0000	10	VBACKUP fat DASD 65,536+ tracks
	0000 0100	04	DF/SMSdss dumped data set
	0000 0100	04	BACKUPCC/DBACKUP I/O PATH
	0000 0010	02	Data set processed with Concurrent Copy
Fifth	0000 0011	03	PDSE processed using FAMS dump
	0000 0010	02	PDSE processed using FAMS unload
	0000 0001	01	New DF/SMSdss backup format
	0000 0001	01	USEDSSIO-Dumped data set
Sixth	1000 0000	80	PRIMARY2 record
	0100 0000	40	Data set in DFNINDEX was renamed
	0010 0000	20	Fast Archive
	0001 0000	10	Data set archived with NEWNAME or NEWHLQ
Seventh	1000 0000	80	Device Flags present
	0100 0000	40	PRV flag for primary
Eighth			(Not used at this time)
Ninth			(Not used at this time)
Tenth			(Not used at this time)

LISTD Sample Report

The following is a sample LISTD Report:

SYSVIEW ISPF2 CA11 ----- OUTPUT, Output -----										01/05/06 15:23:56									
Command ==>										Scroll *==> CSR									
-----										Lvl 4 Row 175-196 Col 1-132/132									
DB5MG014 JOB 2086 DDname - SYSPRINT Stepname - LISTD Procstep - LISTD										-----									
2006.003 JAN 03, 2006										PAGE 1									
TUESDAY 1.32 PM										CA Disk r11.5									
SOURCE DEV DS ALLOCATION DIRECTORY WRITTEN ARCHIV FIRST BLOCK K/M/G/T FLAGS																			
DATA SET NAME VOLUME TYPE ORG TYPE QTY BLKS MBRS TIME DATE EXPDATE KEY BLOCK COUNT BYTES 1 2 3 4 6																			
-----										-----									
DSKAM006.WORK.FILE SMD053 3390 PS TRK 1 0 0 1322 2006.003 2006.033 #99615										1 1 7.01K D800880000									
NEWHLQ.DB5BK002.WORK.PS2 SMDN01 3390 PS TRK 1 0 0 1313 2006.003 2006.004 #99658										1 1 18.37K 1800880000									
NEWHLQ.WORK.DB5AM002.PS2 SMD003 3390 PS TRK 1 0 0 1313 2006.003 2006.003 #99660										1 1 18.81K D800880000									
QAPROD.DB5BK022.DUMMY.PDS SMD006 3390 P0 CYL 1 5 1 1325 2006.003 2006.007 #99610										1 1 3.77K 1800880000									
QAPROD.DB5RR034.LOADLIB SMD006 3390 P0E CYL 1 0 0 1314 2006.003 2006.005 #99655										1 2 32.03K 0004880000									
QAPROD.DB5RR035.LOADLIB SMD043 3390 P0E CYL 1 0 0 1053 2006.003 2006.005 #99682										1 2 40.40K 0004880000									
QAPROD.P11287.DBORR018. SMDN03 3390 PS TRK 1 0 0 1325 2006.003 2006.004 #99612										1 1 4.63K 1800880000									
TESTDATA																			
QDMSTEST.DB5AM000.WORK.PDS1 SMD048 3390 P0 TRK 2 1 1 1313 2006.003 2006.003 #99664										1 1 5.22K D800980000									
QDMSTEST.DB5AM000.WORK.PS SMD047 3390 PS TRK 1 0 0 1312 2006.003 2006.003 #99671										1 1 5.05K D800880000									
QDMSTEST.DB5AM009.R90012.PS SMD048 3390 PS TRK 1 0 0 1312 2006.003 2006.008 #99669										1 1 5.24K D800880000									
QDMSTEST.DB5AM010.PSETSTB SMD059 3390 PS CYL 1 0 0 1312 2006.003 2006.008 #99668										1 15 396.89K C800880000									
QDMSTEST.DB5AM010.PSETST1 SMD059 3390 PS CYL 1 0 0 1313 2006.003 2006.008 #99668										16 1 26.70K C800880000									
QDMSTEST.DB5BK001.WORK.PDS1 SMDN02 3390 P0 TRK 2 1 1 1315 2006.003 2006.004 #99657										1 1 4.75K 1800980000									
QDMSTEST.DB5BK001.WORK.PDS2 SMDN02 3390 P0 TRK 2 1 1 1315 2006.003 2006.004 #99657										2 1 4.76K 1800980000									



LISTD Report Field Description

The LISTD ALLOC QTY field can be first extent tracks (or cylinders) or total tracks (or cylinders), depending on the specification of sysparm ALLOCPRI. For further information, see the sysparm description for ALLOCPRIy in the *Systems Guide*. The LISTD report field labeled BLKS MBRS/MULTI-VOL indicates:

- If the data set is a PDS, this field contains the number of directory blocks and the number of members in the PDS, except in those cases where the data set was archived using DFSMSdss as the data mover. In which case, the literal DSS-DUMPED will appear in the field.
- If the data set is a multivolume data set (not a PDS), this field contains the number of volumes from which the data set was archived.
- Any other conditions result in this field being blank.

The LISTD report field labeled FIRST BLOCK indicates the block number (from the beginning of the archive data set) where the data set begins.

The LISTD report field labeled BLK CNT (data set Name Index report) or BLOCK NAME (Archive Volumes report) indicates the number of blocks used to contain the data set.

The LISTD report field labeled K/M/G/T represents the actual number of kilobytes, megabytes, gigabytes, or terabytes written. If compression is used, this represents the amount used after the data set is compressed.

LISTF Command

Index records are created for each archive data set. The first 90 bytes of the 256-byte record can be listed with the LISTD command. Information gathered from the format 1 of the data set at archive time, and other additional fields can be listed with the LISTF command.

LISTF Command Syntax

The LISTF command syntax is:

LISTF DSNAME=, VOLUME=, DATE=, DUPLICATES

DSNAME=

By default, all archived data sets are listed. Specify a list of one or more data set names or patterns to restrict the display to those entries with matching names. (A maximum of 20 entries is supported.)

VOLUME=

This optional parameter can be used to restrict the listing of data set index entries to only those that are contained on the specified archive volumes. The list can consist of tape volume serials, keys generated for the archvol data sets written to disk, or prefixes for either of them. The list must always be for the primary volumes, even if they are disabled. (A maximum of 20 entries is supported.)

DATE=

This optional parameter restricts the list of data set index entries displayed to only those created on the specified date. The date value can be entered in any accepted CA Disk format.

DUPLICATES

This optional parameter indicates that all versions of the specified data sets are to be listed. If this parameter is not specified, only the most recent copy of the data set will be displayed.

LISTF Report Field Descriptions

The LISTF CREATE, LAST MOD, LAST USED, and EXPIRE fields are those dates from the format-1 DSCB of the data set that was archived or backed up.

The LISTF RES CNT field shows the number of times the data set has been restored.

The LISTF USER FIELD displays in character format the contents of the DSNINDEX user field. Update this field by using the DINXUFEX user exit as documented in DINXUFEX —Update DSNINDEX User Field in the *Systems Guide*.

The LISTF UPDATE USERID field is either:

- The USERID of the creating or using jobname
- Accounting code as defined in the FMT1 DSCB prior to backup or archive of the data set.

For more information on this field, see Installing the CA Disk SVC in the *Installation Guide*.

LISTF Sample Report

The following is a sample LISTF Report:

LISTS Command

The LISTS command is used to list SMS data set information that is stored in the index record if a data set is SMS managed.

SMS Information Within The Archives

The LISTS report will scan the DSNINDEX subfile of the CA Disk FILES and list out data sets along with their respective Management Class, Storage Class, Data Class and Storage Group names.

LISTS Command Syntax

The LISTS command syntax is:

```
LISTS DSNAME=, VOLUME=, DATE=, DUPLICATES
```

DSNAME=

By default, all archived data sets are listed. Specify a list of one or more data set names or patterns to restrict the display to those entries with matching names. (A maximum of 20 entries is supported.)

VOLUME=

This optional parameter can be used to restrict the listing of data set index entries to only those that are contained on the specified archive volumes. The list can consist of tape volume serials, keys generated for the archvol data sets written to disk, or prefixes for either of them. The list must always be for the primary volumes, even if they are disabled. (A maximum of 20 entries is supported.)

DATE=

This optional parameter restricts the list of data set index entries displayed to only those created on the specified date. The date value can be entered in any accepted CA Disk format.

DUPLICATES

This optional parameter indicates that all versions of the specified data sets are to be listed. If this parameter is not specified, only the most recent copy of the data set will be displayed.

LISTS Flags

The flag fields of the DSNINDEX record use individual bits to reflect data set attributes. The flag fields that pertain to the LISTS command are:

Binary	Hex	Meaning
1000 0000	80	ARCDT altered due to last backup copy (pre 8.2)
0000 1000	08	Management Class bypassed
0000 0100	04	Altered date field (8.2 and above)
0000 0010	02	Last backup copy and DSNLCOPY updated to indicate data set was deleted (8.2 and above)
0000 0001	01	Guaranteed backup flag

LISTS Report Field Descriptions

The LISTS report displays information from the DSNINDEX record pertaining to DFSMS.

LISTS Sample Report

The following is a sample LISTS Report:

SYSVIEW ISPF2 CA11 ----- OUTPUT, Output ----- 01/05/06 15:25:12									
Command ==>					Scroll *==> CSR				
----- Lvl 4 Row 44-65 Col 1-132/132									
DB5MG014 JOB 2086 DDname - SYSPRINT Stepname - LISTD Procstep - LISTD									

2006.003 JAN 03, 2006 S M S D A T A S E T N A M E I N D E X PAGE 1									
TUESDAY 1.32 PM CA Disk r11.5									

SOURCE WRITTEN ARCHIVE DATA MANAGMENT STORAGE STORAGE SMS									
DATA SET NAME VOLUME TIME DATE EXPDATE KEY CLASS CLASS CLASS GROUP FLGS									

DSKAM006.WORK.FILE SMD053 1322 2006.003 2006.033 #99615 DEFAULT DEFAULT SMDPOOL SMDVOLS 08									
NEWHLQ.DB5BK002.WORK.PS2 SMDN01 1313 2006.003 2006.004 #99658									

NEWHLQ.WORK.DB5AM002.PS2	SMD003	1313	2006.003	2006.003	#99660	DEFAULT	DEFAULT	SMDPOOL	SMDVOLS	08
QAPROD.DB5BK022.DUMMY.PDS	SMD006	1325	2006.003	2006.007	#99610	DEFAULT	DEFAULT	SMDPOOL	SMDVOLS	08
QAPROD.DB5RR034.LOADLIB	SMD006	1314	2006.003	2006.005	#99655	DEFAULT	DEFAULT	SMDPOOL	SMDVOLS	08
QAPROD.DB5RR035.LOADLIB	SMD043	1053	2006.003	2006.005	#99682	DEFAULT	DEFAULT	SMDPOOL	SMDVOLS	08
QAPROD.P11287.DB0RR018.	SMDN03	1325	2006.003	2006.004	#99612					
TESTDATA										
QDMSTEST.DB5AM000.WORK.PDS1	SMD048	1313	2006.003	2006.003	#99664	DEFAULT	DEFAULT	SMDPOOL	SMDVOLS	08
QDMSTEST.DB5AM000.WORK.PS	SMD047	1312	2006.003	2006.003	#99671	DEFAULT	DEFAULT	SMDPOOL	SMDVOLS	08
QDMSTEST.DB5AM009.R90012.PS	SMD048	1312	2006.003	2006.008	#99669	DEFAULT	DEFAULT	SMDPOOL	SMDVOLS	0C
QDMSTEST.DB5AM010.PSETSTB	SMD059	1312	2006.003	2006.008	#99668	DEFAULT	DEFAULT	SMDPOOL	SMDVOLS	0C
QDMSTEST.DB5AM010.PSETST1	SMD059	1313	2006.003	2006.008	#99668	DEFAULT	DEFAULT	SMDPOOL	SMDVOLS	0C
QDMSTEST.DB5BK001.WORK.PDS1	SMDN02	1315	2006.003	2006.004	#99657					
QDMSTEST.DB5BK001.WORK.PDS2	SMDN02	1315	2006.003	2006.004	#99657					
.										

IXMAINT Utility General Information

This utility provides the capability to process the entire CA Disk archive data set name index and delete specified records. It can delete both expired and unexpired index entries. Through command parameters, you can restrict it to processing the index entries for only certain specified data sets.

Used as a general maintenance tool, IXMAINT can be run on a daily basis with preset commands to perform periodic cleanup of the index files, or it can be used to accomplish specific tasks on an as-requested basis. Its main objective is to delete index entries for unwanted data sets. (If the index entry being deleted is the last copy of an archived data set, or is SMS-managed, the catalog entry can also be removed. For details, see the sysparm description for UNCATPSUn in the *Systems Guide*.)

If during processing only one index entry is remaining and the data set does not exist online, CA Disk can issue a TSO message. This message warns that only one copy of the data set is left. The message is sent to the userid that archived or backed up the data set, or, if the index entry expiration date has been updated, to the userid that last updated the index entry. The issuance of the TSO message is controlled by the sysparm IXUSRNOT, described in IXUSRNOTn in the *Systems Guide*.

Note: This message will be issued for every IXMAINT command that detects the condition, even in simulate mode.

This utility deals with the DSNINDEX subfile of the Files Data Set. There is interaction with the ARCHVOLS subfile so that archive tapes or archive data sets on disk will be released from the archives when there are no longer any DSNINDEX records that see it.

ENQUEUE RESERVE

To ensure FDS integrity when run in a multiple CPU environment, a RESERVE is done against the volume containing the data set. This RESERVE is done for the duration of the step. For this reason, it can be desirable to schedule the run during low-activity times. Also consider placing the data set on a volume that will not be in use by other CPUs during this activity.

For more information, see the sysparm description for IXMQTIME0000 in the *Systems Guide*.

Note: You can override this RESERVE by specifying sysparm RSUPPRES, described in RSUPPRESn in the *Systems Guide*. However, caution should be taken to ensure that a cross system enqueue package is active, especially if the data set resides on a shared DASD.

Tape Management Support

IXMAINT supports tape management systems as follows:

- With the appropriate interface module specified as the value to sysparms TMSCTLEX and/or TAPCTLEX
- By controlling the catalog status of the ARCHVOL
- With GDG, TAPEPOOL, and EDM managed tape expiration.

For important information, review the following areas:

- Customizing the CA Disk Tape Management Support in the *Installation Guide*.
- Method 1 —Controlling Tapes Via the EDM in the *Installation Guide*.
- Tapepool Considerations in the *Systems Guide*.
- Sysparm description for ARCTNAMEd in the *Systems Guide*.
- User exit descriptions for TMSCTLEX —Tape Management System Control Interface Exit in the *Systems Guide*.

Graceful Shutdown

IXMAINT jobs can be shutdown gracefully by using sysparm IXMREPLY. A graceful shutdown will allow the jobs to complete all work in progress and produce the normal reports for error conditions encountered and work successfully completed before the jobs terminate. With IXMREPLY active, a message is issued to the operator at the beginning of the job and remains on the console for the duration of the job. If the operator replies, END to this message, it indicates processing is to be stopped early. When using an FDB, an inquiry is made to validate the ARCHVOL record is empty (has no DSNINDEX records) so it can be deleted even when the IXMAINT is shut down. However when using an Files Data Set, all the DSNINDEX records must be processed before being able to delete an ARCHVOL record. So, when using the shut down option for an FDS, no ARCHVOLs will be deleted.

Condition Code Settings

The following list contains possible condition codes that can be issued while executing the IXMAINT Proc.

0

Successful completion.

4

Informational and/or warning messages have been issued.

8

Critical errors encountered, processing terminated.

16

Command syntax error, processing terminated.

SMS Considerations

When using the VOLDELETE and PURGE commands the utmost care should be taken. DATA LOSS can occur if used improperly.

The VOLDELETE command does not issue normal SMS checks for individual data sets and will expire an ARCHVOL based solely on the ARCHVOL expiration date. This expiration date is set based on the value of sysparm DYNEXPDT, described in DYNEXPDT99365 in the *Systems Guide*.

The PURGE command deletes unexpired data sets even if they are SMS-managed.

JCL - IXMAINT

Use the JCL below as an example for index maintenance commands.

```
//S1      EXEC  IXMAINT  
  
//SYSIN   DD   *  
(commands go here)
```

Deleting Expired Data Sets

The DSNDELETE command causes all expired data sets in the archives to be deleted from the DSNINDEX. If you have specified sysparm UNCATDEL with a value of Y, it also deletes all data sets in the archives that were archived to an expiration date of Julian date 1999.000 and which are no longer cataloged.

Note: If the catalog status of a data set cannot be determined because a catalog is either damaged or offline, no action will be taken.

The entire index is scanned for expired data sets. If all data sets for an archive volume are deleted through processing, the ARCHVOLS record will also be deleted and the volume will be released from the archives.

In addition, CA 1 Tape Management, EPIC, RMM, or Control-T (CTT) customers should read the section Method 1 - Controlling Tapes through the EDM in the *Installation Guide*.

If it is determined that no DSNINDEX records exist for a given ARCHVOLS record, a flag bit is checked to ensure that the volume has been closed. If the volume is still flagged as OPEN, it will not be deleted unless the creation date is less than today's date minus one (that is, CREDIT TODAY-1). This check maintains integrity even if ARCHIVE and IXMAINT are running concurrently over midnight.

For more information, see the sysparm description for IXMCLEANy in the *Systems Guide*. This sysparm is used as a general cleanup tool to correct pointer errors that exist in the FILES.

Note: For CA TLMS users, a JCL keyword can be overwritten by a specific RMF rule. For more information, see *CA TLMS User Guide*.

DSNDELETE Command

The DSNDELETE command syntax is:

```
DSNDELETE  
SIMULATE,DSNAMES=, VOLUMES=, COPIES=, DATE=, LIST=, SCRATCH=, DELPROF=, SMSNOSMS, PRIMARY  
2=, NOVALIDATE
```

SIMULATE

This optional parameter indicates that the execution is to be simulated. Messages will be issued as if processing has taken place, but no updates to the archive index will be performed.

DSNAMES=

This optional parameter indicates that processing is to be limited to the data set names and patterns specified in the supplied list. From 1 to 20 entries can be supplied.

VOLUMES=

This optional parameter indicates that processing is to be limited to the volumes specified in the supplied list. For archive data sets on disk (rather than tape), this list consists of the CA Disk generated keys rather than the actual volumes. From 1 to 40 volume serial numbers, prefixes, or patterns can be supplied.

Note: Sysparm IXMCLEAN can increase the number of volumes selected for processing. The purpose of this sysparm (described in IXMCLEANy in the *Systems Guide*) is to cleanup broken chains between ARCHVOLS and DSNINDEX records. So even you restrict the number of volumes to be selected by specifying VOLUMES= on your command, IXMCLEAN will add to this list any unchained ARCHVOL.

COPIES=

This optional parameter can be defined 1-9999 to indicate that this number of versions of each data set is to be maintained in the archives. For example, if COPIES=3 was specified, all but the three most current unexpired versions of a data set are to be deleted from the archives, regardless of their expiration dates.

Note: GDG data set names are not considered special for this support. For the example given, the three most current versions of each DSN.GxxxxVyy-named data set will be kept.

A version is determined by the data set name and original volume of a data set. If multiple copies of a data set are archived on the same date they are considered one version of the data set. Member level backups, described in the section Member Level Backup in the chapter "Backup/Archive", are not counted as a copy under this process.

For additional information on other processing options, see the keyword parameter SMSNOSMS, and the sysparm descriptions for DSNUPVLn and IXMDUPDTn in the *Systems Guide*.

DATE=YYDDD

This parameter can specify a date in any accepted CA Disk format to be used as the current date whenever date comparisons are made.

LIST=

Normally a message is printed for each data set deleted from the archives. If you want to suppress the messages, use this parameter. LIST=NONE will suppress all messages. ALL is the default.

SCRATCH=

This keyword parameter tells CA Disk whether or not to scratch the archive data set on disk when all DSNINDEX records that point to it have been deleted and the ARCHVOLS record is being deleted as a result. SCRATCH=YES is the default. SCRATCH=NO will keep the archive data set on disk even though there are no more DSNINDEX records and the ARCHVOLS record is being deleted.

DELPROF=

This keyword parameter tells CA Disk whether or not to delete the discrete RACF profile associated with a DSNINDEX record being deleted. (This is the discrete profile that CA Disk created and saved when the DSNINDEX record was built. Generic profiles are not touched, regardless of the value of this parameter.) DELPROF=YES is the default. DELPROF=NO will keep the discrete profile even though the DSNINDEX record is being deleted.

SMSNOSMS

Normally, the DSNDDELETE command manages SMS and non-SMS DSNINDEX records quite differently. SMS DSNINDEX records are managed according to their Management Class attributes, whereas non-SMS records are managed by the combination of their expiration date and any DSNDDELETE parameters. This parameter causes all DSNINDEX records to be managed as non-SMS.

Note: This keyword parameter is only valid when used with the COPIES= parameter. For details, see COPIES=.

For example, to cause all DSNINDEX records (SMS and non-SMS) to be reduced to 2 copies each, execute the following command:

```
DSNDDELETE COPIES=2,SMSNOSMS
```

PRIMARY2=

This optional parameter indicates whether processing is to be limited by type of DSNINDEX record. PRIMARY2=YES is the default which indicates all DSNINDEX record types are processed. PRIMARY2=NO will keep PRIMARY2 type DSNINDEX records from being processed even though they may be expired. PRIMARY2=ONLY will cause only PRIMARY2 type DSNINDEX records to be processed.

NOVALIDATE

This optional parameter indicates that validation of the DSNINDEX and ARCHVOLS subfiles for and FDS will not be done. By default, IXMAINT confirms the integrity of index and data address pointers as well key sequence within the FDS for the DSNINDEX and ARCHVOLS subfiles before proceeding to delete records. If a pointer is in error indicating some sort of data set corruption has occurred, processing will be terminated with a user abend code U100 to ensure the corruption is noticed by those running the system. If the abend occurs, see the FDS Processing section of the *Systems Guide* for instructions on FDS recovery.

Deleting Expired Volumes

The ARCHVOL expiration date is established based on the value set for sysparm DYNEXPDT, described in DYNEXPDTE99365 in the *Systems Guide*. The recommended value for this sysparm is E99365 (never scratch), its default value. Using this recommendation ensures that an ARCHVOL record will exist in the archives, as long as it contains one or more unexpired DSNINDEX records.

For details on how to delete an ARCHVOL created under this recommendation, see [Deleting Expired Datasets](#).

The VOLDELETE command is used to delete ARCHVOLS that were created outside of this recommendation. It causes all expired archive volumes (and its contents), either on tape or on disk, to be deleted from the archives. An archive volume (ARCHVOLS record) is considered expired if its expiration date is less than or equal to the current date. Any ARCHVOLS record not expired will not be processed. (See EXPDSNS for the exception to this.)

The EXPDSNS parameter adds a *DSNDELETE DSN=* function to VOLDELETE. Any non-SMS DSNINDEX record that has expired by EXPDT or catalog control will be deleted even though the ARCHVOL has not expired. (SMS-managed data sets are bypassed by VOLDELETE processing even with specifying the EXPDSNS parameter). However, if VOLDELETE is expiring a volume that did not have a *never expire* expiration date, all data sets on that volume, SMS managed and non-SMS, will be purged without regard to EXPDT or SMS verification.

If the ARCHVOLS expiration date is *never expire* or *catalog control*, VOLDELETE will provide no processing benefit over DSNDELETE. DSNDELETE is the only command needed for removal of expired DSNINDEX and ARCHVOL records from the Files Data Set along with assurance of SMS verification.

If it is determined that no DSNINDEX records exist for a given ARCHVOLS record, a flag is checked to ensure that the volume has been closed. If the volume is still flagged as OPEN, it will not be deleted unless the creation date is less than today's date minus one (that is, CREDIT < TODAY-1). This check maintains integrity even if ARCHIVE and IXMAINT are running concurrently over midnight.

Unlike the DSNDELETE command, data sets that have expired can remain in the archive index if the ARCHVOLS record they correspond to has not yet expired. However, the user should be very cautious if the VOLDELETE command is used in an environment where any of the following are in place at archive time:

```
//ARCHIVE0 DD LABEL=EXPDT=96350
//ARCHIVEC DD LABEL=EXPDT=96350
//SYSIN    DD *
SCAN REALVOLS
  SELECT DSN=/
    ARCHIVE EXPDT=97200
OR
```

```
//ARCHIVE0 DD LABEL=RETPD=60
//ARCHIVEC DD LABEL=RETPD=60
//SYSIN    DD *
SCAN REALVOLS
  SELECT DSN=/
    ARCHIVE RETPD=365
OR
```

```
//SYSIN    DD *
SCAN REALVOLS
  SELECT DSN=/
    ARCHIVE EXPDT=97200
//SYSPARMS DD *
DYNEXPDT=96350
OR
```

```
//SYSIN    DD *
SCAN REALVOLS
  SELECT DSN=/
    ARCHIVE RETPD=365
//SYSPARMS DD *
DYNEXPDT=60
```

In the previous examples, the CA Disk-calculated ARCHVOLS expiration date is overridden. In every case, the data sets are kept in the archives for a longer period than that of the ARCHVOLS they reside on.

When the VOLDELETE command is executed in the previous environment, unexpired data sets will be deleted; because they exist on volumes that have expired.

VOLDELETE Command

The VOLDELETE command syntax is:

DSNDELETE

VOLDELETE SIMULATE, VOLUMES=, EXPDSNS, DATE=, LIST=, SCRATCH=, DELPROF=, NONVALIDATE

SIMULATE

This optional parameter indicates that the execution is to be simulated. Messages will be issued as if processing has taken place, but no updates to the archive index will be performed.

VOLUMES=

This optional parameter indicates that processing is to be limited to the volumes specified in the supplied list. For archive data sets on disk (rather than tape), this list consists of the CA Disk-generated keys rather than the actual volumes. From 1 to 40 volume serial numbers, prefixes, or patterns can be supplied.

EXPDSNS

This optional parameter indicates that in addition to expired volumes, all expired data sets are also to be deleted, regardless of their presence on an unexpired volume. Specifying this parameter is equivalent to running DSNDELETE VOL=/,DSN=/ before running VOLDELETE. That is, all expired DSNINDEX records encountered during processing will be deleted regardless of what ARCHVOLS record they point to. In addition, if all DSNINDEX records pointing to a given ARCHVOLS record are deleted, the ARCHVOLS records will also be deleted. This can be confusing in that if this occurs, it is possible that an ARCHVOLS record can be deleted that was not specified on the VOLDELETE command of this job.

Note: This parameter does not process SMS-managed data sets.

DATE=

This parameter can specify a date in any accepted CA Disk format to be used as the current date (instead of the real current date) whenever date comparisons are made.

LIST=

Normally a message is printed for each data set deleted from the archives. If you want to suppress the messages, use this parameter. LIST=NONE will suppress all messages. ALL is the default.

SCRATCH=

This keyword parameter tells CA Disk whether or not to scratch the archive data set on disk when the ARCHVOLS record is being deleted. SCRATCH=YES is the default. SCRATCH=NO will keep the archive data set on disk even though the ARCHVOLS record is being deleted.

DELPROF=

This keyword parameter tells CA Disk whether or not to delete the discrete RACF profile associated with a DSNINDEX record being deleted. (This is the discrete profile that CA Disk created and saved when the DSNINDEX record was built. Generic profiles are not touched, regardless of the value of this parameter.) DELPROF=YES is the default. DELPROF=NO will keep the discrete profile even though the DSNINDEX record is being deleted.

NOVALIDATE

This optional parameter indicates that validation of the DSNINDEX and ARCHVOLS subfiles for an FDS will not be done. By default, IXMAINT confirms the integrity of index and data address pointers as well key sequence within the FDS for the DSNINDEX and ARCHVOLS subfiles before proceeding to delete records. If a pointer is in error indicating some sort of data set corruption has occurred, processing will be terminated with a user abend code U100 to ensure the corruption is noticed by those running the system. If the abend occurs, see the FDS Processing section of the *Systems Guide* for instructions on FDS recovery.

Purging Archive Data Sets/Volumes

Use the PURGE command to delete entries from the archive index regardless of their expiration dates. It applies to both archived data sets (DSNINDEX records) and archive volumes (ARCHVOLS records). Like the DSNDELETE command, when all data sets are deleted that exist on an archive volume, the archive volume will be released from the archives.

The PURGE command performs an uncatalog action for the tape (per sysparm UNCATARC), uncatalogs archived data sets (per sysparm UNCATPSU), and interfaces to tape management systems to release the tape (per sysparm TMSCTLEX).

Careful consideration should be done prior to doing a PURGE. If a PURGE must be done prior to a REBUILD, to prevent the previous actions from occurring, specify the following sysparms in the PURGE job:

```
'UNCATARC'  
'UNCATPSUX'  
'TMSCTLEX      '(sysparm followed by 8 blanks)  
'IXMCLEANY'
```

If it is necessary to retrieve one (or only a few) data set(s) from an ARCHVOLS, see Archive Index Rebuild Utility for a method to retrieve the data sets that will not cause any undesirable side effects that PURGE can cause.

Note: The ARCHVOL entry is also enqueued to prevent it from being deleted while in use.

PURGE Command

The PURGE command syntax is:

PURGE SIMULATE, VOLUMES=, DSNAMES=, LIST=, SCRATCH=, DELPROF=, NOVALIDATE, FORCE

SIMULATE

This optional parameter causes messages to be issued as if processing has taken place, but no updates to the archive index will be performed.

VOLUMES=

This parameter is required if the DSNAMES parameter is not specified; otherwise it is optional.

This optional parameter indicates that processing is to be limited to the volumes specified in the supplied list. The list should consist of primary volumes only, not copy volumes. Any copy volume that is listed will be ignored. Copy volumes and multivolume chains are automatically removed when their associated primary volumes are purged. For archive data sets on disk (rather than tape), this list consists of the CA Disk-generated keys rather than the actual volumes. From 1 to 40 volume serial numbers, prefixes, or patterns can be supplied.

Note: To remove a copy volume without removing the primary, use the IXUPDATE RESET command with COPY=NULL, or purge and rebuild the primary volume.

DSNAMES=

This optional parameter (required only if the VOLUMES parameter is not specified) indicates that processing is to be limited to the names specified in the supplied list. From 1 to 20 data set names, prefixes, or patterns can be entered. The names see DSNINDEX entries to be purged, not to ARCHVOLS entries.

LIST=

Normally a message is printed for each data set deleted from the archives. If you want to suppress the messages, use this parameter. LIST=NONE will suppress all messages. ALL is the default.

SCRATCH=

This keyword parameter tells CA Disk whether or not to scratch the archive data set on disk when the ARCHVOLS record is being deleted. SCRATCH=NO is the default. SCRATCH=YES will scratch the archive data set from disk after the ARCHVOLS record has been deleted.

DELPROF=

This keyword parameter tells CA Disk whether or not to delete the discrete RACF profile associated with a DSNINDEX record being deleted. (This is the discrete profile that CA Disk created and saved when the DSNINDEX record was built. Generic profiles are not touched, regardless of the value of this parameter.) DELPROF=NO is the default for PURGE, which will keep the discrete profile even though the DSNINDEX record is being deleted. DELPROF=YES causes discrete profiles to be deleted along with the index records.

NOVALIDATE

This optional parameter indicates that validation of the DSNINDEX and ARCHVOLS subfiles will not be done. By default, IXMAINT confirms the integrity of index and data address pointers as well key sequence within the Files Data Set for the DSNINDEX and ARCHVOLS subfiles before proceeding to delete records. If a pointer is in error indicating some sort of data set corruption has occurred, processing will be terminated with a user abend code U100 to ensure the corruption is noticed by those running the system. If the abend occurs, see the Files Data Set Processing section of the *Systems Guide* for instructions on Files Data Set recovery.

FORCE

This optional parameter should be used when the archive volume being purged has been flagged as containing FAST Archive entries. FORCE is needed to prevent the unintentional deletion of the FAST Archive entries.

SMS and IXMAINT

IXMAINT uses SMS Management Class information to determine eligibility of data sets for Index Maintenance.

Except for the addition of the MGMTCLAS parameter, documented below, external IXMAINT processing remains unchanged. Internally however, IXMAINT will determine when it is running in an SMS environment and use the Management Class information as documented below to make its decisions.

SMS fields CA Disk Supports

Number of Backup Versions

Data set Exists on Primary (MC). This optional field can be defined 1 to 13, or blank. The default is 2. When this field is defined with a number, the number indicates the maximum number of backup versions to keep for a data set. When this field is defined as blank, an unlimited number of backup versions will be kept. For more information, review the sysparm description for DSNUPVL in the *Systems Guide*.

Note: The setting of sysparm DSNUPVL can affect all of the SMS fields that pertain to Backup. This sysparm controls whether or not multiple backup copies of the data set from different volumes are counted as the same data set. It is recommended that this sysparm be set to 'Y' in DFSMS environments. This is because of volume pooling and the fact that SMS does not allow uncataloged copies. If this sysparm is left set to 'N', the same data set name from another volume will start over on the copy count for SMS managed copy count checks. Potentially, this can dramatically increase the number of backup copies kept in IXMAINT.

Number of Backup Versions

Data set Deleted from Primary (MC). This optional field can be defined 0 to 13, or blank. The default is 1. When this field is defined with a number, the number indicates the maximum number of backup versions of the data set to keep in the backup pool. When this field is defined with 0, all backup versions are deleted from the backup pool once IXMAINT detects that the data set is deleted. When this field is defined with a blank, unlimited versions will be kept.

Retain Days Only Backup Version

Data set Deleted (MC). This optional field can be defined 1 to 9999, NOLIMIT, or blank. The default is 60. When this field is defined with a number, the number indicates the number of days to keep the most recent backup version of a deleted data set with no archive versions starting from the day CA Disk detects that it has been deleted. If *Retain Days Extra Backup Versions* has a value of NOLIMIT, then all remaining backup versions will be deleted after this number of days as well. When this field is defined NOLIMIT, the number of days the most recent version will be kept is not limited. When this field is defined with a blank, other IXMAINT command parameters will be used to process the data set.

Retain Days Extra Backup Versions (MC)

This optional field can be defined 1 to 9999, NOLIMIT, or blank. The default is 30. When this field is defined with a number, the number indicates the number of days to keep the backup versions other than the most recent one. When this field is defined NOLIMIT, all backup versions, up to the value of *Number of Backup Versions* (Data Set Deleted), of a data set will be kept for the number of days specified by *Retain Days Only Backup Version*. When this field is defined with a blank, other IXMAINT command parameters will be used to process the data set.

Determining How Long To Keep DSNINDEX Records

By default, an archive copy of an SMS-managed DSNINDEX record will be kept until the Expire Days Non-Usage (MC) is met, and/or the Expire After Date/Days (MC) has occurred.

In theory, this works quite well. But in the real world, this data set can be archived and restored only to be modified, backed up, and eventually archived again. The result is two archive DSNINDEX copies waiting for the expiration date to occur. To complicate matters, the first archive copy would have its expiration date extended because of the restore. (that is, the restore date would take precedence over the format-1 DSCB USEDT). This means that both copies would be kept for several years, or worse yet, indefinitely if the Management Class expiration date attributes were defined as NOLIMIT.

Therefore, it is suggested that sysparm RESIXRPD (described in RESIXRPDnn in the *Systems Guide*) be used to determine how long to keep DSNINDEX records which have previously restored. However, to ensure that there is at least one copy of a DSNINDEX record at all times, RESIXRPD should be used in conjunction with the Guaranteed Backup (SG) value.

Alternatively, the sysparm IXSMSDEL (described in IXSMSDELn in the *Systems Guide*) can be used to delete obsolete SMS archive DSNINDEX records from the DSNINDEX subfile during processing. When IXSMSDEL is set to Y (and IXMAINT is processing under SMS rules), the most current Archive SMS DSNINDEX is checked to see if it was re-catalogued at archive time and to validate that the current catalog volume is the recatalog volume (see RECATVOL sysparm). The archive DSNINDEX record is deleted if the current volume is not the recatalog volume, if the data set is uncatalogued, or if the archive DSNINDEX is not the most current.

Expiration Dates Inspected

Archive DSNINDEX records are inspected to see if a format-1 DSCB expiration date exist. If one does not exist, the following fields are then used to expire the data set:

Expire After Days Non-usage (MC)

This optional field can be defined 1 to 9999, or NOLIMIT. The default is NOLIMIT. When this field is defined with a number, the data set becomes eligible for expiration that number of days after the last use date. When this field is defined NOLIMIT, the data set can be kept indefinitely.

Expire After Date/Days (MC)

This optional field can be defined 1 to 9999, yyyy/mm/dd, or NOLIMIT. The default is NOLIMIT. When this field is defined with a number, the data set is eligible for expiration that number of days after the creation date. When this field is defined with a date, the data set becomes eligible for expiration on that date. When this field is defined NOLIMIT, the data set can be kept indefinitely.

Rolled-off GDS Action (MC)

This optional field can be defined MIGRATE, EXPIRE, or blank. The default is blank. When defined as MIGRATE, CA Disk deletes the rolled-off data set unless the expiration attributes have not expired. When this field is defined EXPIRE, CA Disk deletes the rolled-off data set unless the explicit expiration date in the format-1 DSCB is not met. When this field is defined with a blank, CA Disk deletes the rolled-off data set unless expiration attributes have not expired.

GDG and SCRATCH Attribute

When a GDG is defined with the SCRATCH attribute and there is no catalog entry, the data set is deleted (unless the data set expiration date has not been met).

Command Format

MGMTCLAS= is an optional parameter indicates the management class upon which to base IXMAINT processing. Processing for those SMS managed data sets that do not have a management class specified in their DSNINDEX record will be processed according to the rules defined in this Management Class. The command syntax is as follows:

```
VOLDELETE/DSNDELETE MGMTCLAS=
```

SMSPRINT and WHYFLAGS Diagnostics

When the DD statement SMSPRINT is included in any IXMAINT job, diagnostic messages displaying WHYFLAGS are printed to SYSOUT.

```

ADSDM584 9999 *****
ADSDM584 9999 *                DESCRIPTION OF IXMAINT WHYFLAGS DIAGNOSTIC                *
ADSDM584 9999 *****
ADSDM584 9999 *COLUMN - VALUE *OFF- * DESCRIPTION*                                *
ADSDM584 9999 *          1      * SET*                                              *
ADSDM584 9999 *0123456789012345* *                                              *
ADSDM584 9999 -----*-----*-----*-----*-----*-----*-----*-----*-----*
ADSDM584 9999 *-          *+0 *ERROR ON SMS - USE NON-SMS*                          *
ADSDM584 9999 *B          *+0 *BYPASS DATA SET DUE TO SMS*                        *
ADSDM584 9999 *N          *+0 *NOT SMS MANAGED*                                  *
ADSDM584 9999 *A          *+0 *SMS MANAGED BUT PROCESSED AS NON-SMS BECAUSE OF SMSNOSMS PARAMETER*
ADSDM584 9999 *S          *+0 *SMS MANAGED VOLUME*                              *
ADSDM584 9999 *C          *+1 *DSN IS CATALOGED*                                *
ADSDM584 9999 *L          *+1 *CATALOG LOCATE DONE*                              *
ADSDM584 9999 *U          *+1 *NON ZERO RETURN CODE FROM CATALOG LOCATE*          *
ADSDM584 9999 *V          *+1 *DSN IS UNCATALOGED AND MAX BACKUP VERSION DSN DELETED SPECIFIED*
ADSDM584 9999 *N          *+2 *DSN IS CATALOGED AND MAX BACKUP VERSION DSN EXISTS SPECIFIED*
ADSDM584 9999 *2          *+3 *PRIMARY2 OF OF PRIM1/PRIM2 PAIR WHEN PRIMARY2 PARM = YES*
ADSDM584 9999 *P          *+3 *PRIMARY2 NOT PAIRED OR PRIMARY2 PARM = ONLY*        *
ADSDM584 9999 *P          *+4 *DSNDELEX USER EXIT SAID TOKEEP RECORD, DO NOT DELETE IT*
ADSDM584 9999 *X          *+4 *DSNDLPEX USER EXIT UNCATALOG*                    *
ADSDM584 9999 *Z          *+4 *DSNDELEX USER EXIT SAID TODELETE RECORD*            *
ADSDM584 9999 *Y          *+4 *IXMREVEX USER EXIT SAID TO DELETE RECORD*          *
ADSDM584 9999 *K          *+4 *IXMREVEX USER EXIT SAID TO KEEP RECORD, DO NOT DELETE IT*
ADSDM584 9999 *M          *+5 *CALCULATED MANAGMENT CLASS EXPIRE AFTER DAYS EXPIRATION DATE*
ADSDM584 9999 *D          *+6 *MANAGMENT CLASS EXPIRE DAYS NON-USAGE SPECIFIED*    *
ADSDM584 9999 *S          *+6 *USING MANAGMENT CLASS EXPIRE DAYS NON-USAGE*        *
ADSDM584 9999 *D          *+7 *USING DSNINDEX FORMAT 1 DSCB EXPIRATION DATE (DSNDSIED)*
ADSDM584 9999 *D          *+8 *USING DSNINDEX EXPIRATION DATE (DSNEXPDT)*          *
ADSDM584 9999 *M          *+8 *USING MANAGMENT CLASS EXPIRATION DATE*              *
ADSDM584 9999 *M          *+9 *NTH COPY OF FILE. NOT FIRST COPY*                  *
ADSDM584 9999 *1          *+9 *THIS IS LAST COPY OF DSN - DELETED*                *
ADSDM584 9999 *F          *+9 *THIS IS FIRST COPY OF THIS DSN, BUT IS NOT THE LAST COPY*
ADSDM584 9999 *E          *+9 *THIS IS THE LAST COPY OF THIS DSN - DATA SET EXISTS*
ADSDM584 9999 *L          *+10 *MANAGMENT CLASS DAYS TO KEEP LAST BACKUP VERSION SPECIFIED*
ADSDM584 9999 *D          *+10 *MANAGMENT CLASS DAYS TO KEEP BACKUP VERSION SPECIFIED*
ADSDM584 9999 *M          *+10 *CALCULATED DAYS TO KEEP BACKUP VERSION*            *
ADSDM584 9999 *N          *+10 *DSN DELETED. DSNINDEX EXPIRATION DATE MODIFIED.*   *
ADSDM584 9999 *Y          *+10 *DSN DELETED*                                    *
ADSDM584 9999 *C          *+11 *BEGAN EXPIRE CHECKS*                             *
ADSDM584 9999 *D          *+11 *DSNINDEX EXPIRATION DATE (DSNEXPDT) BEING USED*     *
ADSDM584 9999 *M          *+11 *EXPIRATION DATE CALCULATED FROM MANAGMENT CLASS DAYS TO KEEP BACKUP USED*
ADSDM584 9999 *X          *+11 *DSN IS EXPIRED*                                  *
ADSDM584 9999 *L          *+12 *DSNINDEX EXPIRATION DATE (DSNEXPDT) = 99000 AND SYSPARM UNCATDEL = Y*
ADSDM584 9999 *U          *+12 *DSNINDEX EXPIRATION DATE (DSNEXPDT) = 99000, SYSPARM UNCATDEL = Y, UNCAT*
ADSDM584 9999 *X          *+12 *SMS GDG DEFINED WITH SCRATCH AND NOW UNCATALOGED*   *
ADSDM584 9999 *D          *+12 *SMS BYPASS, DMSAR RUNNING, SYSPARM IXSMSDEL = Y*    *
ADSDM584 9999 *W          *+12 *SMS DUPLICATE ARCHIVE, SYSPARM IXSMSDEL = Y*        *
ADSDM584 9999 *Y          *+12 *SMS UNCATALOGED, SYSPARM IXSMSDEL = Y*            *
ADSDM584 9999 *Z          *+12 *SMS CATALOGED NOT PSEUDO VOLUME, SYSPARM IXSMSDEL = Y*
ADSDM584 9999 *C          *+13 *DID CSI INQ TO GET CATALOG INFO*                  *
ADSDM584 9999 *L          *+13 *DID CATALOG LOCATE FOR THIS DSN*                  *
ADSDM584 9999 *P          *+13 *DID NOT DO CATALOG LOCATE FOR THIS DSN, USING PREVIOUS CATALOG LOCATE*
ADSDM584 9999 *U          *+13 *DSNINDEX EXPIRATION DATE (DSNEXPDT) = 99000 AND SYSPARM UNCATDEL = Y*
ADSDM584 9999 *X          *+13 *CHECKING FOR EXPIRED DSNs*                        *
ADSDM584 9999 *1          *+13 *CHECKING FOR EXPIRED DSNs AND DELETE COMMAND ACTIVE*
ADSDM584 9999 *A          *+14 *SCRATCH AT BACKUP TIME*                           *
ADSDM584 9999 *B          *+14 *BACKUP COPY*                                    *
ADSDM584 9999 *D          *+14 *SYSPARM DSNUPVOL = Y, DUP VOL *FAKE* USED*         *
ADSDM584 9999 *K          *+14 *CHECKING COPIES*                                *
ADSDM584 9999 *M          *+15 *MEMBER LEVEL ARCHIVE*                            *
ADSDM584 9999 *****

```

Interpreting the WHYFLAGS

The following steps can be used to interpret these flags:

Locate the WHYFLAGS= section of SYSOUT. In the previous example, it looks like this:

```

              11111
            012345678901234
WHYFLAGS = SCN      E M LB

```

Note: Column headings have been added for clarity purposes.

As you can see, these flags are column dependent beginning with column 0. To interpret these flags, you must determine which column the codes are in. By reviewing the previous example, we see that the codes reside in the following columns:

Value	Offset	Description
S	+0	SMS managed volume
C	+1	DSN is cataloged
N	+2	DSN is cataloged and max backup version dsn exists specified
E	+9	Last copy of this DSN - data set exists
M	+11	Expiration date calculated from MC days to keep backup used
L	+13	Did CSI INQ to get catalog info
B	+14	Backup copy

For more information on the meanings of these codes, see the IBM Storage Administration Reference manual.

Archive Index Rebuild Utility

A utility is available that can be used to recreate index records from the archive tapes (or archive data sets on disk) that contain the archived data sets. The support makes use of the control record and format-1 DSCB that are written with the data at the time of archival.

Use of this facility would be appropriate in any of the following situations:

- Restoring data sets at a site other than that at which the data sets were archived.
- The archive index has been destroyed and no suitable backup is available.
- FDS logging (discussed in FDS Logging in the *Systems Guide*), was not in place, and the FDS cannot be fully recovered to a particular point in time.
- The index entries for a given volume have either expired or have been PURGED, but the tape has not been written over yet, and one or more data sets must be restored.

Use of TRecover and Rebuild

If a tape Archive data set has been overwritten and recovered by TRecover, the File Sequence number of the archived data may have changed. It will be necessary to purge the original tape volume by using the PURGE command with the FORCE parameter to remove all entries for the volume. If the Archive volume is not purged, the DSNINDEX entries for the volume can contain incorrect information and cause errors if used to restore data sets.

Full Rebuild

A FULL Rebuild is the default and is performed when none of the parameters ARCONLY, DSNAMES or FILENOS are specified. A Full Rebuild requires that the Archive data set be not defined by an ARCHVOLS entry and that none of the data sets on it be defined by a DSNINDEX entry, with the exception of FAST DSNINDEX entries.

ARCONLY Rebuild

An ARCONLY Rebuild is requested by specifying the ARCONLY parameter on the REBUILD command. It requires that the Archive data set be not defined by an ARCHVOLS entry and allows you to rebuild the ARCHVOLS entry if it has inadvertently been deleted. The optional parameter ARCKEY allows you to specify the original Key (in the case of a Disk Archive data set) or volume serial number (in the case of a Tape Archive data set) and will change any DSNINDEX entries that match the new key. No DSNINDEX entries are rebuilt. An ARCONLY Rebuild is mutually exclusive with a Selective Rebuild and will cause a syntax error.

Selective Rebuild

A SELECTIVE Rebuild is performed when one or both of the parameters DSNAMES or FILENOS are specified. A Selective Rebuild can be performed whether the Archive is defined in an ARCHVOLS entry or not. Data sets that match any pattern in the DSNAMES parameter or the FILENOS parameter are considered for rebuilding. Archived data sets are not rebuilt if there is a current DSNINDEX entry. The ARCHVOLS entry is updated to include any DSNINDEX entries that are rebuilt. A Selective Rebuild is mutually exclusive with an ARCONLY Rebuild and will cause a syntax error.

A Selective Rebuild can be performed when it is necessary to recover only a small number of Data Set entries that have been deleted from the FILES data set and the Archive data set has not been deleted or written over.

A Selective Rebuild will still read the entire Archive data set even if only the FILENOS parameter was specified and all of the requested files were already recovered. This is to allow Rebuild to correctly build the ARCHVOLS entry.

Special Considerations

The UNIT= parameter is used to match with the DYNnUNIT/DYnnUNIT values, and will mark the volume as assigned to DYNn/DYnn. The UNIT parameter must be specified if the tape volume requires one of the special DYNnUNIT/DYnnUNIT assignments. If the volume was originally assigned to one of the DYNn/DYnn units, it should probably be rebuilt with the same unit assignment. The original assignment can be determined by an examination of a LISTV report for the volume in question. The DYNn/DYnn assignment is verified to be the same in the case of a selective rebuild into an existing volume. The TYPE= parameter of the RESET command of the IXUPDTE utility can be used to assign the correct DYNn/DYnn unit if the wrong one is assigned.

Note: A full rebuild of an Archive data set is not allowed if an Archive data set already has an existing entry in the ARCHVOLS. In such a case, either the Archive data set must be scratched before resubmitting the job or a selective rebuild requested. To scratch the Archive data set, RESET command with the KEY= and SCRATCH parameters are described in RESET Command.

If it is necessary to retrieve one or only a few data sets from a tape, rather than doing a RESET on the tape, consider doing a selective rebuild of the tape by specifying either or both of the DSNAMES or FILENOS parameters:

1. Allocate a small temporary FDS
2. Perform a REBUILD on the volume pointing to the temporary FDS allocated in step one
3. RESTORE the data set(s) needed, again pointing to the temporary FDS
4. When steps 1-3 are complete, delete the temporary FDS

To avoid having DSNINDEX records point to copy archives, do not rebuild from archives that are considered copy volumes. Rebuild the primary archive and then copy with the COPY utility.

CA 1 Tape Management users that use either the direct or External Data Manager (EDM) interfaces should be aware that CA Disk will not update the TMC expiration date during REBUILD. You will need to update the TMC record manually to reflect the correct expiration date.

Some data set index entries for a given tape Archive data set have either expired or have been PURGED, but the tape has not been overwritten yet, and one or more data sets on that tape must be restored.

JCL - Rebuild Index

The ARCHVOLS and DSNINDEX records for data sets on an archive tape, or in an archive data set on disk, can be rebuilt by use of the following JCL:

```
//REBUILD EXEC REBUILD,DSN=ddd
//SYSIN DD *
REBUILD command and parameters are entered here
```

Note: The PROC symbolics could not support the 255 volumes and so were removed, thus JCL overrides must be used to specify the volume & unit information.

For Multiple Archive Volumes, use the following JCL:

```
//SYSIN DD *
  REBUILD
/*
//SYSPARM DD *
ARCVOLID$
DYN2UNITVTAPE
/*
//ARCHIVES DD DISP=OLD,DSN=&DSN,UNIT=VTAPE,
// VOL=SER=(251458,251495,251651,251739,251745,
// 253682,253690)
//
```

REBUILD Command and Parameters

A REBUILD command is not needed unless you want to specify an expiration date to the ARCHVOLS index entry that is different from the default computation. See the EXPDT parameter below. All other needed information is obtained from the DD statement for the input archive tape, or the input archive data set on disk. Symbolic parameters are provided in the JCL procedure to allow you to identify the input data set. If the archive data set spans multiple volumes, the volume serial numbers must be properly specified in the JCL as shown on the previous section.

The REBUILD command syntax is:

```
REBUILD  
OR
```

```
REBUILD DSNAMES=(HLQ1.yyy.nnn,HLQ2.zzz.nnn,HLQ3.aaa.nnn)  
OR
```

```
REBUILD FILENOS=(01.02,05)  
OR
```

```
REBUILD EXPDT=,FAST,ARCONLY,DSNPRINT=,ARKEY=  
OR
```

```
REBUILD UNIT=,FAST,ARCONLY,DSNPRINT=,ARKEY=
```

ARKEY=

This parameter specifies the original ARCHVOL key for the Archive volume when rebuilding using ARCONLY. Any existing DSNINDEX records that match the data sets on the volume being rebuilt will be reset to the new key assigned by specifying this parameter.

ARCONLY

This optional parameter requests the REBUILD utility to only rebuild the ARCHVOL entry. This parameter can be used to rebuild the ARCHVOL entry when it has been deleted to correct the problems associated with the volume(s).

DSNAMES=

Indicates that a Selective Rebuild is desired. Up to 50 dsname patterns can be specified that will be used to select files to be rebuilt from the archive data set. All files with a dsname that matches any of the patterns will be considered for rebuilding.

DSNPRINT=

This parameter can be specified as YES or NO to request that the data set entries on the Archive Data set be printed. The default value is YES for a normal rebuild and NO for an ARCONLY rebuild.

EXPDT=

The default expiration date for the Archive data set will be the highest date found on all of the individual archived data sets or the expiration date of the Archive data set, whichever is higher. This ensures that the Archive data set is not dropped due to either a DELETE command or a MERGE PROCESS until every data set has indeed expired. This default calculation can, however, be overridden by supplying a date in an accepted CA Disk format on this parameter. This can be desirable in cases where archive retention periods are set by an installation, and not by the data set expiration date or unit.

Note: The create TIME for the ARCHVOLS record will be x'0000' after REBUILD. The exact time is not available.

FAST

Specify FAST to indicate the ARCHVOLS record being built is to be flagged as containing Fast Archive data sets.

FILENOS=

Indicates that a Selective Rebuild is desired. Up to 50 File Numbers can be specified that will be used to select files to be considered for rebuilding from the archive data set.

PRIMARY2=

Indicates that the ARCHVOLs and DSNINDEX record should be rebuilt as PRIMARY2 records. The corresponding flags will be set and appropriate expiration dates used.

UNIT=

The UCB device type determines the default unit type. You can override the default for DYN unit types by specifying the same esoteric name as used in the JCL. This ensures that ARCFLAG2 and ARCFLAG4 are set correctly in the ARCHVOL record.

IXUPDATE Utility

The following set of commands has been supplied to provide a facility to update various fields in the CA Disk archive data set name index (DSNINDEX), and archive volume index (ARCHVOLS). The functions of each are described prior to each command.

JCL - Index Updates

Use the following JCL to execute the volume and data set index update commands:

```
//IXUPDATE EXEC IXUPDATE
//SYSIN DD *
Any combination of DELETE, CHANGE,RESET,BUILDARC commands
/*
```

IXUPDATE Utility Condition Codes

The following table lists the possible condition codes that can occur while executing the IXUPDATE Proc.

0

Successful.

4

Informational messages issued.

8

Errors occurred and explanatory messages issued.

16

Error in the value specified for the TIMEDATE parameter.

DELETE Data Set Index Command and Parameters

The DELETE command is provided to eliminate one or all of the data set index records for a specific data set.

The DELETE data set index command syntax is:

```
DELETE DSNNAME=, VERSION=, TIMEDATE=, PRIMARY2, ALL
```

DSNAME=

The explicit name of a dataset to be deleted from the DSNINDEX.

VERSION=

To delete an older version of the data set contained in the FILES, specify -1 to -99 to indicate successively older versions of the dataset (-99 is the oldest version). By default, the most recent version of the dataset is deleted.

If more than one version of a data set is being deleted at a time, delete them in *reverse* order (that is, -3, -2, -1 rather than -1, -2, -3) to prevent the shifting of the relative version number as each is deleted.

PRIMARY2

This optional parameter specifies that only the PRIMARY2 data set index records are to be deleted. By default, only PRIMARY1 data set index records are deleted.

TIMEDATE=

This optional parameter can be used to provide the specific time and date of the DSNINDEX entry to be deleted. The format is HHMMYYDDD or HHMMYYYYDDD.

ALL

The presence of this optional parameter causes all DSNINDEX records for all versions of the data set to be deleted.

CHANGE Data Set Index Command and Parameters

The CHANGE command is provided to reset the CA Disk expiration date for a specific data set.

The CHANGE data set index command syntax is:

```
CHANGE  DSNNAME=,VERSION=,EXPDT=,F1EXPDT,TIMEDATE=,MGMTCL=
        STORCL=,DATACL=,DEFAULTMC,PRIMARY2,FAST=
```

DSNAME=

The explicit name of the data set for which the CA Disk expiration date is to be changed.

VERSION=

To change the expiration date of an older version of the dataset, specify -1 to -99 to indicate successively older versions of the dataset (-99 is the oldest version). By default, the most recent version of the dataset is changed.

EXPDT=

The date in an accepted CA Disk format that is to be the new CA Disk expiration date for this archived data set.

F1EXPDT=

The date in an accepted CA Disk format that is to be the new Format 1 DSCB expiration date for this archived data set.

Note: RESTORE/RECOVER will use this as the new Expiration Date, provided it does not violate any of the SMS Retention Period Attributes and the DSS path is not being used.

TIMEDATE=

This optional parameter can be used to provide the specific time and date of the DSNINDEX entry to be changed. The format is HHMMYYDDD or HHMMYYYYDDD.

DEFAULTMC

This optional parameter must be specified when the current MGMTCL field is *blank* and the SMS default management class is to be used for expiring of the DSNINDEX record. No EXPDT= parameter is required.

MGMTCL=

This optional parameter provides a way to specify a new SMS Management Class name. There are two ways of using this parameter:

To change or nullify (blank out) the Management Class. If you want to nullify the Management Class name, specify <>. However, when using this value either EXPDT= or DEFAULTMC must also be specified on the CHANGE command.

This ensures that the correct expdt will be specified in the DSNINDEX record, either 1999365 for SMS-managed data sets or a true retpd value for nonsms-managed data sets.

Also, when MGMTCL=<> and EXPDT= parameters are used together, the SMS bypass indicator DSNMCBYP is activated.

It provides a Management Class name, which will be validated by the SMS subsystem prior to use. The SMS bypass indicator DSNMCBYP will be turned off when the MGMTCL field is changed and is no longer blank.

STORCL=

This optional parameter provides new SMS Storage Class information. To nullify (blank out) the Storage Class, specify a value of <>. To change the name of the Storage Class, specify a non-blank value. However, when using this value the EXPDT= must also be specified on the CHANGE command.

Note: CA Disk recognizes *blank* Storage Classes as Non-SMS data sets.

DATACL=

This optional parameter provides new SMS Data Class information. To nullify (blank out) the Data Class, specify <>. To change the name of the Data Class, specify a non-blank value.

PRIMARY2

This optional parameter indicates that only the PRIMARY2 data set index records are to be changed. By default, only PRIMARY1 data set index records are changed.

FAST=

This optional parameter allows the Fast Archive status of the data set to be changed. The value can be either YES or NO. Setting the value to YES will automatically set the Fast Archive status to YES on the associated Archive data sets.

RESET Command

Use the RESET command to do the following:

- Modify the relationship between a primary archive volume and its backup (copy) volume.
- Reset the volume's expiration date.
- Change the value of the DYNn flag.

This command has the following format:

RESET KEY=,TYPE=,COPY=,DISABLE,ENABLE,CLOSED,OPEN,SCRATCH,EXPDT=,FAST=

KEY=

Specifies the key of the archive tape, or the key to the archive data set on disk, for which the status is to be changed.

TYPE=

Specifies the new DYNn value to set the corresponding flag in the ARCHVOLS record. This parameter is valid only for those ARCHVOLS created using DYN1 to DY20 values

COPY=

Specifies the serial number of the first volumes on which a backup copy of the archive data set exists. If the word NULL is specified, spaces are moved into the backup copy field in the ARCHVOLS record being updated.

DISABLE

Causes an archive tape volume (or archive data set on disk) to be disabled. This removes it from being eligible for use by CA Disk restore or recover processing. All subsequent restores that would have been directed to this volume are directed to the copy volume, if one exists.

ENABLE

(Optional) Reverses the status set by the DISABLE parameter. It removes the disabled status from an archive volume. The volume is made available again to restore and recover processing.

CLOSED

(Optional) Resets the indicator that an ARCHVOLS record was not closed properly. This allows the ARCHVOLS volume to be used before the safety grace period of two days is passed for Merge, Purge, or IXMAINT deletion.

OPEN

Turns on the indicator that an archive data set was not closed properly, and prevents its use before the safety grace period of two days *from the data set's creation* is up. This indicator being turned on has no effect if the archive data set is older than two days.

SCRATCH

(Optional) Specifies the deletion of the archive volume record, and its associated copies, if any. It will not cause any data set index records to be deleted. This is most commonly used to delete copy volumes from the CA Disk archives.

EXPDT=

(Optional) Provides the expiration date in an accepted CA Disk format that is assigned to the ARCHVOLS record being updated.

FAST=

(Optional) Allows the Fast Archive status of the Archive data set to be changed.

Valid values: YES or NO.

OPEN

(Optional) Provides the means to mark an ARCHVOLS record being OPEN (never closed). If it is already more than two days old, then this accomplishes nothing.

BUILDARC Command

The BUILDARC command is provided to allow volume records to be created by the user. In rare cases, this can be needed to compensate for the loss of a primary or backup archive data set.

If the status or backup volume serial for the volume being created does not apply, the RESET command can be used to reflect the desired values after the new record is created.

BUILDARC Command Syntax

The BUILDARC command syntax is:

BUILDARCDNAME=, VOLUME=, TYPE=, ARCTYPE=, EXPDT=, FAST=

DNAME=

This required parameter must provide the data set name to be placed in the archvols entry.

VOLUME=

This required parameter lists volumes (either tape or disk) that contain the CA Disk archive data set. As is the case with system created archive volume records, the first volume serial, if tape, becomes the record key. For disk volumes, CA Disk will generate a key.

TYPE=

Specify DASD, TAPE, 3480, 3490 or DYN1 to DY20 to indicate the type of devices you listed for the VOL= parameter.

Note: Specifying 3480 is acceptable for 3480, 3490 and DYN1 to DY20 devices.

ARCTYPE=

Specify PRIME or COPY to indicate whether the ARCHVOLS record being built is to be flagged as the primary index entry or a duplicate copy entry.

EXPDT=

The expiration date in an accepted CA Disk format that is to be assigned to the new ARCHVOLS record.

FAST=

Specify YES or NO to indicate whether the ARCHVOLS record being built is to be flagged as containing Fast Archive data sets or not.

Making Additional Copies of Archive Tapes

If you produce an archive tape without a duplicate copy, it is possible to create a copy after the fact. There are also other reasons why you may want to copy an archive tape. If you had a primary and duplicate copy, but the primary copy was lost, damaged, or otherwise not available, you may want to copy the duplicate copy to regain the protection in having two tapes.

CA Disk must keep track of multiple copies of archive tapes, so when it makes a copy of an existing archive tape, the volume serial of the new copy is recorded in the index of the volume record for which the copy is made.

RESET Command - If you are using the copy utility to replace a volume that is no longer available, you must also use the RESET command. See for instructions on how to **DISABLE** the volume that is no longer available.

For example, if primary volume A has copy volume B, A → B in the archvols entry. If A is destroyed and you want to create another copy C, use B as input to the copy utility, resulting in A → B → C. To create another copy D, use C as input to the utility, such that the pointer chain from A to B to C to D is kept intact. If A is destroyed, disable it. If B is destroyed, disable it. As long as the chain is intact, disabled entries will be bypassed automatically by restore, recover and merge functions.

The volumes to be copied must already have entries in the CA Disk archive tape record index (ARCHVOLS). An input stream contains records indicating the keys to the archive tapes that are to be copied. They are read from the COPYFILE DD statement. Beginning in columns 1 through 16, enter the 6-byte volume serial of the tape volume to be copied. (If you are copying an archive data set residing on disk rather than tape, enter the 6-byte key to the disk archvols record.) Only one key can be supplied per statement. Multiple statements can be entered to copy multiple volumes.

Note: Only the first volume of a multivolume archive data set should be specified in the input stream. The rest of the volumes in the sequence are automatically copied when the first volume is copied. If any one of the volumes of a multivolume archive data set needs to be copied, all have to be copied so that proper volume chaining is maintained on the volume labels.

The following JCL is presented as an example for the utility. It causes a copy volume to be created for archive tape VOLSER 123456. This utility only works for tape.

```
//JOBNAME JOB (ACCT, INFO), ETC.
//COPY EXEC PGM=ADSMI002, PARM=ADSDM227
//STEPLIB DD DISP=SHR, DSN=CAI.DISK.CCUWLOAD
//ABNLDUMP DD DUMMY
//ARCHIVEC DD DISP=NEW, KEEP, DELETE), DSN=CA.DISK.ARCHCOPY,
// VOL=(, , 255),
// LABEL=EXPDT=99365, UNIT=TAPE, , DEFER)
//*ARCHIVER DD DISP=OLD,
//* UNIT=(CART, , DEFER), VOL=(, , 255, SER=CPYME2)
//*ARCHIVES DD DISP=OLD,
//* UNIT=(TAPE, , DEFER), VOL=(, , 255, SER=COPYME)
//FILES DD DISP=SHR, DSN=CAI.DISK.FILES
//MSGPRINT DD SYSOUT=A, DCB=BUFNO=0
//PARMLIB DD DISP=SHR, DSN=CAI.DISK.CCUWPARM
//* UPDATE ARCCPYDV SYSPARM TO APPROPRIATE VALUE
//SYSPARMS DD *
ARCCPYVDYN1
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//COPYFILE DD *
123456
```

Note: A more flexible way to make a partial copy of multiple ARCHVOLS is discussed in the chapter "XCOPY."

Tapes Originally Allocated with DYN1/2/n

Tapes that were allocated using the DYN1/2/n path, and that have the DYN1/2/n flag turned on in their ARCHVOLS record in the Files Data Set, will not have the DYN flag turned on in their copy record. This occurs because the output tape is allocated with JCL and not with the dynamic allocation sysparms.

To turn on the appropriate DYN flag in the copy tape's ARCHVOLS record, code the ARCCPYDV sysparm with the appropriate value, as documented in the chapter "Sysparms" in the *Systems Guide*.

Chapter 11: MERGE

This section contains the following topics:

[About Merge](#) (see page 451)
[Overview](#) (see page 452)
[Processing Logic](#) (see page 461)
[JCL - MERGE Command](#) (see page 462)
[MERGE Command](#) (see page 463)
[SMS and MERGE](#) (see page 470)

About Merge

This function provides a convenient means to manage the index entries for archived data sets and the volumes that contain them.

Command available: MERGE

JCL Proc executed: MERGE

The MERGE function helps you to manage the CA Disk archive data sets that have been put on various disk, and tape cartridge.

CA Disk data compression, activated via the DCDATACP sysparm, has no effect on MERGE. Data is handled whether it's compressed or not.

Overview

Merge processing has been designed to provide maximum flexibility in managing the archive volumes on disk, tape and other storage media, and at the same time make the best use of the available resources.

When data sets are archived, each is assigned a retention period (expiration date). By definition, then, 100 percent of the data sets are unexpired on the day they are first archived. To make the most efficient use of both disk and tape, CA Disk (by default) packs all of these archived data sets into a single file (if archiving to tape), or a single data set allocated on disk.

As each day passes, however, some of these data sets can expire, creating dead space in the archives. Given enough time, all of the data sets will expire and the tape can be recycled or the archive data set on disk can be deleted to reclaim the space. (The IXMAINT function accomplishes this.)

In summary, each archive volume (tape or disk data set) starts out as packed full of 100 percent unexpired data sets. As the days pass, this will eventually drop to zero percent; that is, all data sets expired. The important questions are, (1) how fast does this take place and (2) can you afford to wait until it drops all the way to zero percent? In many cases if you do; tapes and disk space will be wasted. The archives need to be consolidated to reclaim the space.

This is the function that merge performs. Each merge run consolidates the unexpired data sets from one or more tape volumes (or archive data sets on disk) onto new archive volumes, thus freeing up tape volumes and/or disk space. This reduces the space required to store the CA Disk archives, as well as eliminating disk archive data sets with a large percentage of expired data sets. After the unexpired data sets have been successfully merged forward, the input data sets on disk are deleted, and input tapes are returned to *scratch status* — disk space is reclaimed and tapes are made available for reuse. All expired CA Disk index entries are dropped from the archives and unexpired entries are updated to reflect their locations on the new tape or disk archives.

Keeping some archives on disk can be desirable, because it maintains easy access to the data through auto-restore processing, without operator intervention. Disk-to-disk merging provides an efficient means to manage these archives, also without operator intervention.

Specifying What to Merge

Since it is very common to have many archive volumes, it cannot be practical to process all of them in a single merge execution. The time and machine resources required prohibit execution. To help you control the merge process, command parameters can be supplied to control, which archive data sets are to be merged. By running the merge in simulate mode and asking for a tape pull list, you can determine prior to the live merge run precisely what the input to the merge process will be.

The most commonly used of these command parameters is `TYPES=`, which allows you to limit the scan of archive data sets to those residing on specified device types. For example, you can specify that a merge of only the disk archive data sets be run daily, since it processes without operator intervention and makes efficient use of disk space.

The `PERCENT` parameter allows you to specify the threshold of active (unexpired) space that is to be maintained in an archive data set. When the value falls below the percent value specified, the volume is selected for merge processing. This allows archive tapes and/or archive disk data sets that are adequately used to be skipped.

The `MAXVOLS` parameter can be used to limit the merge process to a maximum number of archive volumes. Processing terminates when the maximum volume limit is reached. A warning message indicates that some volumes have been bypassed. The next execution of merge processes the bypassed volumes.

`MERGE` can be further limited to processing only archive volumes whose names match those specified in the `TDSNAMES=` parameter. The `NOMERGEIN` parameter excludes archive volumes created in previous Merge or Xcopy runs from being selected as input to Merge. `INCLUDE` and `EXCLUDE` parameters further refine the input by permitting specific archive volumes to be listed and acted upon. The `RINCLUDE` parameter can be used to include specific real volumes where archive volumes reside.

The `DAYSOLD`, `FDAYSOLD`, `CREDIT` and `FCREDIT` parameters all limit the input based upon the age of the archive volumes; that is, how long ago were they created? Parameters `DATE=` and `EXPDAYS` provide additional flexibility based upon the archive expiration dates. `EXPDAYS` excludes volumes that are about to expire (why waste processing resources if they will expire and be deleted *tomorrow*?), and `DATE=` instructs merge to pretend that *today* is the date specified, rather than the current date, which obviously affects whether merge will think that a data set or volume has expired or not.

`MERGE`, by default, bypasses multi-volume TAPE files with message 4005; however, it does include multi-volume DISK files. If you want to also process multi-volume TAPE files, use the `INCLUDE` command parameter described in `INCLUDE=`. You can use the `MULTVOL` or `INCLUDE` parameters to over-ride the bypassing of multi-volume tape files.

After specifying what to merge, the `MINVOLS` parameter can be used to prevent the merge process if there are not enough archive volumes selected for processing. Processing terminates with a warning message when the minimum number of volumes is not met. The message indicates that the actual merge has been bypassed.

Specifying Merge Output Options

Similarly, great flexibility in controlling the creation of new archive volumes is provided through command parameters and system parameters.

By using the LIMITS parameter, you can specify that archive data sets are to be written to up to five new archive volumes, each containing only data sets within a given range of expiration dates.

Note: SMS-managed data sets are not selected by the LIMITS parameter.

You can also control the device type(s) to which the new archives are to be written through the dynamic allocation sysparms MERPnTYP and MERPnNAM. The most suitable archive medium for each range can also be specified; for example, one range can be written to disk archive data sets, another to tape, and yet another to tape cartridge.

Note: Recent advancements in the cartridge technology field have substantially improved the CA Disk Merge process. Cartridge capacities have increased allowing Data Centers to maintain a smaller number of ARCHVOLS. For an example of this technology, see the section StorageTek Redwood Device in the chapter "General Information" and the section IBM's Magstar 3590 Tape Device in the chapter "General Information." Merge also allows you to produce duplicate and/or tertiary copies concurrently with the creation of the new archive volumes. This is accomplished through dynamic allocation by specifying sysparms MERCnTYP/MERCnNAM or MERTnTYP/MERTnNAM. This process is sometimes referred to as triplexing, and is the most efficient way to produce multiple copies. The copies can also go to different device types, much like the primary copies.

You can, however, create additional copies by executing an extra job step following the primary merge. A duplicate copy or tertiary copy can be produced depending on what is created in the primary merge. See COPY for a further description.

The CA Disk MERGE PROC has the COPYFILE DD statement used when specifying the MERGE COPY parameter. The COPYFILE can be set to //COPYFILE DD DUMMY, if not used.

Implementation Concerns

Running merge can be a time-consuming process, especially when a large number of tapes are involved. You can avoid running it for some tape volumes by maximizing tape usage at archive time and ensuring that expiration dates for data sets written to each tape are not spread over a large period of time. This allows tapes to be removed from the archives in a timely manner when all data sets on each tape have expired. Use the index maintenance function DSNDELETE (or possibly VOLDELETE) to remove expired data sets and volumes from the archive index.

The merge process performs functions similar to the index maintenance DSNDELETE (and VOLDELETE) command, in that they both delete expired data sets from the archives. However, the index maintenance functions perform at a much more efficient rate. To improve merge performance, run the DSNDELETE (or VOLDELETE) index maintenance command prior to running merge.

Use SYSPARM MERRELTMMnn, where a Restore can be delayed by a Merge job. Specify Merge release time of 01 to 99 minutes (default is 00) to be used in both Merge and Restore. If MERRELTMM is non-zero and there is room on the current volume for the next data set, the time interval since the last output volume swap is compared to MERRELTMM. If that time interval is greater than or equal to MERRELTMM, a test ENQ for DMSWVOLS is issued for that volume. If a Restore has been waiting for that volume, the volume is released by Merge. Restore only issues the DMSWVOLS ENQ if MERRELTMM or ARCRELTMM is non-zero. Tuning can be done within Merge with MERRELTMM from 01 to 99 minutes, depending on individual needs.

Note: Sysparms ARCHPACK and ARCDKBZ are used by merge in the same manner in which archive processing uses them. The packing option should (probably) never be turned off (ARCHPACK specified with a value of N) if merge is creating tapes, since it would merge forward only one data set per tape, creating an enormous number of new archive tapes, and defeating the whole purpose of merge. ARCDKBZ should be set to an efficient value common to all the output devices you will be using.

Use SYSPARM MERRELTMMnn, where DMSAR can be delayed by a Merge job. Specify Merge release time MERRELTMM of 01 to 99 minutes (default is 00) to be used in both Merge and DMSAR. During Merge, if MERRELTMM 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 MERRELTMM. If that time interval is greater than or equal to MERRELTMM, 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 MERRELTMM is non-zero; the actual value of MERRELTMM is picked up in Merge. Tuning can be done within Merge with MERRELTMM from 01 to 99 minutes, depending on individual needs.

Operation Concerns

Merge is designed to run on a scheduled basis, based on your installation's needs. As stated previously, disk-to-disk merge can be run on a daily basis, since it requires no operator intervention and can quickly free up DASD space. For tapes, merge is usually run on a weekly or monthly basis, depending on the number of archive volumes created and the amount of space that can be recovered from them. Not all CA Disk archive volumes need to be run through merge. Process only those volumes that are wasting a large amount of space, or those archive data sets on disk that can now be moved to tape for the remainder of their retention period.

During a merge run, input 3480 cartridges use the DD statement `//ARCHIVER`, while input disk drives are referenced using the DD statement `//ARCHIVED`.

Therefore, depending on what is being merged, one tape and/or one 3480 drive will be needed for input. Since the ARCHVOLS records contain device type information and data set names, CA Disk can dynamically allocate each of these input types as appropriate. (JCL can be provided for tape and 3480s, if desired.)

If the output is to disk, CA Disk always allocates the disk archive data set dynamically. To allocate all primary output devices dynamically, the data set names and the device types must be supplied in sysparms MERPnNAM and MERPnTYP. Concurrent duplicate output copies can also be created and dynamically allocated by supplying proper values for sysparms MERCnNAM and MERCnTYP. If the concurrent copies are not wanted, sysparms MERCnTYP must be set to NULL. Concurrent tertiary output copies can also be created and dynamically allocated by supplying proper values for sysparms MERTnNAM and MERTnTYP. By default, the MERTnTYP is set to NULL, so no concurrent tertiary copies are produced. For more information about these sysparms, see their sysparm descriptions in the *Systems Guide*.

(Tape or 3480 devices can be allocated via JCL if desired by supplying `//ARCHIVE1`, `//ARCHIVE2`, ... DD statements for the primary output, `//ARCHIVC1`, `//ARCHIVC2`, ... for the copy devices, and `//ARCHVT1`, `//ARCHVT2`, ... for the tertiary devices.)

You can optionally specify a system parameter that allows the computer operator to terminate merge prematurely (a graceful shutdown). Message 4254 is issued to the operator at the beginning of the merge process. If the operator replies to the message to indicate that Merge processing is to be stopped, a response of *END* will terminate the job at the end of the current input volume or a response of *NOW* will terminate the job at the end of the current data set. Processing will complete normally, and all reports and statistics will be generated as normal. This option permits you to start merge and run until the resources are needed for something else. Thus merge can be run in otherwise unused machine time. Specify this option through sysparm MERREPLY.

TAPEPULL Processing

Since some installations require that all tapes be pulled from the tape library and be provided with the job that will use them before it goes into execution, a facility to generate such a list for merge processing is provided. This facility is invoked through the TAPEPULL parameter.

Two steps are required:

1. Run MERGE with the SIMULATE and TAPEPULL=GEN parameters (note that if you fail to specify SIMULATE, CA Disk will force a simulation run for you). This will accomplish two things:
 - Produce the list of tape volumes that will be used as input for the LIVE merge run, enabling operators to pull them from the library
 - Save the list in a disk data set, such that a subsequent LIVE run can use them as input

Note: The tape pull list will specify only tape volumes to be used for input for the merge run. If output is directed to tape, appropriate tape volumes will be called for during the live run, as they are needed.

There is a significant distinction between the printed tape pull list and the list written to disk. The printed list will not contain any keys of disk archive data sets that can have been selected for processing, whereas the list written to disk (which can be used by merge in a subsequent live run) will contain these keys. It is assumed that the printed list is used to actually allow tape volumes to be pulled and readied prior to the live run.

The TAPEPULL List written to disk will only include the first volume of multi-volume files. The TAPEPULL printed List will include all volumes for operations use.

2. Run MERGE without the SIMULATE parameter and with TAPEPULL=USE specified.

It is important to note that when running merge using the TAPEPULL option, the volume selection is done during step 1 when the list is being generated. Therefore, all selection criteria desired should be specified on the command at this time.

Note: All parameters specified on TAPEPULL=GEN must also be specified on TAPEPULL=USE. The only additional parameters that can be added to the command are COPY and EXCLUDE.

When using the TAPEPULL function, an additional JCL statement must be supplied in the merge JCL procedure. It can be supplied as a JCL override or be included directly in the JCL procedure. The DD statement *TAPEPULL* must point to a sequential data set with the attributes RECFM=FB, LRECL=80, BLKSIZE=4000. The data set should be allocated prior to the merge run.

MERGE, by default, bypasses multi-volume TAPE files with message 4005; however, it does include multi-volume DISK files. If you desire to also process multi-volume TAPE files, use the INCLUDE command parameter described in INCLUDE=. You can use the MULTVOL or INCLUDE parameters to over-ride the bypassing of multi-volume tape files.

Use the following DD statement as an example. It must be included for both the TAPEPULL=GEN and TAPEPULL=USE options.

```
//TAPEPULL DD DSN=SAMS.DISK.MERGE.TAPEPULL,DISP=OLD
```

Fast Merge Processing

When Fast Merge Processing is enabled, CA Disk can reuse existing backups or archives to merge a data set with no data movement. CA Disk checks the data set to see if it is unmodified since the previous Archive/ Backup. The product checks by reading the Format 1 DSCB record from the ARCHVOL and using the same variables that were used on the Modified criteria. When an eligible data set is found, the current DSNINDEX record is updated using the ARCHVOL information from the existing DSNINDEX record. The DSNINDEX is flagged as a FAST record. The ARCHVOLS record & tape expiration are updated to keep the longest expiration date but not all dates are compatible. The new Fast Merge DSNINDEX record is kept until Ixmaint expires. If Merge is processed, it is merged forward. All copies of the DSNINDEX are treated as different versions for Restore and Ixmaint processing.

Because the FILES must be checked for every data set being merged, the I/O count for the FILES increases. We recommend that you use this feature where the archive volume is on DASD and the unchanged backups are on tape. Using this feature lets you take advantage of the potential total processing time and ARCHVOL utilization reduction.

Note: Primary2 DSNINDEX records are not used as the target for a Fast Merge for the Primary1 version. Also see the section Fast Archive Processing in the chapter "Backup/Archive."

Note: Verify that the Sysparm DSNDUPVL is set to Y, if you use volume pooling. Setting the value to Y prevents the selection of older DSNINDEX versions.

Abend/Restart Considerations

During merge processing, the DSNINDEX is updated to reflect the new volume of residence as each data set is successfully copied to its new volume. If the merge step terminates abnormally, the ARCHVOLS file record for the current output will have incomplete statistical fields. These are normally updated during archive CLOSE processing. The absence of this information will not prevent the restoration of a data set from the output, nor will it cause any problems during processing of the output in a subsequent merge.

If desired, you can correct the incomplete statistical information in the ARCHVOLS record by scratching it from the files data set (RESET command), then using the REBUILD command to reconstruct the ARCHVOLS record from the archive data set, whether on tape or disk.

Merge processing can be restarted by submitting the same JCL as used in the failing job. If the same selection tests are specified, processing will continue with the data set being processed when the failure took place.

As noted previous, CA Disk considers any tape it has written to be a good tape, even though an abend can have occurred during processing. If you are using a tape management system in your installation, care should be taken to see that it does not treat such tapes as scratch tapes. (Note that this is not an issue if you use the EDM option of your tape management system.) Some tape management systems can default to giving *abend tapes* a one-day retention period. If this is the case, you should either change your tape management system to not do this for CA Disk tapes, or take clerical means to properly save the CA Disk tapes after an abend.

Simulate Mode Considerations

By using the SIMULATE parameter on a MERGE job, CA Disk instructs MERGE to execute without actually mounting any ARCHVOLS or moving any DSNINDEX records. The result is a report indicating which DSNINDEX and ARCHVOL records would have been selected in a live run. Fast processing cannot be simulated because no volumes are mounted and the Format 1 DSCB record is not available.

Because output devices are not allocated during simulations, CA Disk cannot accurately predict the number of output ARCHVOLS there will be in a live run. If you only want to know which DSNINDEX and/or ARCHVOL records will be selected for input by MERGE, then this is not a concern. However, certain steps need to be taken if your intent is to determine the approximate number of ARCHVOLS that will be required for output.

MERGE has a facility to simulate specific device types to predict volume capacity, and when a volume switch would occur. This is accomplished by specifying the appropriate DYNnUNIT/DYnnUNIT sysparm with the CA Disk value for the device type to be simulated as indicated below:

- The MERP1TYP (MERC1TYP and/or MERT1TYP, if desired) sysparm should specify one of the following Media Option types:
 - For a 3480, use sysparm DYNCUNIT
 - For DYNn, use sysparm DYNnUNIT
 - For DYnn, use sysparm DYnnUNIT

The DYNnUNIT/DYnnUNIT sysparm, related to the previous Media Options, should be specified with one of the cartridge device types listed in the section Simulate Mode Considerations in the chapter "XCOPY."

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 3480/3490 devices use CARTFEET. For more information, see the sysparm descriptions of CARTCALC, CARTPRCT, and CARTFEET of the *Systems Guide*.

- To specify a unit address using the DYNnUNIT or DYnnUNIT sysparm, it must be preceded by an EBCDIC slash (/).

Processing Logic

By default, the archive data sets are sorted in alphanumeric order that is based on the volume serials. You can modify the sysparm MERGSORT to sort the data sets by least space that is used or custom user sort. All of the input selection tests are applied to determine which archive volumes (tapes or disk archive data sets) to process.

The ARCHVOLS record corresponding to each of the selected volumes is then examined (a flag bit is tested) to ensure that the volume closed properly. If the volume did not close properly, it can be in use. Do not process the volume by merge. This condition can occur if archive and merge jobs are running simultaneously. To maintain data integrity, even if the jobs are running concurrently over midnight, CA Disk merges an *open volume* only if its creation date is less than today's date minus one (CREDIT < TODAY-1).

The file number (position) sorts the DSNINDEX records within each archive volume. This permits all data sets being merged to be located by *skipping forward*, whether this is a tape spin or a seek of the disk heads. Each unexpired data set is merged forward to a new archive volume. The data set index entry is updated to reflect its new location (live mode only). The data set expiration date with the LIMITS= parameter determines which of the new output devices is used. If its expiration date is greater than the highest limit specified, it is merged to the archive data set associated with the highest limit.

Note: Since SMS-managed DSNINDEX records have an expiration date of 1999.365, they are always written to the ARCHVOL associated with the highest limit.

Index entries for expired data sets are deleted. When all index entries for a volume have been either deleted or updated to reflect their new locations on a new archive volume, the old ARCHVOLS record is deleted. When any of the moved data sets are to be restored, CA Disk calls for the new archvol.

Each archive volume expiration date that merge creates is taken from the sysparm value for DYNEXPDT. Also, the date can come from the value you specified in an overriding JCL. The default value of DYNEXPDT is the *never expire* Julian date of 99365. If you specify a different non-zero value, it is used. In an EDM environment, Leave DYNEXPDT at its default value, which allows DSNINDEX and ARCHVOL records to expire correctly.

For more information, see the section Year 2000 Considerations in the chapter "General Information."

To determine the average length of archive tape volumes being used, merge processing relies on the specification of sysparms. The information determines the amount of space that is used on each ARCHVOL. Be sure to review sysparms CARTCALC, CARTFEET, CARTPRCT, and TAPEFEET, of the *Systems Guide*, and include them in the SYSPARMS member of parmlib if needed.

Note: The data set index entry expiration date determines candidates to merge forward. This means that if the EXPDT or RETPD parameters were hard-coded on the //ARCHIVE DD statement during an archive run, you can have an unexpired data set on an expired archive volume. The unexpired data set is still called for in merge processing to ensure the integrity of merge. No unexpired data set is dropped. If the expired tape has been written over or the archive data set on disk has been deleted, errors result when trying to locate the data set to be merged. When you use hard-coded JCL expiration dates, run the VOLDELETE command of IXMAINT before running merge to prevent the previous problem.

Merge processing creates report records for all volumes and data sets processed, indicating the actions that are taken. They are created concurrently with the MERGE and INDEX UPDATE step, but must be resorted into the proper sequence for the report writer.

JCL - MERGE Command

The MERGE command shown with the sample JCL below selects as input to the merge process those archive volumes that are less than 70 percent used. (*Archive volumes* refer to both archive tape volumes and disk archive data sets. *Less than 70 percent used* means that the unexpired data sets occupy less than 70 percent of either (a) the length of the tape, or (b) the disk space allocated to the disk archive data set.) The unexpired data sets from these selected volumes will be moved to one of three new output units, based on expiration date. Those moved to either of the first two expiration groups will have a duplicate copy created concurrently. Those moved to the third (highest) expiration group will not have a duplicate created. The following illustration is a sample JCL for the MERGE Command:

```

EDIT .CNTL(MERGE) - 01.00          Columns 00001 00072
Command ==>          Scroll ==> CSR
***** Top of Data *****
000100 //JOB CARD JOB (XXXX,XXX), 'NNNNNNNN', CLASS=X, MSGCLASS=T, NOTIFY=&USERID
000200 //TAPEMERG EXEC MERGE
000300 //MERGE.SYSPARMS DD *
000400 MERP1NAMDMS.MERGPR1
000500 MERP1TYPDISK
000600 MERC1NAMDMS.MERGCOP1
000700 MERC1TYPTAPE
000800 *
000900 MERP2NAMDMS.MERGPR12
001000 MERP2TYPTAPE
001100 MERC2NAMDMS.MERGCOP2
001200 MERC2TYPTAPE
001300 *
001400 MERP3NAMDMS.MERGPR13
001500 MERP3TYP3480
001600 MERC3TYPNULL
001700 //MERGE.SYSIN DD *
001800 MERGE PERCENT=70,LIMITS=(90,180,365)
***** Bottom of Data *****
Aa A TCP/IP R 4 C 15 14:40 3/02/98

```

MERGE Condition Codes

The following condition codes are returned from the stepname MERGE during the MERGE process:

0	Successful execution
1	Successful execution (extra copy step needed)
4	Warning messages issued
5	Warning messages issued (extra copy step needed)
8	Error terminated processing
9	Error terminated processing (extra copy step needed)
16	Critical error caused merge termination

MERGE Command

Following is a description of the MERGE command and its associated parameters. Only one MERGE command can be supplied per execution.

MERGE Command Syntax

The MERGE command syntax is:

```
MERGE SIMULATE, TYPES=, LIMITS=, PERCENT=, DATE=, EXPDAYS=, TAPEPULL=, MULTVOL, MAXVOLS=,
TDSNAMES=, NOMERGEIN, INCLUDE=, RINCLUDE=, EXCLUDE=, FCREDIT=, FDAYSOLD=, CREDIT=,
DAYSOLD=, LISTALL, COPY, UNCATALOG, DEVTYPE=, PRIMARY2, FAST=, MSGBYPAS, MINVOLS=, USEARCH
VOL, VBACKUPVOLS
```

SIMULATE

This parameter causes MERGE to execute without mounting any tape volumes or moving any archived data sets. It will process the DSNINDEX and ARCHVOLS files and produce reports indicating what would have occurred in a live run.

TYPES=

This optional parameter can be specified to limit processing to those archive data sets residing on specified device types. You can specify TAPE, DISK, 3480, DYN1 – DY20, or any combination. If this parameter is omitted, processing defaults to all device types.

LIMITS=

This optional parameter governs the distribution of data sets on the new output volumes. It specifies a list of one to five limit values of either Julian dates or retention periods in ascending sequence. The retention periods are 1- to 5-digit numbers and the Julian dates are 7-digit dates (that is, yyyyddd). Each value represents a boundary for the data sets that are to be merged to that output group. When a data set's expiration date is greater than a boundary limit, it is merged to the next output group, with the exception of the last boundary limit. If a data set's expiration date exceeds even the last boundary value, it will still be merged to the last output group. (A maximum of 5 entries is supported.)

Fast Archived datasets, with varying expiration dates that are input to Merge, when using LIMITS= require special handling to prevent them from going to differing output volumes. For this condition with LIMITS=, Merge saves the output volume assigned to the first of the Fast group and insures that same output volume is assigned to all other members of the group.

Note: SMS-managed data sets will always be written to the ARCHVOL associated with the highest limit. However, since their expiration date is dynamic (that is, it can be changed by alterations to the Management Class, or by adding more backup copies into the files data set, and so on.), there is really no validity in using the LIMITS parameter on these data sets.

PERCENT=

This optional parameter indicates the minimum percent of used space that is to be maintained on each archive volume. When the unexpired data sets on a volume account for less than this percentage of the total volume, the volume is selected for merge processing.

PERCENT=0 — Select no archive volumes based on the PERCENT parameter. Use the INCLUDE parameter for input volume specification.

PERCENT=nn — Select the volume when its percent used drops below nn.

PERCENT=100 — (default) Process all archive volumes regardless of the space used on them.

Note: If you specify the USEARCHVOL parameter, the value that is used for the PERCENT test may differ from the value that would be calculated when USEARCHVOL is not specified. This difference is the result of USEARCHVOL using the percent utilization field (ARCPCTUS) from the ARCHVOL record. For more information, see USEARCHVOL.

DATE=

This optional parameter can specify that a date in an accepted CA Disk format be used instead of the current date in determining which archived data sets have expired.

EXPDAYS= nnnn

This parameter is provided to exclude archive volumes from the merge processing if either the archive volume will expire within nnnn days or all of the data sets on the archive volume will expire within nnnn days (where nnnn is a number of days from the current date or the date specified on the DATE parameter).

The current date plus this number of days is compared to the archive volume expiration date and the highest expiration date of all the data sets on the volume. If the date is greater than either of these expiration dates, the volume is excluded from MERGE processing.

TAPEPULL=

This is an optional parameter. See Operation Concerns and TAPEPULL Processing, for a complete description of the TAPEPULL support, including the need for the //TAPEPULL DD statement.

Specify TAPEPULL=GEN to generate a hardcopy list of the archive tape volumes selected for merging; that is, the input tapes that must be pulled from the tape library. This list is also saved in a disk data set and used with the TAPEPULL=USE option as described below. Normally the SIMULATE parameter is supplied along with this parameter.

Specify TAPEPULL=USE to limit processing to those input volumes listed when the TAPEPULL=GEN parameter was specified on a previous execution. The only additional optional parameters that are valid when processing in this mode are COPY, LIMIT, and EXCLUDE.

MULTVOL

This optional parameter allows multi-volume tape archive volumes to be eligible for Merge. Other parameters can over-ride MULTVOL. By default, Merge bypasses multi-volume tape archive volumes.

MAXVOLS=

xxx

This optional parameter can specify a maximum number of input volumes to be processed for this execution of MERGE, which is 999,999 volumes. Default value for this parameter is 150.

Multi-volume archvols are a single data set, so they only count as one input volume.

TDSNAMES=

dsnamelist

This optional parameter can specify 1 to 20 archive volume data set names or patterns for which processing is to be limited.

NOMERGEIN

This optional parameter causes archive volumes created during previous merge runs to be excluded (unless also in an INCLUDE list).

INCLUDE=

A list of one or more archive volumes or patterns (or keys to disk archive data sets) to be included in the MERGE input without regard to their eligibility for other reasons. Their presence in this list overrides even the EXCLUDE and NOMERGEIN parameters.

The volumes specified in this parameter will be processed as well as those selected as a result of the PERCENT parameter. Specify PERCENT=0 to cause only these volumes to be processed. (A maximum of 100 entries is supported.)

RINCLUDE=

A list of one or more real volumes or patterns to be included in the MERGE input without regard to their eligibility for other reasons. All archive data sets residing on the real volumes specified will be included. The RINCLUDE parameter operates similar to the INCLUDE parameter in that it overrides all other selections except TYPES*. It can be used independently or in conjunction with the INCLUDE parameter. Multi-volume archive data sets will not be processed unless the first volume is specified in the RINCLUDE.

The volumes specified in this parameter are processed as well as those selected as a result of the PERCENT parameter. Specify PERCENT=0 to only process these volumes. (A maximum of 100 entries is supported).

EXCLUDE=

A list of one or more archive volumes or volume patterns (or keys to disk archive data sets) to be excluded from MERGE processing, unless in an INCLUDE or RINCLUDE list. A maximum of 50 entries is supported.

FCREDT=

This optional parameter can be specified to limit the scan of ARCHVOLS records to those whose creation date is greater than or equal to (*from or since this creation date*) the date specified via this value. The date value must be in an accepted CA Disk format. This parameter and the FDAYSOLD= parameter are mutually exclusive. Do not specify both on the same MERGE command or the command will be rejected.

FDAYSOLD=

This optional parameter can be specified to limit the scan of ARCHVOLS records to those whose creation date is greater than or equal to the current date (or an alternate date specified in the DATE= parameter) minus the number of days specified on this parameter. The maximum value for this parameter is 9999. This parameter and the FCREDT= parameter are mutually exclusive. Do not specify both on the same MERGE command or the command will be rejected.

CREDT=

This optional parameter can be specified to limit the scan of ARCHVOLS records to those whose creation date is less than or equal to the date specified via this value. The date value must be in an accepted CA Disk format. This parameter and the DAYSOLD= parameter are mutually exclusive. Do not specify both on the same MERGE command or the command will be rejected.

DAYSOLD=

This optional parameter can be specified to limit the scan of ARCHVOLS records to those whose creation date is less than or equal to the current date (or an alternate date specified in the DATE= parm) minus the number of days specified on this parameter. The maximum value for this parameter is 9999. This parameter and the CREDT= parameter are mutually exclusive. Do not specify both on the same MERGE command or the command will be rejected.

LISTALL

Normally report records and messages generated by this function are produced only for the volumes actually processed. If you want to print statistics and messages for volumes not processed, specify this parameter on the MERGE command.

COPY

The primary output data sets created by merge can have duplicate and/or tertiary copies created concurrently, often referred to as triplexing. (This is controlled by the dynamic allocation sysparms or by providing the proper JCL directly.) This COPY parameter indicates whether or not an additional copy is to be produced in an extra job step following the primary merge.

If duplexing is performed in the primary merge, this parameter causes the keys of the duplicate copies to be written to a disk data set and passed to the subsequent step where they can be copied again, creating tertiary copies. If duplexing is not done, the key of the primary (instead of the duplicate) is passed to the copy step.

This parameter merely causes the keys of the appropriate new archive volumes to be written (saved) in a disk data set. You must request procedure MERGCOPY directly after the MERGE procedure if you want to actually execute the extra copy step. An ARCHVOLS record will be made for each new copy volume and properly chained from the primary (or duplicate) ARCHVOLS record.

UNCATALOG

By default, MERGE does not change the catalog status of any data set that it merges. Specify this UNCATALOG parameter to have CA Disk uncatalog data sets that are cataloged for auto-restore purposes after each data set is successfully merged.

You can specify this parameter on your disk-to-tape Merge runs if you wished to prevent auto-restore from tape, but allow it from disk.

Before specifying this parameter, see the section DSNDELETE Command "FILES Maintenance" and the section BACKUP Command in the chapter "Backup/Archive" regarding the combination of sysparm UNCATDEL and an expiration date of 1999.000.

DEVTYPE=

By default, MERGE processes all device types. This optional parameter can be used to limit processing to only those DSNINDEX records residing on a particular device type. Possible values are: 3380, 3390, 3480, 3490, REDW (StorageTek's Redwood device) or 3590 (IBM's Magstar device).

PRIMARY2

This optional parameter causes only PRIMARY2 archive volumes to be selected for Merge. No other parameters can override PRIMARY2. By default, Merge bypasses PRIMARY2 archive volumes.

FAST=

Specify YES or NO to indicate whether the data set selected for processing should use the Fast Merge path.

MSGBYPAS

Specify this parameter on the MERGE command to show the ARCHVOLS and the reason why they were bypassed for processing.

MINVOLS=xxx

This optional parameter can specify a minimum number of input volumes to be processed for this execution of MERGE. Default value for this parameter is 0.

USEARCHVOL

This optional parameter causes MERGE processing to bypass reading the DSNINDEX records in order to determine the actual percent utilization of the archive data set. The percent utilization field (ARCPCTUS) in the ARCHVOL record is used instead in the selection process. Because ARCPCTUS is updated during IXMAINT, the USEARCHVOL parameter can provide a significant performance improvement by not reading the DSNINDEX record. The longer is the period after IXMAINT was run, the larger the percent difference between MERGE with and without USEARCHVOL could be. This option is only available when a Files Data Base is being processed.

VBACKUPVOLS

This optional parameter causes only VBACKUP volumes to be selected for MERGE. By default, MERGE bypasses VBACKUP volumes.

Note: If the PACK parameter is not used for VBACKUP, sysparm ARCHPACK must be set to N to maintain the one VBACKUP volume data set to one Archvol relationship. Using ARCHPACK set to Y will put multiple VBACKUP data sets on one ARCHVOL.

Sample MERGE Reports

The following is a sample MERGE Report:

DATA SET NAME	ARCDATE	KEY	FILE	KILO-BYTES	BLOCK COUNT	EXPDATE	ACTION TAKEN	NEW KEY	NEW FILE	OUT-DO
MONMA10.ANZ.BIN	2006.023	576123	1	4	1	2006.028	MERGED	576673	1	1
MONMA10.ANZ.BIN	2006.023	576126	1	4	1	2006.028	MERGED	576673	13	1
MONMA10.ANZ.REPORT1	2006.023	576123	2	43383	1357	2006.028	MERGED	576673	2	1
MONMA10.ANZ.REPORT1	2006.023	576123	2	43383	1357	2006.028	MERGED	576673	2	1
MONMA10.ANZ.REPORT2	2006.023	576123	3	23	1	2006.028	MERGED	576673	3	1
MONMA10.ANZ.REPORT2	2006.023	576123	3	23	1	2006.028	MERGED	576673	3	1
MONMA10.ANZ.REPORT3	2006.023	576123	4	54	2	2006.028	MERGED	576673	4	1
MONMA10.ANZ.REPORT3	2006.023	576123	4	54	2	2006.028	MERGED	576673	4	1
MONMA10.ANZ.REPORT4	2006.023	576123	5	32	1	2006.028	MERGED	576673	5	1
MONMA10.ANZ.REPORT4	2006.023	576123	5	32	1	2006.028	MERGED	576673	5	1
MONMA10.ANZ.REPORT5	2006.023	576123	6	54	2	2006.028	MERGED	576673	6	1
MONMA10.ANZ.REPORT5	2006.023	576126	2	54	2	2006.028	MERGED	576673	14	1
MONMA10.A300.JCLLIB	2006.023	576123	7	6	1	2006.028	MERGED	576673	7	1
MONMA10.A300.JCLLIB	2006.023	576123	7	6	1	2006.028	MERGED	576673	7	1
MONMA10.BASE90.INSTALL	2006.023	576123	8	359	12	2006.028	MERGED	576673	8	1

SMS and MERGE

MERGE processes DSNINDEX records of SMS data sets like all other DSNINDEX records that contain 1999.365 in their EXPDATE field. In other words, since SMS managed data sets have 1999.365 specified; SMS managed data sets are considered never expire for MERGE processing. It is IXMAINTs responsibility for the expiration or the deletion of SMS data sets.

MERGE does, however, use the last restore date to determine the number of level 1 days non-used. This determination is used in the merging forward process.

The SMS field that CA Disk supports is as follows:

- Level 1 Days Non-usage (MC) This optional field can be defined 0 to 9999, NOLIMIT, or blank. The default is 60. When this field is defined with a number other than 0 (see the section SMS and ARCHIVE in the chapter "Backup/Archive" for more information about 0), that number of days must pass since the last Restore/Recover access from a disk archive of the data set before the data set is eligible for merge to tape (level 2). When this field is defined NOLIMIT, the data set remains on disk archive (level 1). When this field is defined with a blank, other MERGE command parameters will be used to process the data set.

RESIXRPD Concern

The use of this sysparm may cause adverse affects on SMS managed data sets. In situations where you don't want to keep multiple archive DSNINDEX records (that is, 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 but 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 further 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).

Chapter 12: MOVE/COPY

This section contains the following topics:

[About Move and Copy](#) (see page 471)
[General Information](#) (see page 472)
[Moving Data to Different Device Types](#) (see page 476)
[Special Considerations](#) (see page 476)
[Reblocking](#) (see page 481)
[User Exits](#) (see page 483)
[MOVE Command](#) (see page 483)
[COPY Command](#) (see page 487)
[VSAM Cluster Considerations](#) (see page 491)
[SMS and MOVE/COPY](#) (see page 493)
[Sample MOVE/COPY Commands](#) (see page 494)
[Sample Move/Copy Report](#) (see page 495)

About Move and Copy

The Move/Copy function provides the capability to migrate data sets directly from one DASD volume to another by using a single, common command format. This frees the user from unnecessary involvement with different data set organizations, attributes and device types. Data sets are selected through the use of DSCL selection criteria. DSCL provides a powerful and convenient means to configure all or portions of many volumes. Migration capabilities include CA Disk archive to disk data sets.

Move/Copy can process both VSAM and non-VSAM data sets, whether they are SMS-managed or not. It can be used to move or copy non-SMS data sets to SMS volumes and make them SMS managed. ACS rules are used to determine the data set placement for SMS data sets. In some cases, SMS data sets can be redirected back to their original Storage Class or Storage Group. To ensure this will not happen, review your ACS rules carefully.

Commands Available: MOVE (Implicit/Explicit) and COPY (Explicit)

JCL Proc Executed: DMS

General Information

The basic element of this function is the ability to migrate data sets directly from one DASD volume to another using a single common command format, freeing the user from unnecessary involvement with different data set organizations, attributes and device types. This basic element is also combined with the ability to scan VTOCs from one or more volumes and process all or subsets of the data sets, to provide a powerful and convenient means to configure all or portions of many volumes. Also supported is the migration of OS catalogs (non-VSAM) that execute at several times the speed of IEHMOVE. For all Move/Copy operations, 3390 and 3390-2 devices are considered to be *like* devices.

Error Recovery

If copy processing fails, the target data set will be deleted and the source data set will be unchanged, as before processing started. This is an automatic process, which requires no work on the part of the user.

If move processing fails, the target data set will be deleted and the source data set will be recovered to its original status. This is an automatic process which requires no work on the part of the user.

Note: This process is done in the AMSEXEC and AMSSORT steps.

AMSSORT and AMSEXEC Steps

If CA Disk abends during move processing, the source data set is still recovered to its original state. This capability is provided by a recovery facility executed in the AMSSORT and AMSEXEC steps of the DMS procedure. These steps are only executed in the event of an abend during the move process.

The AMSRFILE dd statement provides input to these two steps. The AMSEXEC step will rename any data set left in a suspended state due to an abend based on the information provided by AMSRFILE. By default, the disposition of the AMSRFILE is a temporary data set.

If a system IPL or a S222 abend occurs, you would have to manually locate the name of the source and target data sets, delete the target and alter the source back to the original name. To assist you in locating these data sets, CA Disk issues informational message 4017. Or you can simplify the cleanup process by creating a GDG data set to use for the AMSRFILE dd statement. This will allow the AMSRFILE to remain after an IPL or job cancel, and will provide you with the data set name to be renamed as well as the target data set name to be deleted during your manual cleanup.

Graceful Shutdown

MOVE and COPY jobs can be shutdown gracefully by using sysparm DSCLRPLY. A graceful shutdown will allow the jobs to complete all work in progress and produce the normal reports for error conditions encountered and work successfully completed before the jobs terminate.

With DSCLRPLY active, a message is issued to the operator at the beginning of the job and remains on the console for the duration of the job. If the operator replies, *END* to this message, it indicates processing is to be stopped early. For details, see the section DSCL Graceful Shutdown in the chapter "MOVE/COPY."

ENQ/DEQ

By default, exclusive access to a data set is maintained for the duration of its use for move processing. This ensures that needed catalog and scratch actions can be completed successfully. If exclusive control cannot be obtained (other users still accessing the data set) in order to do the processing, an appropriate bypass message will be issued.

By default, shared access to a data set is maintained for the duration of its use for copy processing. If shared control cannot be obtained (another user accessing the data set exclusively) in order to do the processing, an appropriate bypass message will be issued.

Scratch/Catalog Defaults

When a data set is moved, the default action is to scratch the source data set and recatalog it to the target device if it was cataloged previously. Uncataloged data sets are left uncataloged. The CATALOGALL parameter can be used to cause these to be cataloged as well.

Target Allocation

If more than one target volume is specified, the default allocation technique is to allocate each successive data set processed to the next volume in the list. A second technique of filling each volume in the list before switching to the next volume can be selected by specifying the FILL parameter.

Standard DADSM allocation facilities are used that allow a primary space request to be split into as many as five extents before issuing the SPACE NOT AVAILABLE message. Contiguous space allocation (only one extent allowed) can be requested by specifying the CONTIG parameter on the MOVE or COPY command.

For both MOVE and COPY, no attempt is made to do exact positioning of data sets on the target volume, that is, standard DADSM allocation rules are in effect: the smallest free space extent large enough to contain the new data set is used. This means that for a new pack with the VTOC at the beginning, each data set will be placed as close to the VTOC as possible, moving progressively down the pack. If exact placements for certain data sets are desired, they can be preallocated. CA Disk will ensure that all attributes in the format-1 DSCB are properly set.

Note: When processing VSAM, the MOVE command relies on SRM for volume selection. For more information, see the parameter description for NEWVOLUMES.

Excluding Data Set Names from Migration

Data sets that are to be exempted from move or copy processing should have their data set names or patterns placed in a member in the parmlib data set. Any member name not previously used can be defined for this purpose. To invoke the use of this member, specify its name as the value for the TABLE= parameter, of the DSCL selection criteria.

Each entry consists of one data set name or data set name pattern (for example, *SYS* / or *USA.SOURCE.LIB*).

Renaming Data Sets

Before you move a data set, CA Disk dynamically links to IDCAMS and issues an ALTER NEWNAME against the original data set name or VSAM cluster. As a result, the last character in the data set name or VSAM cluster name increases by two alphabetic characters each time. This increment allows CA Disk to define the data set or cluster on the target volume with the original name or cluster name.

For example, while moving cluster *userid.APPLE.CLUSTER*, CA Disk issues an ALTER NEWNAME, resulting in the following name change:

userid.APPLE.CLUSTET

Messages

Messages are normally issued only for error conditions encountered or for special warnings. Data sets being bypassed due to the parameters specified and system defaults in effect are generally not mentioned. Sysparm VCBYPASS can be specified with a value of Y to request additional messages stating why a data set was skipped by the MOVE command; these messages are always printed by the COPY command.

The dd statement DIAGDTOC can also provide some helpful diagnostic messages while moving VSAM.

Restrictions

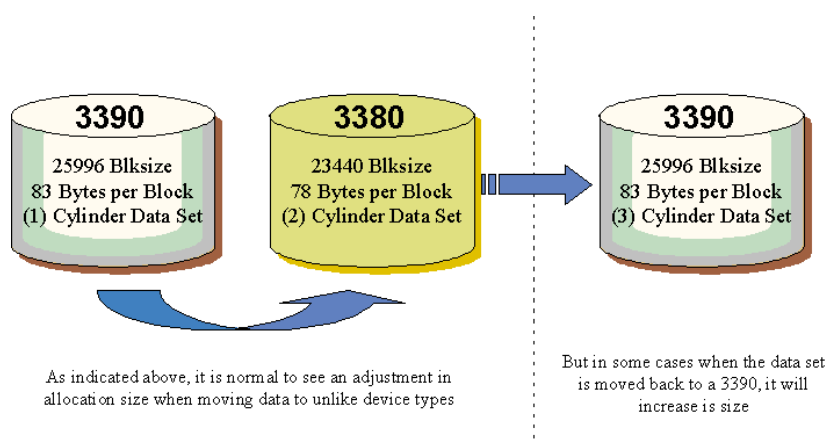
The following is a list of restrictions that apply solely to the Move/Copy function. Additional restrictions are documented in the section General Restrictions in the chapter "General Information."

- Load libraries are not reblocked.
- ISAM data sets will be allocated as a single prime area, with the index and overflow suballocated. If you want to maintain separate areas for each, you must preallocate them as desired, prior to the migration.
- Disconnected aliases in a PDS will not COPY/MOVE to a pre-allocated PDSE.
- PDS members with user TTRNs in their directory entries will not COPY/MOVE to a preallocated PDSE.
- NEWNAME is not allowed on the COPY command if the source allocation is PO and target allocation is PDSE.
- APFLST or LNKST data sets. CA Disk will bypass unless explicitly selected by data set name.

Moving Data to Different Device Types

Whenever data sets are being moved between different device types, CA Disk attempts to adjust the required space allocation accordingly, based on the BLKSIZE, RECFM, and DSORG for each data set. The accuracy of the space conversion depends on how well the actual data reflects the specified attributes.

For example, when a data set is moved from a 3390 to a 3380, an allocation adjustment to the target 3380 must be made. This adjustment is necessary because the track size of a 3390 is larger than that of a 3380. In this particular case, the allocation size of the target data set will most likely increase. In fact, it is normal for a 1 cylinder 3390 data set that is moved to a 3380 to increase to 2 cylinders on the target volume, as illustrated as follows:



However, if CA Disk under-allocates space during a device conversion, you can want to consider using one or more of the following sysparms:

ADPCTPDS
ADPCTOTH
ADTRACKS

For more information, see Space Allocation Algorithm in the *Systems Guide*.

Special Considerations

Move/Copy processes several types of data sets, but each type is processed in a unique way. The following discussion describes these differences.

GDG Data Sets

When moving or copying GDG data sets, the following principles will be followed in regards to catalog action:

NEWNAME:

When this parameter is specified, COPY treats the target DSNAME as a new data set creation. The target data set will always be cataloged and placed in the sphere, whether the sphere is full or not.

For non-SMS data sets without NEWNAME:

- If a catalog entry exists, the entry is left intact.
- If no catalog entry exists, the data set will be cataloged unless doing so would force another entry out of the catalog (that is, the sphere is full).

For SMS data sets without NEWNAME:

1. If a catalog entry exists at the time of the move/copy operation, and it has a valid status, the status will be left intact. The only exception is when the original data set is in ROLLED-OFF status, the data set is placed in DEFERRED status on the new volume.
2. If a catalog entry exists but a valid status cannot be detected, or if a catalog entry does not currently exist, the status is assumed to be ACTIVE, and CA Disk will attempt to roll in the data set, unless doing so would force another entry out of the catalog.
3. If the NEWNAME parameter is being used, the status from the original data set will be used to determine catalog action.

If CA Disk cannot set a data set's catalog entry to the status CA Disk feels is correct, message 3729 will be issued to indicate the status the data set was left in, and the reason for placing the data set in that status.

For more information, see the User Exit description for GDGCATEX —GDG Data Set Catalog Action in the *Systems Guide*.

PSE Data Sets

Extended sequential data sets are simply multi-volume sequential data sets that are managed by DFSMS. This combination resulted in some special considerations that each user should be aware of. See the section Special Considerations While Processing PSEs in the chapter "Basic System – DMS PROC" for a complete list of these considerations.

Model DSCB Processing

Data sets with zero extents allocated are most commonly acting as model DSCBs and should be treated separately from other data sets. Depending on how your DSCL selection criteria are set up, zero extent data sets can be automatically selected. To prevent selection of model DSCBs, use the DSCL EXCLUDE command with MODEL DSC criteria parameter.

Note: Sometimes model DSCBs do not have valid DSORG information associated with them and CA Disk can bypass them for this reason. Consider specifying sysparm VCUDSORG with a value of Y to allow them to be moved.

BDAM Data Sets

Special considerations apply to the migration of BDAM data sets. The type of addressing used, indicated by the OPTCD field of the DSCB, dictates the migrate options available as described below.

Relative Block Addressing

No restrictions with respect to the combination of the source and target device types.

TTR Addressing

can be migrated between devices of the same type. Attempts to migrate between unlike devices can result in diagnostic messages and the data set being bypassed, or the migration can be successful, depending on the device types and the data involved.

For more information, see the sysparm description for HONORTTRn in the *Systems Guide*.

Absolute Addressing

A preallocated data set that meets the absolute address needs is required. If this condition is satisfied, the data set can be copied. To force awareness that these special data sets are being copied, the ABSOLUTE and PREALLOC parameters must be used. Sysparm UNMOVEVC must also be specified.

RECFM=U OR V

BDAM data sets of this format are not preformatted with fixed-length records, and can contain either completely empty or partially empty tracks embedded within. They use either TTR or absolute CCHH addressing techniques, and can be migrated between devices of the same type. Attempts to migrate between unlike devices can result in diagnostic messages and the data set being bypassed, or the migration can be successful, depending on the device types and the data involved.

The empty or partially empty status of the contained tracks is maintained. For more information, see the section Direct Access and Physical Sequential Data Sets in the chapter "General Information."

SYSCTLG Processing

SYSCTLG data sets can be migrated between like or unlike devices at several times the speed of IEHMOVE. A data set is treated as a system catalog when either the source and/or the target data set name is SYSCTLG. This permits the backing up of SYSCTLG data sets to DASD at high speed with the option of renaming the backup copy.

The capacity of the catalog can be changed significantly by preallocation of a larger SYSCTLG on the target volume. The migration support will format the additional space in the target allocation with empty catalog blocks, and update the TTR in the volume index control entry to reflect the additional capacity.

When a catalog is migrated between unlike device types, all TTR pointers in the catalog are converted. The conversion includes the validation of the TTR contained in the source catalog, and can detect either an invalid TTR or catalog entry type. These conditions cause the messages 425 and/or 426 to be issued, and the invalid block to be copied forward without modification. Although the invalid blocks are most likely disconnected, a machine comparison of the output from an IEHLIST of both the source and target catalogs can be used to validate the processing.

CA Disk does not disconnect the CVOL catalog from the VSAM master catalog. It also does not reconnect the CVOL catalog to the VSAM master catalog. The user with IDCAMS utilities must do this manually. For this reason the user should always consider the status of the catalog when CA Disk is moving it. Where are the catalog entries for the data sets being moved? If the catalog is being moved in the same run, it is possible that some data sets will lose their new catalog status because the catalog was copied to a new volume, but data sets copied subsequent to the catalog move were still being cataloged in the old copy of the catalog.

It is recommended that CVOL catalogs be moved as a separate operation, either before or after all other data sets are moved, but not within the same process. After a CVOL catalog has been moved, ensure that all of its connecting entries to the master catalog are rebuilt immediately.

To assist you in disconnecting and reconnecting a CVOL catalog that has been moved to a new volume, the following sample commands for the IDCAMS utility are provided. Consult your IBM IDCAMS manual for more specific information.

To delete high-level index pointers from your master catalog to your old CVOL (on the old volume), use commands similar to the following for each high level index:

```
DELETE (INDEX1) ALIAS
```

To define a new CVOL catalog in your master catalog:

```
DEFINE NONVSAM (NAME(SYSCTLG.V_____) DEVT( ) VOL( ))
```

To reconnect the high-level index pointers to the CVOL catalog on the new volume:

```
DEFINE ALIAS (NAME(INDEX1) RELATE(SYSCTLG.V_____) )
```

Reblocking

An additional feature of the Move/Copy function is the option to reblock sequential, partitioned, indexed sequential and direct access data sets that consist of either fixed- or variable-length records.

Note: RECFM=U, LOAD LIBRARIES, cannot be reblocked.

The maximum block size to be given to the migrated data sets is specified as a limit value to Move/Copy. As each data set is migrated, it is reblocked to the largest block size less than or equal to the limit value. Either a global limit can be specified to apply to all data sets of the previous types, or different limits can be given for various combinations of device types, DSORG and RECFM. To define the different limit values, a list of reblocking entries must be placed in a member in the parmlib data set, each entry in the format as defined below.

When reblocking keyed data sets, the actual reblocked size can be less than the limit specified. If the new block size plus the key length will not fit on a track of a particular device type, CA Disk will attempt to reduce the block size. If this is not possible, the keyed data set will not be reblocked.

As an aid when doing reblocking, sysparm NOBLKMSG is provided to request explanatory messages when a data set cannot be reblocked.

Note: When the target data set is marked as reblockable as indicated by the DS1REBLK bit in the FORMAT 1 DSCB and your system is at DFP level 3.1 or higher, the use of the sysparm REBLKBIT=Y and the BLKSIZE= parameter under either MOVE or COPY are mutually exclusive.

If you would like CA Disk to reblock data sets that were allocated prior to DFP 3.1, you can use the DSCB Update Utility (described in DSCB Update Utility in the *Installation Guide*) to turn the DS1REBLK bit on. This will allow system defined reblocking to occur.

Specifications

Reblocking entries are placed in a parmlib member of the user's choice. The parmlib member name is specified in the BLKSIZE= parameter of the COPY and MOVE commands. Each parmlib member entry has the following format.

```
'AAAAAABBBBCCD'
```

Where:

AAAAAA	The six-character generic device name must be one of the following. This field is left-justified.			
	2305-1	2305-2	2314	3330
	3330-1	3340	3350	3375
	3380	3380E	3380K	3390
	33902	33903	33909	9340
	F6421			
BBBBB	The limit blocksize must be specified as a five- digit number with leading zeros as appropriate.			
CC	Optional, but if present, must specify one of the following DSORGs: ■ IS Indexed Sequential ■ PO Partitioned ■ PS Sequential			
D	Optional, but if present, must specify one of the following RECFMs: ■ F Fixed ■ V Variable			

When reblocking lists are used, the only required fields are device type and limit blocksize (DSORG and RECFM are optional). Several entries can be given for each device type in order to specify different limits for various DSORG and RECFM combinations. The first entry found that will apply to a selected data set is used. If no applicable entry is found, the data set is migrated without reblocking.

Note: Reblocking entries should always be enclosed in quotes when putting them into the parmlib member. This ensures that truncation does not occur as a consequence of embedded blanks in the device name field.

User Exits

Specifics regarding the use of user exits can be found in User Exits in the *Systems Guide*. Only a brief description is presented here.

DSCLSCEXmmmmmmmmmm

This exit is given after DSCL has selected the data set for move or copy processing. The exit can allow processing to continue or stop processing for each data set. This exit is called for both VSAM and non-VSAM data sets. For complete details of this user exit, see DSCLSCEX —DSCL Screening Exit in the *Systems Guide*.

VCCOPYEXmmmmmmmmmm and VCDMIGEXmmmmmmmmmm

Each of these exits is given control just prior to the migration of a non-VSAM data set. The first exit is used by COPY command processing, the second by MOVE. CA Disk has done all selection screening and the user's exit module is given a chance to override the migration. For complete details of these user exits, see VCCOPYEX —Move/Copy Screening in the *Systems Guide*.

MOVE Command

Using the DSCL MOVE action command moves data sets. Selection of data sets to be processed is determined by use of the DSCL selection criteria.

MOVE Command Syntax

The MOVE command syntax is:

MOVE ABSOLUTE, BLKSIZE=, CATALOGALL, CONTIG, CREATE, CYL, FILL, PREALLOC, PDSECONV,
MERGE, REDUCE, TOVOL=, TRACK, VATHRESH=, NEWVOLUMES, STORCLAS=, MGMTCLAS=, DATACLAS=

ABSOLUTE

This optional parameter is provided to permit the override of the default action of not copying a BDAM data set with absolute addressing. This parameter requires that the data set be preallocated on a like device, and that the PREALLOC parameter also be specified. This parameter is applicable only to non-VSAM data sets.

BLKSIZE=

When reblocking is desired, this optional parameter must be specified in one of the two following ways:

- A 1- to 5-digit number to be used as a global limit for the new block sizes.
- A 1- to 8-character name of a member containing reblocking limit block sizes by device type, DSORG and RECFM.

When reblocking keyed data sets, the reblocked size can be less than the limit block size specified. For more information, see Reblocking. This parameter is applicable only to non-VSAM data sets.

CATALOGALL

The presence of this parameter will cause even uncataloged source volume data sets to be cataloged to the target volume. Default processing leaves uncataloged data sets uncataloged, recataloging only those data sets that are cataloged at the start. This parameter is applicable only to non-VSAM data sets.

CONTIG

Specify this parameter if the new data sets are to be moved to the target volume only if they can be allocated in one contiguous extent. (Default allocation uses standard DADSM allocation, which will usually result in one extent, but will allow up to five if the volume is badly fragmented.) This parameter is applicable only to non-VSAM data sets.

CREATE

Specify this parameter if the creation date for the target data set is to be set equal to the current date. Default processing leaves the original creation date unaltered. Specifying this parameter will cause the DASD Billing support to charge for the target data set from its real creation date, not from the source data set's creation date. This parameter is applicable only to non-VSAM data sets.

CYL

Specification of this parameter will force all allocations on the target volume(s) to be rounded to the next highest cylinder and the allocation type to be cylinder rather than track. This parameter is applicable only to non-VSAM data sets.

FILL

Specification of this parameter will force the allocation of data sets to the first volume in the TOVOL= parameter list until it is full before any allocations are made to the next volume in the list. The default allocation technique is *round-robin*. This parameter is applicable only to non-VSAM data sets.

PREALLOC

This parameter must be entered to permit a move to preallocated data sets on a target volume. The default action is to bypass data sets having an allocation on a target volume. When this parameter is present, only one target volume can be specified. This parameter is applicable only to non-VSAM data sets.

REDUCE

Specify this parameter when the space allocations on the target volumes should be based on amount of space used rather than amount allocated. This parameter is only applicable to PO and PS data set organizations with a non-zero secondary allocation amount. (See the TRACK parameter also.) This parameter is applicable only to non-VSAM data sets.

TOVOL=

Specify this parameter to identify a list of 1 to 20 volume names or volume name patterns to which data sets are to be moved. This parameter can be considered optional for SMS-managed VSAM data sets.

TRACK

When the REDUCE parameter is also present, this parameter causes PS and PO data sets allocated by cylinders but using less than one cylinder to have their allocations changed to tracks and the unused space released. This parameter is applicable only to non-VSAM data sets.

VATHRESH=

This parameter indicates the percent of the volume (for VSAM) that is to be filled. Specify a value from 1 to 99 that indicates the maximum threshold of space CA Disk is to use on a volume. For example, a value of 80 indicates that CA Disk can fill a volume until 80% of its space is used and then will attempt to choose other volumes for new allocations.

NEWVOLUMES

This parameter pertains to the migration of multi-volume data sets. When specified, it will indicate that none of the original volumes on which the data set resides be considered as candidate volumes when data sets are selected by volume.

- If a data set is selected without any volume selection criteria, CA Disk will only use the volumes specified in the TOVOL= parameter as candidate volumes.
- If a data set is selected and volume selection criteria is used, CA Disk will consider all volumes where the data set resides, minus the volumes it was selected for, plus the volumes specified in the TOVOL= parameter as candidate volumes.

For example, a data set resides on volumes A, B, and C. It is selected to move to volumes D, E, and F, and volume selection criteria are used (that is, SELECT VOL=A). With NEWVOLUMES specified, only volumes D, E, and F will be considered as candidate volumes. Without NEWVOLUMES specified, volumes B, C, D, E, and F will be considered.

PDSECONV

Specify this parameter and set SYSPARM SMSALLOC to N to automatically convert PDSE to PDS if the target data set is not eligible to be SMS-managed. During PDSE to PDS conversion, the number of directory blocks allocated will be as few as possible, based on the number of members in the source PDSE.

MERGE

Specifying this parameter and the PREALLOC parameter will cause the pre-allocated target PDS or PDSE to be merged by the source PDS or PDSE. The same named source PDS or PDSE members will overlay the existing target PDS or PDSE members.

DATACLAS=

For a VSAM or non-VSAM SMS managed data set, the value supplied for this parameter is the new Data Class name to be assigned to the data set at allocation time. The Data Class name supplied must be previously defined to SMS.

Note: This parameter is only valid when sysparm SMSALLOC is specified with a Y.

MGMTCLAS=

For a VSAM or non-VSAM SMS managed data set, the value supplied for this parameter is the new Management Class name to be assigned to the data set at allocation time. The Management Class name supplied must be previously defined to SMS.

Note: This parameter is only valid when sysparm SMSALLOC is specified with a Y.

STORCLAS=

For a VSAM or non-VSAM SMS managed data set, the value supplied for this parameter is the new Storage Class name to be assigned to the data set at allocation time. For non-SMS-Managed data sets, supplying this parameter will cause the target data set to be SMS-Managed. In either case, the Storage Class name supplied must be previously defined to SMS.

Note: This parameter is only valid when the sysparm SMSALLOC is specified with a Y.

COPY Command

Data sets are copied by using the DSCL COPY action command. Selection of data sets to be processed is determined by use of the DSCL FIND command.

COPY Command Syntax

The COPY command syntax is:

COPY

ABSOLUTE, BLKSIZE=, CATALOGALL, CONTIG, CREATE, CYL, MERGE, NEWNAME=, NEWVOLUMES, PDSECONV
PREALLOC, REDUCE, TARGETONLY, TOVOL=, TRACK, VATHRESH=, STORCLAS=, MGMTCLAS=, DATACLAS=

ABSOLUTE

This optional parameter is provided to permit the override of the default action of not copying a BDAM data set with absolute addressing. This parameter requires that the data set be preallocated on a like device, and that the PREALLOC parameter also be specified. This parameter is applicable only to non-VSAM data sets.

BLKSIZE=

When reblocking is desired, this optional parameter must be specified in one of the two following ways:

- To 5-digit number to be used as a global limit for the new block sizes.
- To 5-character name of a member containing reblocking limit block sizes by device type, DSORG and RECFM.

When reblocking keyed data sets, the reblocked size can be less than the limit block size specified. For more information, see Reblocking. This parameter is applicable only to non-VSAM data sets.

CATALOGALL

The presence of this parameter will cause even uncataloged source volume data sets to be cataloged to the target volume. The default catalog action for the COPY command is no action. This parameter is applicable only to non-VSAM data sets.

CONTIG

Specify this parameter if the new data sets are to be copied to the target volume only if they can be allocated in one contiguous extent. (Default allocation uses standard DADSM allocation, which will usually result in one extent, but will allow up to five if the volume is badly fragmented.) This parameter is applicable only to non-VSAM data sets.

CREATE

Specify this parameter if the creation date for the target data set is to be set equal to the current date. Default processing leaves the original creation date unaltered. Specifying this parameter will cause the DASD Billing support to charge for the target data set from its real creation date, not from the source data set's creation date. This parameter is applicable only to non-VSAM data sets.

CYL

Specification of this parameter will force all allocations on the target volume(s) to be rounded to the next highest cylinder and the allocation type to be cylinder rather than track. This parameter is applicable only to non-VSAM data sets.

MERGE

Specifying this parameter and the PREALLOC parameter will cause the pre-allocated target PDS or PDSE to be merged by the source PDS or PDSE. The existing target PDS or PDSE members will be overlaid by the same named source PDS or PDSE members.

NEWNAME=

A different name can be given to the data set on the target volume if security checking permits it. A relative GDG name is permitted. This parameter is required for VSAM data sets.

NEWVOLUMES

This parameter pertains to the migration of multi- volume data sets. When specified, it will indicate that none of the original volumes on which the data set resides be considered as candidate volumes when data sets are selected by volume.

- If a data set is selected without any volume selection criteria, CA Disk will only use the volumes specified in the TOVOL= parameter as candidate volumes.
- If a data set is selected and volume selection criteria is used, CA Disk will consider all volumes where the data set resides, minus the volumes it was selected for, plus the volumes specified in the TOVOL= parameter as candidate volumes.

As an example; A data set resides on volumes A, B, and C. It is selected to be copied to volumes D, E, and F, and volume selection criteria is used (that is, SELECT VOL=A). With NEWVOLUMES specified, only volumes D, E, and F will be considered as candidate volumes. Without NEWVOLUMES specified, volumes B, C, D, E, and F will be considered.

PDSECONV

Specify this parameter and set SYSPARM SMSALLOC to N to automatically convert PDSE to PDS if the target data set is not eligible to be SMS-managed.

PREALLOC

This parameter must be entered to permit a copy to preallocated data sets on a target volume. The default action is to bypass data sets having an allocation on a target volume. When this parameter is present, only one target volume can be specified. This parameter is applicable only to non-VSAM data sets.

REDUCE

Specify this parameter when the space allocations on the target volumes should be based on amount of space used rather than amount allocated. This parameter is only applicable to PO and PS data set organizations with a non-zero secondary allocation amount. (See the TRACK parameter also.) This parameter is applicable only to non-VSAM data sets.

TARGETONLY

Specifying this parameter forces the target allocation to the volume specified on the TOVOL= parameter. This is useful when making a copy of a data set with a different name, and you want both source and target data sets to remain on the source volume. This parameter is applicable only to VSAM data sets.

TOVOL=

Specify this parameter to identify a list of 1 to 20 volume names or volume name patterns to which data sets are to be moved. If this parameter is not specified, CA Disk will attempt to copy the data set to its original volume. This can fail if the NEWNAME= parameter is not specified or if space does not exist on the original volume.

TRACK

When the REDUCE parameter is also present, this parameter causes PS and PO data sets allocated by cylinders but using less than one cylinder to have their allocations changed to tracks and the unused space released. This parameter is applicable only to non-VSAM data sets.

VATHRESH=

This parameter indicates the percent of the volume (for VSAM) that is to be filled. Specify a value from 1 to 99 that indicates the maximum threshold of space CA Disk is to use on a volume. For example, a value of 80 indicates that CA Disk can fill a volume until 80% of its space is used and then will attempt to choose other volumes for new allocations.

DATACLAS=

For a VSAM or non-VSAM SMS managed data set, the value supplied for this parameter is the new Data Class name to be assigned to the data set at allocation time. The Data Class name supplied must be previously defined to SMS.

Note: This parameter is only valid when sysparm SMSALLOC is specified with a Y.

MGMTCLAS=

For a VSAM or non-VSAM SMS managed data set, the value supplied for this parameter is the new Management Class name to be assigned to the data set at allocation time. The Management Class name supplied must be previously defined to SMS.

Note: This parameter is only valid when sysparm SMSALLOC is specified with a Y.

STORCLAS=

For a VSAM or non-VSAM SMS managed data set, the value supplied for this parameter is the new Storage Class name to be assigned to the data set at allocation time. For non-SMS-Managed data sets, supplying this parameter will cause the target data set to be SMS-Managed. In either case, the Storage Class name supplied must be previously defined to SMS.

This parameter is only valid when the sysparm SMSALLOC is specified with a Y.

VSAM Cluster Considerations

CA Disk processes all components of VSAM clusters selected for processing. When any VSAM component name is encountered during the scan of a volume, and selection criteria causes it to be processed, it and all of its related components will be moved or copied. Related components will be moved even if they exist on volumes other than the source volume.

Move and copy processing runs above the 16meg line for VSAM data sets.

The movement of data is accomplished by two methods, control interval method and IDCAMS service.

Control Interval method is done via VSAM IO macros. After the data is moved, the target data set is a mirror image of the source data set. note that the control interval method SHOULD NOT be used to move the data to a different device type because the image of the target data set cannot be the same as the source data set. The data is moved using the control interval method in the following cases:

1. The data set is Linear or RRDS data set.
2. The data set is ESDS and the record count fields in the statistic are zero.
3. The data set is ESDS, the record count fields in the statistic are not zero, and the value of sysparm 'VSARCFMT' is 'C'.

IDCAMS moves data logically and any free space in the control interval is reconciled. In the event of an error, you can see IDCAMS error messages in your CA Disk report. These messages will supplement error messages issued by CA Disk.

The data is moved using IDCAMS services in the following cases:

1. The data set is KSDS.
2. The data set is ESDS and the record count fields in the statistic are not zero. The sysparm 'VSARCFMT' can be used to force the data to be transferred using the control interval method.

The following special considerations apply to usage of the target volume list when processing VSAM clusters:

1. When a VSAM cluster's data and index components are moved from the same volume, they will be moved to a single target volume. When they exist on different source volumes, CA Disk will attempt to place them on different target volumes.
2. Alternate index components are allocated separately from data and index components. When multiple target volumes are specified, the alternate index component can be placed on the same volume as the base cluster, or on a different volume, depending on the results of the allocation algorithm.
3. System Resource Manager (SRM) routines select the volume to be used for the target allocation from the target volumes.
4. An attempt will be made to avoid placing cluster components on volumes that have less free space than specified on the VATHRESH parameter. If this is not possible, allocation will choose the volume with the largest free space that can contain the cluster.
5. If a data set is selected without any volume selection criteria, CA Disk will only use the volumes specified in the TOVOL= parameter as candidate volumes. In other words, for a multi-volume cluster, the cluster can end up using one or more of its original volumes. To force CA Disk to use only NEW volumes, include the NEWVOLUMES parameter described in NEWVOLUMES.

SMS and MOVE/COPY

With the MOVE/COPY command under DSCL, Storage Administrators have the capability to move or copy data sets from one DASD device to another.

In an SMS environment, at allocation time, Automatic Class Selection routines determine which storage group a data set will ultimately reside within. CA Disk will honor this selection. Because of this, it is possible that when attempting to move or copy a data set, it will end up back on the same volume it came from.

In the example below, it is desired to move small data sets to a Storage Group reserved for small data sets. The ACS routines have been set to place data sets of sizes less than 50,000 kilobytes into a Storage Group called SMALL.

```
SCAN REALVOLS
  SELECT SGNAME=LARGE, CRI=(KBYTES,LT,50000)
  MOVE TOVOL=/
```

The previous accomplishes the following:

1. Selects those volumes contained within the Storage Group called LARGE.
2. Selects data sets that are smaller than 50,000 kilobytes.
3. Moves them according to the ACS routines.

Note: The previous is an example of how to make sure that only large data sets remain in the LARGE pool or how to initially get large and small data sets into their respective pools when the pools are first formed.

Guaranteed Space Attribute

When MOVE/COPY is processing VSAM data sets, CA Disk honors the Storage Class attribute Guaranteed Space under the following conditions:

1. The TOVOL= DSCL parameter is used to determine the volume(s) to which the data set is to be moved or copied. However, the volume(s) must be accepted by SMS's ACS routines.
2. Unlike MOVE/COPY in other situations, all volumes specified on the TOVOL= parameter are used to allocate the new data set. For example: If you specify three volumes as TOVOL=(VOL001,VOL002, VOL003), the resulting data set will be allocated on three volumes. If you specify TOVOL=VOL/, the resulting data set will reside on as many volumes as matches the VOL/ mask. In other words, to MOVE or COPY, it does not matter how many volumes the data set resides on.
3. The NEWVOLUMES parameter should always be coded in order to obtain correct results. If NEWVOLUMES is omitted, CA Disk can attempt to include some of the current source volumes in the list of volumes for the new data set. This is unlikely to be desirable for these allocations.
4. The VATHRESH parameter is ignored.
5. You must use an explicit FIND statement to select these data sets.

Sample MOVE/COPY Commands

The following JCL shows the use of the MOVE command and the flexibility that the DSCL selection criteria provides:

```
SCAN REALVOLS
*****
* MOVE ALL SEQUENTIAL data setS BEGINNING WITH LABS.KSV WITH GREATER THAN *
* 10 TRACKS FROM VOLUME LABS50 TO VOLUME LABS80                               *
*****
      SELECT DSN=LABS.KSV./, CRITERIA=(TRKS,GT,10,AND,DSORG,EQ,PS),VOL=LABS50
      MOVE TOVOL=(LABS80)
*****
* MOVE MODEL DSCBS ONLY                                                         *
*****
      SELECT CRITERIA=(MODEL DSC, EQ, YES),VOL=SYS0/
      MOVE TOVOL=(SYSX01)
```

The following JCL shows the use of the COPY command:

```
FIND DSN=LABS.KSV.JCL
*****
* COPY LABS.KSV.JCL TO VOLSER DMSK01, RENAMING IT TO LABS.KSV.JCL.OLD          *
*****
      COPY TOVOL=(DMSK01),NEWNAME=LABS.KSV.JCL.OLD
```

Sample Move/Copy Report

The following is a Sample MOVE/COPY Report:

SDSF OUTPUT DISPLAY DISK0490 JOB10001 DSID 121 LINE 1 COLUMNS 02- 133														
COMMAND INPUT ==> SCROLL ==> CSR														
1997.297 OCT 24, 1997 COMMAND STATEMENTS PAGE 1														
FRIDAY 8.34 PM CA: DISK 9.0.2														
-----2-----3-----4-----5-----6-----7-----8-----9-----0-----1-----2-----3-----														
FIND DSN=QDMSTEST.CA.DISK0490.FILES CMD00001														
COPY TOVOL=DMS903,NEWNAME=QDMSTEST.CA.DISK0490.FILES.NEW CMD00002														
FIND DSN=QDMSTEST.CA.DISK0490.PS CMD00003														
COPY TOVOL=DMS903,NEWNAME=QDMSTEST.CA.DISK0490.PS.NEW CMD00004														
FIND DSN=QDMSTEST.CA.DISK0490.PDS CMD00005														
COPY TOVOL=DMS903,NEWNAME=QDMSTEST.CA.DISK0490.PDS.NEW CMD00006														
1997.297 OCT 24, 1997 UOL VOLUME CONFIGURATOR PAGE 1														
FRIDAY 8.34 PM CA: DISK 9.0.2														

DATA SET NAME / NEWNAME SOURCE DEVICE TARGET DEVICE DS SRC TRG OP KEY ACT SOURCE SOURCE TARGET														
VOLUME TYPE VOLUME TYPE ORG BLKSZ BLKSZ LRECL CD LEN S R KBYTES TRACKS TRACKS														

QDMSTEST.CA.DISK0490.FILES DMS902 3390 DMS903 3390 DA 27990 27990 0 00 0 C 2,549 45 45														
QDMSTEST.CA.DISK0490.FILES.NEW DMS902 3390 DMS903 3390 DA 27990 27990 0 00 0 C 2,549 45 45														
QDMSTEST.CA.DISK0490.PS DMS902 3390 DMS903 3390 PS 9040 9040 00 00 0 C 849 15 15														
QDMSTEST.CA.DISK0490.PS.NEW DMS902 3390 DMS903 3390 PS 9040 9040 00 00 0 C 849 15 15														
QDMSTEST.CA.DISK0490.PDS DMS902 3390 DMS903 3390 PD 9040 9040 00 00 0 C 849 15 15														
QDMSTEST.CA.DISK0490.PDS.NEW DMS902 3390 DMS903 3390 PD 9040 9040 00 00 0 C 849 15 15														

TOTAL SOURCE TARGET														
DATASETS TRACKS KILOBYTES TRACKS KILOBYTES														

MIGRATED..... 3 75 4,247 75 4,247														
SCRATCHED..... 0 0 0 0 0														

Chapter 13: Idle Space Release

This section contains the following topics:

[About Idle Space Release](#) (see page 497)

[RELEASE Command](#) (see page 501)

[SMS and RELEASE](#) (see page 504)

About Idle Space Release

This function releases unused space from partitioned, physical sequential, and VSAM data sets. You use the DSCL selection criteria select data sets for release using DSCL selection criteria. The DSCL action command RELEASE initiates the release process. The amount of unused space released can be controlled by percent parameters so that a portion of the unused space is retained.

Command: RELEASE

JCL Proc Executed: DMS

Graceful Shutdown

Shut down the Idle Space Release jobs using the sysparm DSCLRPLY. A graceful shutdown allows the jobs to complete all work in progress before they terminate. The jobs also produce the normal reports for error conditions that occur before they terminate.

With DSCLRPLY active, a message is issued to the operator at the beginning of the job and remains on the console during the job. Replying *END* to this message stops all processing early.

Restrictions

- System temporary data set names
- SYSCTLG data sets
- Model DSCBs
- Checkpoint data sets
- CA Disk Archive data sets
- Key range VSAM data sets
- VSAM data sets with greater than 12 extents
- RRDS VSAM type data sets
- LDS
- KSDS data sets with embedded indexes
- PSE

Data sets with the high level qualifier of SYS1 are bypassed by default. To select these data sets, use sysparm RLSESYS1. Unmovable data sets and empty data sets are selected by default using the DSCL selection criteria. To prevent space from being released from unmovable data sets, use RLSEUNMVy. For more information, see RLSEUNMVy in the *Systems Guide* or see UNMOVE -- Unmovable attribute (yes/no) in [DMS PROC](#) (see page 112).

Release processing releases idle space from both empty non-VSAM and VSAM data sets. If you allocate space for data sets in one job, and then load data into them in a second job, do not release space from empty non-VSAM data sets. If you select the data bases for processing, each is reduced to the minimum allocation based on its type. That is, either a one-track, a one-cylinder, or a one-MCU data set (reduction to zero tracks that are allocated is not done).

Empty VSAM data sets have a HURBA (High Used Relative Byte Address) of zero. CA Disk uses the HURBA to calculate how much space to release. For space release, CA Disk calculates the HURBA as if equal to the number of bytes in the Control Area. Exclude from processing those data sets, such as SYS HASPACE, that appears empty or over allocated but are not. Use the data set name or volume exclusion lists to exclude these data sets.

For multivolume, non-VSAM data sets, idle space is only released if the idle space is on the last volume. and only if the physical and logical last volumes are the same. For example, if four volumes have data and volume 4 is the logical last volume, then any idle space on volume 4 is released. If only volume 1 and volume 2 have data, then volume 2 is the logical last volume. The idle space is not released, because volume 2 is not the last physical volume. FMT1/8 and DSCB operating system limitations for multi-volume data sets cause this scenario.

Percentage calculations are bypassed because they are based on data set space by volume and would be incorrect. All idle space is released.

Messages

For data sets selected for processing, CA Disk releases idle space using an EXCLUSIVE ENQ (DISP=OLD). If a data set is in use by another task, a diagnostic message is issued and the release is not attempted.

The User Exit USERSENQ is invalid for Idle Space Release Processing. If you specify this User Exit, an error message is issued and a user abend occurs.

Other messages are issued only for those cases where an attempted space release failed. Use sysparm RLSEDIAG to generate diagnostic messages with the reasons why data sets are bypassed.

System Parameters

The following sysparms are relevant to the Idle Space Release function:

PDSEFAMS	Specify release processing for PDSE data sets
RLSEDIAG	Specify printing of release bypass messages
RLSEMAXS	Specify resetting of secondary space allocation
RLSEMINs	Specify resetting of secondary space allocation
RLSESECR	Specify resetting of secondary space allocation
RLSESYS1	Specify release processing for SYS1 data sets
RLSEUNMV	Specify how to process unmovable data sets
VSVERIFY	Specify verifying of VSAM data sets if needed

VSAM Considerations

Unused space is released from ESDS and KSDS VSAM type data sets only. Selection for release processing is by cluster name or alternate index, not at the VSAM component level. Only VSAM data sets cataloged in ICF catalogs are selected. The Release function does not process VSAM data sets with more than 12 extents.

Space is released from the data and index components for KSDS data sets and from the data component for ESDS data sets. A new HARBA (High Allocated Relative Byte Address) value is calculated for the VSAM data set. The calculation uses the percentage that you specify, the HURBA (High Used Relative Byte Address) value, and the Control Area boundary. The new HARBA must be on a Control Area boundary. After the new HARBA value is calculated, the amount of physical space to release from the last extent is determined. The unused space is released only from the last extent used. The VTOC and the catalog are updated to reflect these changes.

The percentage-used value that you specify is applied against the HARBA and HURBA values of the VSAM data set. The value is not applied to the physical space allocation of the data set. The amount of physical space that is released cannot correspond to the percentage specified by you. For more information, see the [Sample RELEASE Report](#) (see page 503).

The PCTSECONDARY and CONVERT DSCL RELEASE action command parameters are not valid for VSAM data sets, and are ignored. Since the PCTSECONDARY parameter value is ignored, you cannot display it on the Idle Space Release report for VSAM data sets.

CA Disk can convert the VSAM indexes that are allocated by cylinder to track allocation. Since the CONVERT parameter is ignored for VSAM, the conversion occurs without regard to the value specified for the parameter. This can be done only if the following conditions are true:

1. Cylinder allocates the Index originally.
2. The Index Control Area boundary is one track.
3. The Index is released down to less than one cylinder.

PSE Considerations

Extended sequential data sets are multi-volume sequential data sets that are managed by DFSMS. This combination resulted in some special considerations that each user should be aware of. A complete list of these considerations is located in the section Special Considerations While Processing PSEs in the chapter "Basic System –."

Note: Support for Idle Space Release is limited to single stripe data sets only.

See **DMS PROC**, Basic System

RELEASE Command

Release processing is done through the DSCL RELEASE action command. To select data sets to be processed, use the DSCL selection criteria.

The RELEASE command syntax is:

```
RELEASE PCTUSED=,PCTSECONDARY=,CONVERT
```

PCTUSED=

Specify the percent of used space that is desired in the data sets after processing. If no value is specified, all space in the data set is used. (Default is 100 percent.)

PCTSECONDARY=

(Optional) Include this parameter to reset the amount of secondary space that is allocated for non-VSAM data sets selected for release of idle space. Specify a percentage (0-100) of the new primary allocation (after release of idle space) to be used to compute a new secondary space allocation.

CONVERT

This parameter causes non-VSAM data sets using less than one cylinder and allocated by cylinders, to have their allocations changed to tracks. (The secondary space allocation quantity is also appropriately converted; for example, one 3350 cylinder converts to 30 tracks.)

RELEASE Command Examples

The following JCL shows the use of the RELEASE command, and the flexibility that the DSCL selection criteria provides:

```
//JOBNAME JOB (acct,info),etc.

// EXEC   DMS
SCAN REALVOLS
*****
* SELECT SEQUENTIAL, NON-EMPTY data setS, ALLOCATED IN      *
* TRACKS WITH SECONDARY SPACE ALLOCATION, THAT HAVE GREATER *
* THAN 25 PERCENT USED AND LESS THAN 50 PERCENT USED.      *
*                                                           *
* RELEASE SPACE SO THAT SELECTED data setS HAVE 75 PERCENT *
* USED SPACE, AND RESET THE SECONDARY SPACE.                *
*****
      SELECT CRITERIA=(PCTUSED,NE,0,AND,DSORG,EQ,PS,AND,
        PCTUSED,GT,25,AND,PCTUSED,LT,50,AND,
        ALLOTOKS,EQ,YES,AND,ALLOSEC,GE,
        VOLUMES=WORK/,DSNAMES=USER./
      RELEASE PCTUSED=75,PCTSECONDARY=25
*****
* SELECT PARTITIONED AND VSAM data setS, THAT HAVE GREATER *
* THAN 50 PERCENT USED AND LESS THAN 90 PERCENT USED.      *
*                                                           *
* RELEASE SPACE SO THAT SELECTED data setS HAVE 95 PERCENT *
* USED SPACE,                                              *
*                                                           *
* RESET THE SECONDARY SPACE FOR THE SELECTED PARTITIONED *
* data setS AND CONVERT CYLINDERS TO TRACKS.              *
*****
      SELECT CRITERIA=(DSORG,EQ,P0,OR,DSORG,EQ,VSAM,AND,
        PCTUSED,GT,50,AND,PCTUSED,LT,90,AND,
        ALLOCYLS,EQ,YES),
        VOLUMES=MVS/,DSNAMES=TEST./
      RELEASE PCTUSED=95,PCTSECONDARY=50,CONVERT
```

RELEASE Command Sample Report

The following is a Sample RELEASE Report:

2015.005 JAN 05, 2015 I D L E S P A C E R E L E A S
E PAGE 1
MONDAY 6.12 PM CA
DISK r12.5

DEV DS ALLO ----OLD
TRACKS-----NEW TRACKS----- --RELEASED----- -COMMAND
DATA SET NAME VOLUME TYPE ORG TYP ALLOC IDLE %US
ALLOC IDLE %US BYTES TRKS %US %2ND

***** THIS IS A SIMULATION RUN

YEOMI01.J1804553.BACKUP1										D86810	3390	PS	CYL	50085	50070
1	315	300	5	2.72G	49770	100									
ASMDSK.TEST.NOV1714.CYM2										D86810	3390	PS	CYL	80010	77001
4	3150	141	96	4.21G	76860	100									
HAWWI01.GLOBAL.TESTFILE										D86810	3390	PS	CYL	75285	74568
1	720	3	100	4.08G	74565	100									
BUTOS01.TESTFILE										D86810	3390	PS	TRK	65535	62526
5	3009	0	100	3.42G	62526	100									
LIGGA01.WEEKLY.TEST.SEQ.DOWNLOAD										D86810	3390	PS	TRK	65537	62528
5	3009	0	100	3.42G	62528	100									
YEOMI01.J1804553.KSDS.DATA										D86810	3390	AM	CYL	80010	66409
17	13665	64	99	3.63G	66345	100									
YEOMI01.J1804553.KSDS.INDEX										D86810	3390	AM	CYL	630	593
6	315	278	11	17.66M	315	100									
ASMDSK.TEST.FORECAST.KSDS.DATA										D86810	3390	AM	CYL	50085	50085
0	15	15	1	2.74G	50070	100									
ASMDSK.TEST.FORECAST.KSDS.INDEX										D86810	3390	AM	CYL	630	630
0	315	315	1	17.66M	315	100									
ASMISM.MONTHLY.BACKUP										D86810	3390	PS	CYL	50085	50070
1	15	0	100	2.74G	50070	100									

* AN ASTERISK BEFORE DATA SET NAME INDICATES RELEASE FAILED WHILE PROCESSING THE DATA SET

* AN ASTERISK BEFORE ALLOCATION TYPE INDICATES IT HAS BEEN CONVERTED FROM CYLINDER TO TRACK ALIGNMENT

2015.005 JAN 05, 2015 I D L E S P A C E R E L E A S
E PAGE 2
MONDAY 6.12 PM CA
DISK r12.5

T O T A L S
-----RELEASED-----
-----DATASETS-----
VOLUME TRACKS BYTES TYPE TRACKS BYTES GRAND: TRACKS BYTES PO
PS VSAM OTHER

```
-----  
-----  
  
D86810  481.80K  27.01G  3390  481.80K  27.01G      481.80K  27.01G      0  
6        4        0
```

SMS and RELEASE

CA Disk uses the Management Class and Storage Class attributes to determine the data set eligibility for Idle Space Release. External Idle Space Release processing remains unchanged. Internally it determines when it is running in an SMS environment and uses the Management Class and Storage Group information to make its decisions.

The SMS fields that CA Disk supports are as follows:

- Partial Release (MC) This optional field can be defined Y, YI, N, C, and CI for VSAM and non-VSAM data sets. The default is N. When this field is defined Y or YI, the data set is eligible for idle space release.
- When this field is defined C or CI with a non-zero secondary space allocation, it is eligible for idle space release. When this field is defined N, it is not eligible.
- Auto Migrate (SG). You can define this required field as YES, NO, INTERVAL, or PRIMARY. When this field is defined as YES, INTERVAL, or PRIMARY, the data set is eligible for idle space release. When this field is defined NO, it is not eligible.

Chapter 14: PDS Compression

This section contains the following topics:

[About Compression](#) (see page 505)

[General Information](#) (see page 506)

[JCL — PDS Compression](#) (see page 507)

About Compression

PDS Compression is provided to:

- Recover the dead space within a partitioned data set.
- REDUCE or INCREASE the size of the data set.
- REDUCE or INCREASE the size of the directory of the data set.

Change the allocation type of a partitioned data set.

The reduction or increase amounts are based on the contents of the data set. PDS compression can be accomplished implicitly by the designation of volumes to be scanned for eligible datasets, or explicitly by the specification of a data set name.

Commands available: SCAN, COMPRES, and RELOADS.

JCL Proc executed: COMPRES.

General Information

This facility is provided not only to recover the dead space within a PDS, but also to enlarge or shrink the size of both the data set and its directory, based upon their content. The compression is accomplished by copying each data set to a work file, calculating the reallocation requirements, then copying it back. A message is issued each time the copy-back phase of a compress is begun.

Since a single work file is used for compressing all of the datasets, the size of the work file must be large enough to accommodate the compressed form of the largest data set to be encountered.

The work file can be on any DASD device type (it can span multiple volumes) and given any block size, since CA Disk restructures the data to satisfy the target work file.

Note: If you direct this work file to a specific volume serial (by overriding the COMPWRK DD statement with a VOL=SER parameter), COMPRESS will fail with a U200 abend. To avoid this abend, you must also include a DISP= parameter on your override statement.

In the event that a PDS that was copied to the work file cannot be reloaded due to an unsuccessful allocation or some other reason, **message 3179, ACTIVE WORK FILE, CONTAINS dsname**, will be issued and the job will terminate with a USER 200 abend. The work file will be kept to allow the data set to be recovered via the RELOAD command. The PDS can be reloaded to the original volume or to a new volume.

PDS compression can be accomplished implicitly by the designation of volumes to be scanned for eligible datasets, or explicitly by the specification of a data set name.

The report produced by PDS Compress contains two lines for each data set processed. The first line presents the status of the data set prior to compression. The second line shows its status after compression.

Restrictions:

An informational message will be issued to indicate that the PDSE data sets will not be compressed.

CA Disk will bypass APFLST or LNKLST data sets unless explicitly selected by data set name.

Condition Codes

The following is a list of condition codes that can occur during the execution of PDS Compress:

0

Successful completion

4

No datasets selected or some bypassed in reallocation

8

Data set(s) resource cannot be acquired (for example security or enqueues)

12

Unrecoverable error encountered in data set(s)

16

Command error

JCL — PDS Compression

Partitioned datasets are compressed by executing the following JCL:

```
//jobcard
//PDSCOMP EXEC COMPRES
//SYSIN DD *
    <enter PDS SCAN/COMPRESS commands here>
/*
```

SCAN Command

The SCAN command can be used to select one or several datasets for compression from one or more volumes.

During SCAN processing, datasets that are eligible for compression, but which are in use by other tasks, are noted and bypassed until the end of processing for that volume. At that time, the compress procedure re-examines the bypassed datasets to see if they are still in use by other tasks. If not, they are processed. For those datasets still in use, CA Disk issues an informational message and no processing is performed.

Note: SMS managed datasets that are not cataloged will only be processed when the USE parameter is specified. Otherwise, the data set will be bypassed.

The parameters described below specify the selection criteria as well as the reallocation values.

The SCAN command syntax is:

```
SCANSIMULATE, VOLUMES=, EXTABLE=, DSNAME=, THRESHOLD=, EMPTY, USE, SHRINK, CONVERT,  
    ALLOCATE=, RESERVE, PASSWORD
```

SIMULATE

This parameter causes the output report to be produced as if compression had been performed. However no actual compression occurs.

Note: SIMULATE has the following implications:

- Since data integrity is not an issue during SIMULATE processing, no ENQ or RESERVES are performed. This eliminates system contention for datasets and enhances performance.
- Sysparm ENQDSSIM can be specified with a value of Y to cause ENQUEUEs to be issued. For further details, see the sysparm description for ENQDSSIMn in the *Systems Guide*.
- The directory is always read, whether you specify SIMULATE or not.
- CA Allocate, SMS, or other allocation products can override the actual allocation. So the selected data sets will not be compressed as indicated.
- Smaller empty PDSs can be selected for compression, but not actually changed because of minimum directory requirements.

To further enhance performance during SIMULATE processing, only the count fields of the member data blocks are read, which provides sufficient information to calculate reallocation values.

VOLUMES=

A list of one or more volumes or volume prefixes must be specified to designate the volumes to be scanned for PDS datasets to be compressed. Default processing looks for these volumes only as online real disk packs. (A maximum of 50 entries is supported.)

EXTABLE=

Specify for this parameter the name of the parmlib member that contains entries consisting of either complete data set names or patterns that are to be excluded from compression. This parameter can be nullified during SIMULATE runs by specifying sysparm PDSEXSIM with a value of N. This allows even the excluded datasets to be reported on during SIMULATE runs, although live runs will exclude them.

DSNAMES=

Candidates for compression can be restricted to those datasets having names or pattern names that are specified in the list value of this optional parameter. (A maximum of 10 entries is supported.)

THRESHOLD=

This parameter provides a method of limiting those datasets selected for compression. It allows you to define five conditions, one of which must be met before a PDS will be eligible for processing. Datasets can be selected based on threshold ceilings, so you can recover dead space and/or increase allocation amounts to reduce *out of space* problems. Datasets can also be selected based on threshold floors, so you can recover overallocated space. (This parameter applies only to PDSs containing members. See EMPTY For a similar test for PDSs containing no members.)

The conditions have the following format and meaning: (UC,XC,DC,UF,DF)

- **UC *Used Ceiling*** — The maximum percentage of allocated space (excluding the space allocated to the directory blocks) that is used. This value can be thought of as a *ceiling* to the threshold range. Any data set using more than this percentage of allocated space will be eligible for processing. A value of 0 will nullify this threshold test.
- **XC *Extent Ceiling*** — The maximum number of extents used by each data set. This can be thought of as a threshold *ceiling*. Any data set going into more than this number of extents will be eligible for processing. A value of 0 will nullify this threshold test.
- **DC *Directory Ceiling*** — The maximum percentage of directory space that is used. This can be thought of as a *ceiling* to the threshold range. Any data set that has a directory using more than this percentage of allocated directory space will be eligible for processing. A value of 0 will nullify this threshold test.
- **UF *Used Floor*** — The minimum percentage of allocated space (excluding the space allocated to the directory blocks) that is used. This value can be thought of as a *floor* to the threshold range. Any data set using less than this percentage of allocated space will be eligible for processing. A value of 0 will nullify this threshold test.
- **DF - *Directory Floor*** — The minimum percentage of directory space that is used. This value can be thought of as a *floor* to the threshold range. Any data set that has a directory using less than this percentage of allocated directory space will be eligible for processing. A value of 0 will nullify this threshold test.

THRESHOLD= (continued)

For conditions UC and DC, specifying a value of 100 or more will cause all datasets to fail the threshold ceiling test. For condition XC, specifying a value of 16 or more will cause all datasets to fail the threshold ceiling test.

Since these thresholds provide the upper and lower limits for datasets that are considered ineligible for processing, it is not valid to specify for conditions UF and DF a value higher than the value specified in conditions UC and DC, respectively.

See also the USE, SHRINK and CONVERT parameters for how they are used in conjunction with THRESHOLD=.

EMPTY

This parameter provides an additional method for selecting datasets for compression. By specifying EMPTY, you extend PDS Compress processing to include empty datasets (that is, PDSs containing no members).

- If this parameter is not present, empty datasets will not be selected for processing. However, in explicit mode (that is, the COMPRESS command) the SPACE= parameter will force the compress even if the data set is empty.
- If the EMPTY parameter is specified along with THRESHOLD=(0,16), only empty PDSs will be selected.
- Selected empty datasets are indicated by an EF in the THRESHOLD EXCEEDED field of the PDS Compression report.
- When the EMPTY parameter is specified, processing is affected by the USE parameter as follows:
 - If the USE parameter is present, any dead space once used for members that have been deleted is eliminated.
 - If the USE parameter is not present, all primary space (excluding the space required for the directory blocks) is freed. The space thus freed is divided by 15 (the number of possible secondary extents) and added to the current secondary allocation amount. This in effect compresses the data set down to its smallest amount while still allowing room for future expansion. See also the CONVERT and SHRINK parameters.
- When EMPTY is specified, the following conditions for compression can be displayed in the *THRESHOLD EXCEEDED* column of the compression report.
- *EC Empty Ceiling* -- This condition is triggered when the DS1LSTAR field of the Format-1 DSCB does not point to the first record after the directory. This can occur when all members have been deleted, and the data set has not yet been compressed.
- *EF Empty Floor* -- This condition is set for empty PDS's when EMPTY was specified on the command line and the USE parameter was not specified. The action is to free all space except that needed for the directory.
- Smaller, empty PDSs cannot be further compressed because of minimum directory requirements, CA Allocate, SMS, or other products that override allocation.

USE

The presence of this parameter forces the use of the existing data set extent(s). Any allocation changes, if specified, are confined to directory adjustment; that is, the PP and SS sub parameters of the ALLOCATE= parameter are ignored and only the DD readjustment is done.

- For empty datasets (if the EMPTY parameter is also specified), the presence of this parameter will cause the ALLOCATE= parameter to be ignored.
- If the USE parameter is present and sysparm PDSENQSH is specified with a value of Y, only a SHARED ENQ is maintained for the data set during compression. In addition, ISPF and linkage editor enqueues are issued to preserve the data set's integrity.

SHRINK

Specify this parameter to allow the data set allocation to be reduced. Default processing (this parameter absent) will not allow a data set to be reduced in size. The ALLOCATE= parameter must be present for SHRINK to be active.

Empty datasets (if the EMPTY parameter is also specified) will be reduced in size regardless of whether or not the SHRINK parameter is specified. Specifying the SHRINK parameter, however, provides an even greater space savings, because it reduces the directory allocation down to one track (or cylinder—see CONVERT parameter), thus permitting the primary space allocation to also be reduced to a single track (or cylinder).

CONVERT

This parameter can be used in conjunction with the SHRINK parameter for datasets currently allocated in cylinders. If it is specified on the same command as the USE parameter, CONVERT will be ignored. When a new space allocation is calculated and is less than one cylinder, this parameter converts the space allocation amount and type from cylinders to tracks (or blocks). This includes datasets allocated with the ROUND request in which the blocks are rounded up to space equal to an integral number of cylinders.

- For empty datasets (if the EMPTY parameter is also specified), the CONVERT parameter will have the following effect:
- If the SHRINK parameter is not present, the primary allocation will be reduced to the minimum number of tracks required to hold the directory blocks.
- If the SHRINK parameter is also specified, the primary allocation (including the directory blocks) will be reduced to one track.

ALLOCATE=

This optional parameter specifies three values used to reallocate the primary space, secondary space and directory space. The format and meanings are as follows:

(PP,SS,DD)

- PP—The percentage of primary space (excluding the space allocated to the directory blocks) to be used after reallocation and compression. A value of 0 can be used to force the new allocation equal to the original.
- SS—The new secondary allocation quantity (excluding the space allocated to the directory blocks) expressed as a percentage of the primary. A value of 0, which is the default, can be used to force the new allocation equal to the original.
- DD—The percentage of directory space to be used after reallocation and compression. A value of 0, which is the default, can be used to force the new allocation equal to the original.

See the USE, SHRINK and CONVERT parameters for how they are used in conjunction with ALLOCATE=. Also see the SPACE= parameter (described under the explicit COMPRESS command in the following section.)

RESERVE

This parameter is provided for use in a multiple- CPU environment that does not use a global enqueue facility, to protect against other CPUs making concurrent updates. It causes a RESERVE to be issued against the volume in addition to the shared or exclusive ENQ already issued on the data set name. (See USE.) If this parameter is omitted, only the ENQ is done. The RESERVE and/or ENQ are done just prior to the compress of each data set and DEQ is issued immediately upon completion.

PASSWORD

This parameter is active only if sysparm PASSWORD has been specified with a value of Y. Specify this parameter to allow CA Disk to compress password- protected datasets without prompting the operator for the password. If you do not specify this parameter, CA Disk will bypass these password-protected datasets.

COMPRESS Command

The COMPRESS command must be used to explicitly compress individual datasets.

The COMPRESS command syntax is:

```
COMPRESS SIMULATE,DSNAME=,VOLUME=,THRESHOLD=,EMPTY,USE,SHRINK,CONVERT,  
        ALLOCATE=,SPACE=,RESERVE,PASSWORD
```

DSNAME=

This parameter designates the name of the PDS to be compressed.

VOLUME=

This parameter must be supplied when the PDS is not cataloged. If the PDS is SMS managed, the USE parameter must also be specified. Otherwise, the data set will be bypassed.

SPACE=

Use this parameter when you want to provide absolute values (rather than percentages) for the new space allocation of a PDS.

- When processing in implicit (SCAN) mode, using percentage values for calculating the reallocation amounts permits efficient processing of numerous datasets. However, when specifying an explicitly named PDS (COMPRESS), the desired reallocation amounts are often known, and can be supplied directly using this parameter. SPACE= can be used only with explicit processing.
- To request actual space reallocation quantities, specify SPACE= in conjunction with the ALLOCATE= parameter. When thus combined, the SPACE= parameter supplies the allocation type, while the ALLOCATE= parameter supplies the allocation quantities. For example, specifying means to reallocate the PDS with 30 primary tracks and 10 secondary tracks, and a block directory

```
SPACE=TRK,ALLOCATE=(30,10,5)
```

- If SPACE= is specified without the ALLOCATE= parameter, the data set will be reallocated using the new allocation type. For example, specifying will convert the original allocation to cylinders with the CONTIG attribute.

```
SPACE=CYLCTG
```

- If SPACE= is specified, the USE, SHRINK, CONVERT, EMPTY, and THRESHOLD parameters are ignored.

Note: The use of the SPACE= parameter will force the compression of empty datasets, whether the EMPTY parameter is specified or not.

Specify this parameter with one of the following literal values:

Note: For any type of block allocation, the block size is obtained from the data set's format-1 DSCB. If the block size is 0 (or greater than the maximum block size for the device), the block size is assumed to be the maximum for the new device; therefore the new allocation will be the same number of tracks as the number of blocks specified.

Literal Value	Redefines Allocation to
CYL	Cylinder request
CYLCTG	Cylinder with CONTIG request
TRK	Track request
TRKCTG	Track with CONTIG request
BLK	Average block length request
BLKCTG	Average block length with CONTIG request
RND	Average block length with ROUND request
RNDCTG	Average block length with CONTIG and ROUND request
Literal Value	Redefines Allocation to

Additional Parameters Same as Implicit SCAN Command

All of the following parameters have the same meanings and use as documented for the implicit SCAN command:

SIMULATE

EMPTY

USE

SHRINK

CONVERT

RESERVE

PASSWORD

Additional Parameters Different From the Implicit SCAN Command

The following parameters can have meanings that are different from the implicit SCAN command:

All parameter values set to THRESHOLD= are ignored in explicit processing.

The data set will be reallocated using the values specified for this parameter. (that is, ALLOCATE=(10,0,0) will reallocate the data set with 0 secondary allocation and all unused directory blocks freed.

RELOAD Command

The RELOAD command is used to restart PDS Compress processing for a single specific data set after a system failure occurred during a PDS Compress execution. For example, if a system failure had occurred during a PDS Compress execution and the data set being compressed at the time of the failure had been scratched, reallocated, and was in the process of being reloaded, this procedure would restart the reload process to recover the PDS.

If the system failure occurred during the processing of unloading the PDS to the work data set, there is no need to use the reload command because the original data set is intact. To determine where the compress was interrupted, see the CA Disk message listing. A *copy-back started* message is printed before each data set compressed message. If a *copy-back started* message is not followed by a successful compression message, the RELOAD has not completed successfully, and the RELOAD function must be run to recover the data set.

Note: A target data set for the RELOAD must be pre-allocated with the proper DCB attributes. The DCB attributes are printed in the *reload starting* message as a source of reference.

JCL — RELOAD

Use the following JCL to execute the RELOAD command. Remember, since only one PDS was left in the work file (in the process of being compressed), only one RELOAD command is supported. The syntax of the RELOAD command is:

```
//RELOAD   EXEC  COMPRES  
//SYSIN    DD   *  
           RELOAD DSNAME=, VOLUME=, PASSWORD
```

DSNAME=

(Optional) If this parameter is omitted, CA Disk retrieves the DSNAME from the COMPWORK DD in the COMPRES PROC. Otherwise, this is the name of the PDS that has been preallocated for recovery of the unloaded data set.

VOLUME=

(Optional) If the data set being reloaded is not cataloged, or you do not want CA Disk to use the catalog to locate it, specify the DASD volume serial number on which the data set resides.

PASSWORD

(Optional) Specify this parameter to allow CA Disk to reload password-protected datasets without prompting the operator for the password. If you do not specify this parameter and the data set is password-protected, the reload will fail once again.

Sample PDS Compression Report

The PDS compression report contains two lines for each data set processed. The first line presents the status of the data set prior to compression. The second line shows its status after compression. The following is a sample PDS Compress Report:

File Edit Transfer Options Connection Macro Window Help



SDSF OUTPUT DISPLAY ISPDLM1A JOB28428 DSID 109 LINE 32 COLUMNS 02- 133
COMMAND INPUT ==> SCROLL ==> CSR
1997.301 OCT 28, 1997 PARTITIONED DATA SET COMPRESSION PAGE
TUESDAY 9.43 AM CA:DISK 9.0.2

PRI	EXT	DIR	THRESHOLD EXCEEDED	VOLUME	DEVICE TYPE	-SPACE REQUEST-		EXT	-----TRACKS-----			-DIRECTORY BLOCKS-			--DIRECTORY ENTRIES--		
						TYPE	PRIME SEC		ALLOC	USED	IDLE	ALLOC	USED	IDLE	ALLOC	USED	IDLE
*****	THIS IS A SIMULATION RUN																
CMUS108.BASE.ASM			SSL802	3300	TRK	56	12	1	56	56		15	14	1	89	83	6 42
		DC	SSL802	3300	TRK	56	12	1	56	56		56	14	42	335	83	252 42
CMUS208.BASE.ASM			SSL802	3300	TRK	63	15	1	63	61	2	28	17	3	119	99	20 42
		DC	SSL802	3300	TRK	63	15	1	63	62	1	67	17	50	401	99	302 42
C279302.PLIB			SSL802	3300	BLK	195	158	1	15	12	3	10	10		61	61	41
		DC	SSL802	3300	BLK	195	158	1	15	12	3	40	10	30	241	61	180 41
C279305.CLIST			SSL802	3300	TRK	7	1	1	7	3	4	4	4		23	21	2 41
		DC	SSL802	3300	TRK	7	1	1	7	3	4	15	4	11	89	21	68 41
ISPDK1.LOGON.TABLES			SSL802	3300	BLK	44	15	1	22	18	4	20	10	2	416	360	56 12
		DC	SSL802	3300	BLK	44	15	1	22	19	3	70	18	52	1466	360	1106 12
ISPJHT1.VAN30.JCL			SSL802	3300	TRK	77	5	5	97	95	2	53	41	12	330	255	75 39
		DC	SSL802	3300	TRK	97	5	1	97	50	47	163	41	122	990	255	735 39
ISPKAL1.DMS.CNTL			SSL802	3300	TRK	83	15	1	83	82	1	75	68	7	454	407	47 41
		DC	SSL802	3300	TRK	83	15	1	83	72	11	269	68	201	1618	407	1211 41

Aa P TCP/IP R 2 C 21 9:44 10/28/97

SMS and PDS COMPRESS

CA Disk uses the Storage Group attributes to see if the data set is eligible for compression.

The SMS field that CA Disk supports is as follows:

- Auto Migrate (SG) This required field can be defined as YES, NO, INTERVAL or PRIMARY. When this field is defined as YES, INTERVAL or PRIMARY, the data set is eligible for compression. When this field is defined NO, it is not eligible.

Chapter 15: Sequential Migration to Tape

This section contains the following topics:

[About Migration](#) (see page 519)

[General Information](#) (see page 519)

[JCL - Sequential Migrate](#) (see page 525)

[Sequential Migrate Commands](#) (see page 525)

[Migration Concerns](#) (see page 530)

[MIGRECAT Function Description](#) (see page 530)

About Migration

This CA Disk facility migrates sequential data sets from disk devices to tape. The data sets are written and recataloged as standard labeled tape files, multiple files per tape. Any job that previously accessed its disk data set via the catalog now calls for a tape mount and continues running as before.

Commands available: MODESET, MIGRATE, SCANRE, and CATALOG

JCL Proc executed: MIGRATE

General Information

Sequential data sets can be moved from online storage to tape and recataloged by either explicit or implicit reference. Each DASD data set migrated is written as a standard labeled tape data set and cataloged, enabling any job that formerly referenced the data set through the catalog to continue to do so without change. Exception cases are those sequential data sets that are concatenated to other DASD data sets when read. This will cause an unlike device type error at allocation time. Similarly, if all generations of a GDG are to be read in by referencing the index name only, all of the generations must reside on either DASD or tape, but not mixed.

An attempt is made to catalog all data sets written to tape. Uncataloged non-GDG data sets are cataloged, and previously cataloged data sets (correctly cataloged) are recataloged. 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 guide.)

Installation Options

Several installation sysparms are available to alter the data set selection and processing techniques.

MIGYRCHG

Can be used to cause a new tape to be mounted when the year portion of the expiration date changes. Default processing sorts all data sets to be migrated into descending sequence by expiration date, but does not cause a volume switch when the year changes.

MIGNOEXP

Can be used to permit expired data sets to be migrated. Default processing is for unexpired data sets only.

MIGNXPDT

Can be used to suppress the selection of data sets that have all zeros for an expiration date. Default processing treats them as unexpired and as candidates to be migrated.

MIGCATLG

Can be used to change the catalog action for uncataloged non-GDG data sets. The default value is Y, which means to catalog all non-GDG data sets migrated to tape that were not cataloged before.

MIGCTGDG

Can be used to change the catalog action for uncataloged GDG data sets. The default value is N, which means no change is made in the catalog when an uncataloged GDG is migrated to tape.

MIGPSGOP

Can be used to create a duplicate backup copy concurrent with the primary Sequential Migration tape(s). An entry in the MIGRECAT subfile is also created if this SYSPARM is set to 'Y'. The default is 'N' which does not produce the copy.

MIGDSMAX

Can be used to set an absolute limit to the number of data sets migrated to each output tape. The default is set at 9999.

MIGPSDEN

Can be used to specify tape density for simulate mode processing. It defaults to 6250. In live mode, the actual density used is properly obtained.

MIGBYPAS

Can be used to cause messages to be printed that state why a data set was bypassed for processing.

RACFSEQM

Can be used to also select RACF-protected data sets for migration. By default, CA Disk will bypass RACF-protected data sets from migration.

Expiration Dates Assigned to Data Sets on Tape

For every data set on tape, the header label contains the expiration date field just as the format-1 DSCB does for every data set on disk. As each data set is migrated from disk to tape, a decision is made regarding what value to supply for its expiration date on tape. Sysparms MIGEXPDT and MIGFXPDT control this decision.

The default value of N for MIGEXPDT causes each expiration date on disk (from the format-1 DSCB) to be copied forward to the tape, unless its value is all zeros. If it is zeros, CA Disk will substitute the expiration date (or retention period) that you supply in the MIGRATE dd statement. If you specify a value of Y for MIGEXPDT, the JCL-supplied date will be substituted for every data set being migrated. (If a duplicate copy tape is being created, the expiration dates for it are created in the same manner but the date from the MIGRATEC dd statement is used instead, and it can be different from the primary tape.)

For more information, see the section Year 2000 Considerations in the chapter "General Information."

You should also consider that tape management systems often regard tapes created with data set expiration dates equal to the current date as being work tapes, and therefore won't log the data sets in the tape management catalog. To prevent this, sysparm MIGFXPDT is provided. Its default value of Y causes CA Disk to add one day to the expiration date whenever it finds an expiration date equal to the current date.

Sequential Migration Versus Archival

Although sequential migration to tape and the archival function can both free up disk space, the following differences should also be considered. Migration to tape handles only PS data sets while archive handles all types. Data sets migrated to tape are allocated and read by the application program directly, while archived data sets require that they be restored first, although this can be done automatically by the auto-restore feature. Migrated data sets are cataloged to tape and then managed by conventional means. (This can present a problem in deciding how long to keep the tapes and how to clean up the catalog when the tapes do expire. Migrating only GDG data sets helps manage this problem considerably.) CA Disk manages archived data sets using the index records. Index maintenance functions will recycle tapes, as well as keep the catalog clean by uncataloging appropriate data sets as they expire from the archives.

Multivolume Tape Processing

Data sets migrated to tape are written as standard labeled tape files; there are multiple files per tape. The first data set written does CA Disk create the dummy file it is standard label file 1 (1,SL). The first data set migrated is standard label file 2 (2,SL). The file number is incremented by one for each data set on the tape. If CA Disk predicts that the next data set will not fit on the current tape, it causes a new tape to be mounted and begins again with file 1.

If a data set is too large to fit on one tape (or if CA Disk thought the next data set would fit but it doesn't because the tape is shorter than expected), CA Disk will continue writing the data set on additional tapes as needed. When processing for that data set is complete, CA Disk will ask for a new tape to be mounted for subsequent data sets and will reset the file number to 1.

CA Disk does not stack additional data sets with incremented file numbers on the end of multivolume tapes. The reason is, for the operating system to correctly locate those file numbers for subsequent jobs that need the data sets; each tape in the multivolume set must be mounted, spun to the end to find and count the file numbers, and rewind, until the correct tape and file number are found. By not allowing this, CA Disk ensures that every data set needed can be found on the first tape that is mounted.

Compatibility with Tape Management Systems

CA Disk uses standard system OPEN/CLOSE routines to create data sets on tape. CA Disk passes all data that is normally passed to the system through JCL dd statements for the output tape data sets to OPEN internally. This includes the tape expiration date. This provides compatibility to any user tape management system that gets information at tape open time. Two commonly used tape management systems that use this technique are CA 1 Tape Management and CA TLMS. CA Disk is completely compatible with the way these systems control tape use.

Creation of Sequential Migrate Tape First File

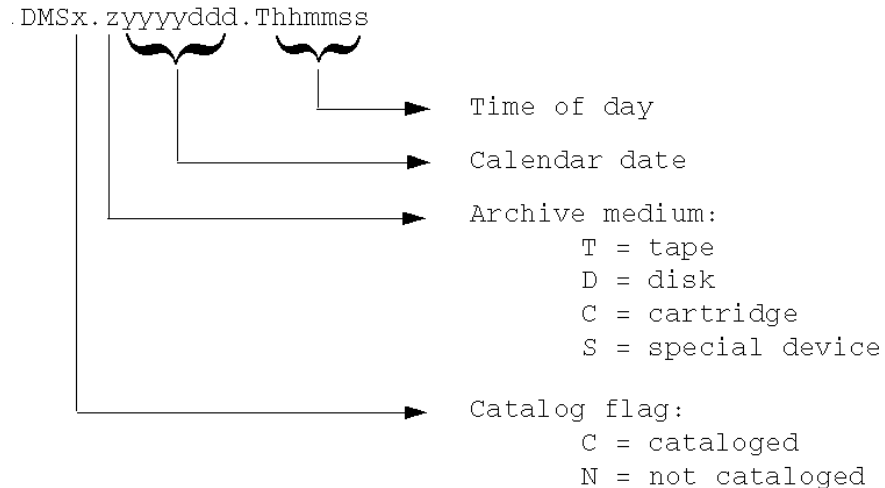
The first file of each sequential migrate tape is a data set created by CA Disk. This data set is a dummy file that contains no information, but its presence helps prevent data loss when you are using a tape library management system such as CA 1 Tape Management.

The expiration date of the first file is the highest expiration date on the tape. It is equal to the expiration date of the second file on the data set. When writing data sets to tape, CA Disk sorts the data sets in expiration date sequence, from the highest to the lowest.

Default processing leaves this first data set uncataloged. Specify the DMYCATLG parameter of the MODESET command to catalog this data set.

The data set name CA Disk assigns to this first file is a combination of the data set name from the MIGRATE JCL and a generated date and time stamp. This CA Disk-generated data set name for the first file helps provide consistency between the sequential migrate tapes, and can make them easier to manage. The first data set name on a tape is part of the criteria CA 1 Tape Management uses in determining what action to take on a tape, such as marking the tape as a scratch tape.

The base name of the data set is taken from the data set name in the MIGRATE dd statement for the primary tape, and the MIGRATEEC dd statement for the copy tape. CA Disk generates a 22-character appendage to the base name as follows:



The previous calendar date is the current date in Julian format, and the time of day is in hours/minutes/seconds. Therefore, the first file's data set name will reflect the date and time it was created, and will always be unique. An example is SYS.MIGRATED.DMSC.C198827.T113015. When considering a base name makes sure that there is enough room for the CA Disk generated appendage.

Assignment of Creation Dates to Output Data Sets

As a default, CA Disk resets the creation date of data sets moved to tape to the current date. This is done because the data set is truly a new data set moved to tape and because some tape management systems rely on the creation date to be equal to the current date (CA 1 Tape Management has this requirement). If you desire that CA Disk leave the creation date on tape as it was found on DASD, specify sysparm MIGCREDT with a value of Y.

Migrating Protected Data Sets

Password-indicated data sets can be selected for migration by using the PASSWORD parameter on the MODESET command. Since conventional tape support does not allow both indicated and non-indicated data sets to be placed on the same tape, this parameter causes ONLY indicated data sets to be scanned for possible selection. If selected and migrated, the original password will be needed to access the tape data set.

If it is desired to process non-indicated data sets as well, a separate job execution is made but without the PASSWORD parameter. This technique results in two different series of tapes being created, one with only indicated data sets and the other with only non-indicated data sets.

Reports

The report generated by migrating one or more data sets to tape can be produced in three different sequences:

1. In data set name sequence
2. In data set name sequence within volume
3. In exact processing sequence (unsorted)
4. By default, the first and second sorted reports are both produced. The third order is produced only if the default sequencing options are changed via sysparm MIGRSORT. Any combination of the three report sequences can be specified.

Restrictions

Sequential data sets with the following attributes are not currently supported:

- USER LABELS — not to be confused with standard labels used on both disk and tape.
- TRACK OVERFLOW — blocks larger than track size.
- DBRC — Recon data sets, if migrated with Sequential Migrate, can cause abends and should be excluded from processing.
- IDRC — this type of hardware compression is unavailable by default. To instruct CA Disk to use an IDRC tape drive, simply add TRTCH=COMP to your JCL DD statement.

Condition Codes

The following table is a list of condition codes that can occur while executing Sequential Migrate:

0

Normal completion, no exceptions (execute remaining job steps)

4

Exceptions, but some data sets were successfully processed (execute remaining job steps)

8

Exceptions, no data sets were processed (bypass remaining job step)

JCL - Sequential Migrate

The moving and recataloging of sequential DASD data sets to tape is accomplished by use of the following JCL:

```
//RECAT EXEC MIGRATE  
//SYSIN DD *
```

Sequential Migrate Commands

MODESET, MIGRATE, and SCAN commands as described below.

- Only a single MODESET command is permitted and must be first in the command stream.
- Multiple MIGRATE and SCAN commands can be entered.

MODESET Command

Use this command to restrict the function of all subsequent commands executed in the job step. Only one such command is permitted and, if present, it must be the first in the input stream.

The MODESET command syntax is:

MODESET SIMULATE,NORECATLG,NOSCRATCH,PASSWORD,DMYCATLG,MIGSMSDS

SIMULATE

This parameter results in messages being displayed as if migration were taking place. However, no data sets are written to tape; no catalog updating or scratching occurs.

NORECATLG

Specification of this parameter deactivates the recataloging of the migrated data sets that would typically be done after their movement to tape.

NOSCRATCH

Specification of this parameter precludes the scratching of the data sets from DASD after they are migrated.

PASSWORD

Specification of this parameter causes processing to include only password-indicated data sets. All nonindicated data sets are bypassed, so the output volume contains the indicated data sets. The same CA Disk password protects each data set on the output volume.

DMYCATLG

Specification of this parameter causes CA Disk to catalog each CA Disk-generated first file. The default is not to catalog these data sets.

MIGSMSDS

Specification of this parameter causes CA Disk to migrate SMS managed data sets provided the requestor has SMS Storage Administrator Authority.

APPEND

Specification of this parameter causes CA Disk to look for previously used volumes and append more data sets thus fully utilizing large capacity volumes.

MIGRATE Command

A single specific data set is explicitly moved from disk to tape by use of the MIGRATE command. More than one of these commands can be included in the input stream. Explicit requests are not limited by the sysparm options addressed in Installation Options.

The MIGRATE command syntax is:

MIGRATE DSNAMES, VOLUME=

DSNAMES=

The name of the data set to be moved to tape. A relative GDG name is permitted.

VOLUME=

If the system catalog does not contain the serial number of the DASD volume on which the data set resides, this parameter is required.

SCAN Command

One or more volumes can be scanned to accomplish the implicit migration of data sets that meet user-specified criteria.

The SCAN command syntax is:

```
SCAN VOLUMES=,DSNAMES=,EXEMPT,METABLE=,GDGSONLY,NONGDGS,MINSIZE=,MAXSIZE=,  
      TYPE=,LASTUSE=,DATE=,GENER=,ANY
```

Note: The volumes= parameter and at least one of the selection criteria (MIN=, MAX=, LAST= or GENER=) is required. If selection based solely on the DSN= list is desired, code MIN=0. MIN=, MAX= and LAST= apply to both GDG and non-GDG data sets. GENER= applies only to GDG data sets. For example, coding GENER= and MIN= on the same command implies two tests for GDGs, one for non-GDGs.

VOLUMES=

This required parameter contains a list of one or more volume serial numbers or prefixes to be scanned. (A MAXIMUM of 50 entries is supported.)

DSNAMES=

A list of one or more data set names or patterns to which migration is to be confined. (A maximum of 10 entries is supported.)

EXEMPT

This parameter causes an exemption list to be examined before migrating any data set. Unless the MET= parameter specifies an alternate member name, the exemption list is assumed to be in parmlib member MIGRATXM.

METABLE=

If an exemption list has been placed in a member other than MIGRATXM, specify the member name as the value of this parameter.

GDGSONLY

The presence of this parameter limits the data sets being considered for processing to only those that have GDG names.

NONGDGS

The presence of this parameter limits the data sets being considered for processing to only those whose names are NOT of the GDG format.

MINSIZE=

Specify a one to nine digit number indicating the minimum size of a data set that is allowed to be migrated. Data sets equal to or larger than the size given will be selected. This optional selection parameter is specified in units indicated by the TYPE parameter.

MAXSIZE=

Specify a one to nine digit number indicating the maximum size of a data set that is allowed to be migrated. Data sets less than or equal to the size given will be selected. This optional selection parameter is specified in units indicated by the TYPE parameter.

TYPE=

Specify the type of units in which the size criterion has been expressed in the MINSIZE or MAXSIZE parameter.

- K—represents Kilobytes (the default type)
- C—represents Cylinders
- T—represents Tracks

LASTUSE=

This optional selection criterion causes unused data sets to be migrated. Specify the number of days that a data set is permitted to remain on disk without being used. When this number of days has elapsed, the data set is selected for migration.

DATE=

The current date is normally used to determine the number of days since a data set was last used. An alternative date to be used in this calculation can be specified for this parameter. It must be in an accepted CA Disk format.

GENER=

This optional selection criterion is used to specify the number of generations of each GDG data set that are to be kept on disk. All older generations will be selected for migration. For example, GENER=2 is interpreted to mean *keep the two most current generations of each GDG on disk and migrate the others to tape*. In other words, DSN=THE.GDG(0) and DSN=THE.GDG(-1) are kept on disk; the (-2), (-3), and so on levels are selected for migration.

Exceptions to the number specified on the GENER= parameter can be supplied in member GDGDASDG. As each GDG data set is being processed, entries in the GDGDASDG member will be searched for a matching data set name or pattern. If found, the number of generations kept on DASD will be the number given in the member entry. If no match is found, the global value supplied by the GENER= parameter will be used. See GDG DASD Generations List in the *Systems Guide* for more details.

Note: The GDGDASDG member is not searched unless the GENER= parameter is present.

ANY

When more than one selection test is specified, this parameter causes the data set to be migrated if ANY single test is satisfied. The default requires that ALL multiple tests be satisfied.

Migration Concerns

A duplicate copy tape for backup purposes can be created concurrently with the primary tape. The following installation requirements must be met for this capability to be used:

- Sysparm MIGPSCOP must be specified with a value of Y.
- The //MIGRATEC dd statement must be in the JCL.
- The FILES must have the MIGRECAT subfile properly initialized
- Consult your local CA Disk installation and support personnel to determine if this option is being used, or to request a change.

The MIGRECAT file is used to contain a record for each data set that is migrated to a primary and copy tape. When a primary tape volume cannot be used, the RECATALOG command is used to read these records and recatalog all of the data sets on the primary volume to the backup volume.

It must also be noted that CA Disk does not provide any formal automated means for managing the entries in the MIGRECAT file. CA Disk has no way of knowing how long to keep them. For this reason, the file will need to be purged occasionally to keep it from becoming excessively large. To accomplish a purge of this (or any subfile), merely unload the FDS, excluding MIGRECAT (filename) from the list of files, and then reload the unloaded records. See FDS Utilities in the *Systems Guide* for more information on UNLOAD and RELOAD.

Catalog Concerns

Also be aware that catalogs will fill up unless some measure is taken to purge unwanted entries. This can be aided by limiting the migration function to GDG data sets. As new generations are created, the older generations cataloged to tape will automatically be uncataloged as part of normal GDG processing.

MIGRECAT Function Description

The MIGRECAT utility allows you to maintain the information in the MIGRECAT subfile and to recatalog data sets from 'old' volumes to the copies of those volumes that were created by Sequential Migrate when the MIGDSCOP Sysparm was specified as 'Y'.

The maintenance of the MIGRECAT information is accomplished by the use of the following JCL:

```
//RECAT EXEC MIGRECAT  
//SYSIN DD *
```

RECATALOG Command

The RECATALOG command accepts the required operand OLDVOLNO and will recatalog to the copy volume the data sets that had been migrated to it. The command may also have the SIMULATE parameter specified to determine what will be done.

The RECATALOG command syntax is:

```
RECATALOG OLDVOLNO=,SIMULATE
```

OLDVOLNO=

The volume serial of the original volume that is no longer available.

SIMULATE

The presence of this keyword causes messages to be produced as if recataloging was taking place when, in fact, no catalog changes occur.

PURGE Command

The PURGE command accepts the optional operand OLDVOLNO and will check the data sets that had been migrated to volumes that match the volume specified to determine if they are still cataloged to it, and will purge the MIGRECAT records for any data sets that are no longer cataloged to that volume and that the expiration date has expired. The optional operands NOEXP and NOCAT instruct the utility to ignore the expiration date and catalog status in the eligibility determination. It also accepts the SIMULATE operand and will perform a test run if it is specified.

The PURGE command syntax is:

```
PURGE SIMULATE, VOLUMES=, DSNAME=, LIST=, SCRATCH=, DELPROF=, NOVALIDATE, FORCE
```

OLDVOLNO

The volume serial, or a pattern, of the original volume that should be checked for Purge eligibility.

SIMULATE

The presence of this keyword causes messages to be produced as if purging was taking place when, in fact, no MIGRECAT records will be purged.

NOEXP

The presence of this keyword causes the expiration date that was stored in the MIGRECAT record to be treated as if it has expired. This is the only method to purge records with a permanent retention date.

NOCAT

The presence of this keyword eliminates the check of the data set's catalog status. It will cause the data set to be treated as if it were no longer cataloged to the original output volume.

LIST Command

The LIST command will print MIGRECAT records that match the pattern specified in the optional operand OLDVOLNO.

The LIST command syntax is:

LIST OLDVOLNO=

OLDVOLNO=

The volume serial, or a pattern, of the original volume that should be listed.

Chapter 16: DASD Billing

This section contains the following topics:

[About Billing](#) (see page 533)

[General Information](#) (see page 534)

[Billing Rates](#) (see page 535)

[User Code/Account Codes](#) (see page 538)

[User Exits](#) (see page 541)

[Accumulate JCL for Non-VSAM Data Sets](#) (see page 542)

[Accumulate JCL for VSAM Clusters](#) (see page 545)

[DASD Billing Total JCL](#) (see page 546)

[Implementation - Suggested Run Procedures](#) (see page 547)

[Definitions of Fields in the DASD Billing Report](#) (see page 550)

[DASD Billing Samples](#) (see page 552)

About Billing

This CA Disk facility provides billing capabilities for the space occupied by non-VSAM data sets and VSAM clusters on DASD. Periodic VTOC scans are made to determine the space occupied by each individual data set. Use assigned billing rates are then applied and accumulated. The VTOC scans can be run as frequently as desired. Each run's billing amount is based on current space occupancy assumed to have existed over the period since the previous scan.

Commands available: ACCUMULATE, BILLING, and EXTEND.

JCL Proc executed: BILLING, VSAMBILL, DMS, and EXTEND.

General Information

The CA Disk DASD Space Billing function accumulates a billing charge against each data set based on two criteria:

- The SPACE OCCUPIED by (allocated to) each data set
- The NUMBER OF DAYS it occupies that space.

The product of these two values provides a billing quantity in either TRACK-DAYS or KILOBYTE-DAYS. The user must supply the billing rate for either one track-day or one kilobyte-day of occupancy. A simple multiplication of the billing quantity times the billing rate yields the billing charge for each data set. For example, a two-track data set residing on DASD for 50 days provides a 100 track-day billing quantity. If the billing rate is one cent per track per day, the charge for this period is one dollar.

To find the space occupied by each data set (item 1 previous), periodic VTOC scans are made and space allocations calculated.

There are two methods available to determine the days of occupancy for data sets. They are referred to as MODE=ELAPSED and MODE=IMMEDIATE.

Elapsed Mode

With MODE=ELAPSED a data set is not charged for a day of occupancy until it exists on the volume one full day. The period of occupancy (item 2 previous) is computed by counting the number of days from the current run date back to one of three dates described below, whichever is most recent.

- The date of the last previous billing scan
- The creation date of the data set,
- A BEGIN= date supplied on the billing command.

CA Disk assumes that the current space occupancy has existed over the period computed. The advantage of ELAPSED mode processing is that a user gets billed for DASD space only after the data set exists for a full day. Disadvantages include that transient data sets do not ever get billed, and you must run two billing steps on the first day of each cycle, one to complete the last cycle and one to start the next. When using this mode, watch for data sets that are deleted and reallocated each day. Depending on the time of the billing run, they can never have charges accumulated for them.

Immediate Mode

IMMEDIATE mode indicates that a data set is charged for a day of occupancy each time it is encountered during a Billing Accumulate run. The maximum is once per day, but the data set need not exist for a full day before it starts accumulating charges.

The advantage of IMMEDIATE mode of processing is that any data set existing at the time of the billing run has charges accumulated for it, and there is no need to make a special run on the first day of a billing cycle to close out the prior cycle. The biggest disadvantage is that a data set can have existed only during the time the billing run is executed and will get a full day's charge against it. This technique also catches data sets scratched and recreated each day.

Billing Rates

The user must select a billing rate for each allocated track or kilobyte per day. Different billing rates can be selected and assigned to different volumes or groups of volumes. As an example, TSO volumes can be billed differently from IMS volumes, or 3350 volumes differently from 3380 volumes. All volumes to be billed at the same rate must be assigned a CATEGORY CODE in the PARMLIB member DSBCATEG. The associated billing rate for each category code must then be entered into parmlib member DSBRATEC.

Basically, a category code is a code assigned to a group of volumes that are to be assigned the same per-day billing rate.

Computing the Billing Charge

As each volume is scanned, its matching category code is retrieved from member DSBCATEG. This code is then used to find the proper rate in member DSBRATEC. Each data set's allocated space and billing period are computed, the rate is applied, and the result is added to the accumulating charge. These charges are stored and updated for each individual data set. The records are placed in the DASDSPCB subfile of the files data set. The VTOC SCAN and ACCUMULATE runs can be executed as often as needed to achieve the data set sampling required.

Assign Volumes to Billing Categories

Volumes that are to be assigned the same billing rate must be given a category code and placed in member DSBCATEG in the parmlib data set.

The format of entries in the DSBCATEG member is: CCVVVVVV

CC

The CATEGORY CODE.

VVVVVV

A VOLSER or PREFIX.

A / can be used to wildcard any or all of the VOLSER. This is the only wildcard that will work for this parameter.

Define Billing Rates for Each Category

Billing rates for each category code must be placed in member DSBRATEC of the parmlib data set. The billing rate is the amount to be charged to the user for each 1000 bytes of DASD space allocated to a data set each day, or the amount to be charged to a user for each track of space used each day. This depends on sysparm DSBUNITS.

The format of entries in this member is: CCRRRRRR

CC

The CATEGORY CODE

RRRRRR

The ASSOCIATED BILLING RATE

The CATEGORY CODE in the DSBCATEG member is matched to those in the DSBRATEC member to find a rate to apply to the volume being processed.

The rate specified by RRRRRR (ASSOCIATED BILLING RATE.) can be a 1- to 5-digit value if the decimal point is also specified, or a 6-digit value if the decimal point is omitted. Whenever you omit the decimal point, CA Disk right-justifies the rate; that is, zeros are added on the left until all six positions are filled. CA Disk then assumes that the decimal point is to precede the first digit (for example, a 4-digit value of 0621 means .000621, not .0621!). You can easily avoid this minor degree of confusion, however, by always specifying the decimal point in the rate field, or always supplying a full character field. The decimal point can be placed anywhere in the RRRRRR field.

The billing rate defaults to being interpreted as a dollars and cents value. To use currencies other than the dollar, see Using Other Currencies.

Example of Assigning Billing Rates

A billing rate of \$ 0.008500 is to be applied to data sets on volumes VOL001, VOL006, and VOL007. A billing rate of \$ 0.007800 is to be applied to volumes VOL002 and VOL004. All volumes beginning with TSO are to be billed at a rate of \$ 0.005562.

Category codes are defined in member DSBCATEG:

```
'12VOL001'  
'12VOL006'  
'12VOL007'  
'13VOL002'  
'13VOL004'  
'20TSO/XX'
```

Billing rates for each category are defined in member DSBRATEC:

```
'12008500'  
'13.0078'  
'20005562'
```

Billing Units System Parameters

The billing unit, either track-days or kilobyte-days, is indicated by the following system parameter.

DSBUNITSuuu System parameter - Specify TRK for the appended uuu if billing is to be done by track-days rather than by the default of kilobyte-days. K/B can be used to specify the default, but it isn't needed.

Using Other Currencies

If you desire to accumulate and charge for DASD space in a unit of currency other than dollars and cents, such as Japanese yen or Italian lire, specify sysparm DSBALTRT with a value of 1. This changes the billing rates assigned in the DSCRATEC member from a dollars and cents value to just a unit of currency expressed with decimal places. If this feature is used, specifying 993.25 in the DSBRATEC member would indicate that for all volumes assigned to category code 99, either 3.25 lire or 3.25 yen are to be charged for each track or kilobyte day. (To review the rules explaining the format of the DSBRATEC table entries, see Define Billing Rates for Each Category.) The rate value shown on the billing reports will be shown as five digits with two decimal places, but the amounts and totals will be shown as full numbers with no decimal places.

Note: This alternate rate specification can also be used for other forms of currency as appropriate.

User Code/Account Codes

When a scan and accumulate run encounters a data set for the first time (no prior billing entry), a user/account code is assigned that becomes a part of its billing record in the DASDSPCB subfile. The user code assigned is significant in that billing reports are grouped and subtotaled for each unique user code. The default user code is the first 12 bytes of the data set name, but it can alternately be specified in any of the following ways:

- With a displacement into the format-1 DSCB where a user code can be found.
- By providing a list of user codes matched to data set names in member USERCODT of the parmlib data set.
- As one of the index levels of the data set name.

Since the user code is assigned and placed in the billing subfile record the first time the data set is processed, the manner used to assign user codes should be implemented prior to running DASD billing; that is, user code assignments cannot be dynamically updated by changing the method by which they are assigned. To change an assigned user code, either user exit DSBUSREX must be used, or the billing subfile entries to be reassigned must be erased during the EXTEND processing by the use of the REMOVE parameter and DASD billing rerun. (Erasing destroys the affected records, which necessitates the rerun.)

A 40-character description can optionally be associated with each user code. It will be printed next to the user code when the billing totals are produced.

Assigning User Codes

The first item that controls the assignment of a user code is sysparm DSCBUSRD. It is used to provide the displacement into the DSCB to find the user code field. If this displacement is zero, which is the default, an attempt is made to find a user code in member USERCODT. If this member is defined, it can supply the code. But by default it is not defined and processing returns. When this member check returns and no code has been assigned, either the DSCBUSRD sysparm-defined displacement is used to find the user code, or an index level of the data set name is used. Since the index level technique must be activated specifically by sysparm DBDSNINX, the resulting default is to use the displacement method. Each of the possible methods is defined in more detail below.

Method 1: Displacement

DSCBUSRDxxx

System parameter- The displacement of the user code in the DSCB must be specified as the xxx value. The default value is 000.

USRCLENxx

System parameter- The length of the user code in the DSCB must be specified as the xx value. The default (and maximum allowed) is 12.

- When the value of the displacement is zero, an attempt is made to find a user code in member USERCODT of the parmlib data set.
- If the displacement is non-zero, the member is never inspected.

Method 2: PARMLIB Member USERCODT

An attempt to find the user code in this member is made only if the displacement is zero. Entries in this member have the following format:

'CCCCCCCCCCCCDDDD...DDDD'

CCCCCCCCCCCC

The 12-character user code.

DDDD....DDDD

A full data set name or a data set name pattern. If the CCC field is less than 12 characters, pad it with blanks out to the DDD field and enclose the entire entry in quotes.

Method 3: Index Level (Node)

DBDSNINXn

System parameter-Change the appended N to 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 USRCLEN.

DBDSNCFDx

System parameter- Specify for the appended x the index level of the data set name that is to be used as a user code. The default value is 1, but activated only if sysparm DBDSNINX is specified with a value of Y.

User Code Description Table

Descriptions can be defined for each user code by placing them in member DSBUSERD of the parmlib data set as follows:

DSBUSERD MEMBER

The format of entries in this member is:

```
'CCCCCCCCCCCCXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
```

CCCCCCCCCCCC

The 12-character user code

XXXX.....XXXX

A 40-character description for the user code. If the user code is less than 12 characters in length, pad with blanks and enclose the entire entry in quotes.

Total Charges

Charges accumulated over a billing period, perhaps daily runs over a month's time, can be totaled (extended) as desired. This causes the accumulated charges to be reset to zero in order to begin the accumulation for a new billing period. Detailed charges are printed for each data set as well as subtotals by user codes and totals by volume and device type.

Exempting Data Sets

Member DSBEXCLU in the parmlib data set can be used to exempt data sets from the billing process. Simply enter their full data set names or matching patterns into the member. Model DSCBs are always excluded by default.

Note: The DSBEXCLU parmlib member only excludes data sets from the data accumulation phase of the Billing process, and not during the extend reporting job. It is therefore possible to exclude data sets from billing for a portion of the billing cycle, and still have them show up on the report for those days that billing was active for the data sets.

Example entries in parmlib member DSBEXCLU:

```
CAI.DISK.CCUWLOAD  
CAI.DISK.CCUWPARM  
ABC.DISK.PARMLIB  
ABC.DISK.FILES  
SYS/
```

User Exits

Specifics regarding the use of user exits can be found in User Exits in the *Systems Guide*. Only a brief description is presented here.

DSBUSREX

This exit gives control to the user program when every billing record is being inserted or updated. One use of this exit is to reset a previously assigned user code. For complete details of this user exit, see DSBUSREX —Screen Billing Record Inserts in the *Systems Guide*.

USRCODEX

This exit gives control to the user prior to looking in member USERCODT for the user code to be assigned. The exit can supply the user code directly. For complete details of this user exit, see USRCODEX —User Code Substitution Exit in the *Systems Guide*.

Condition Codes

The following table is a list of possible condition codes that can occur during the execution of any PROC related to DASD Billing:

Proc Step	CC	Description
Select	0	Successful completion
	4	Command rejected
Accumulate	0	Successful completion
	4	Unable to update one or more records in billing file
	8	Command rejected
	16	Input data set from sort cannot be opened
Extend	0	Successful completion
	4	Nothing can be totaled (no billing data)
	8	Command rejected

Accumulate JCL for Non-VSAM Data Sets

The following JCL is required to accomplish the scan and accumulation for billing records placed in the DASDSPCB subfile.

```
//DAILY    EXEC BILLING
//SELECT.SYSIN DD *           <-- NOTE use of STEP NAME
//ACCUM.SYSIN DD *           <-- NOTE use of STEP NAME
```

The SELECT and ACCUMULATE commands are described on the following pages.

SELECT Command

The SELECT command is used to scan DASD volumes and retrieve the DSCB for each data set on the volume.

The SELECT command syntax is:

```
SELECT VOLUME=,DSNAME=
```

VOLUME=

A list of one or more volumes or volume prefixes to which scanning is to be confined. (A maximum of 100 entries is supported.)

DSNAME=

A list of one or more data set names and patterns to which the processing is to be limited. (A maximum of 25 entries is supported.)

ACCUMULATE Command

The ACCUMULATE command is used to accumulate the billing charges for the data sets on each DASD volume.

The ACCUMULATE command syntax is:

```
ACCUMULATE DATE=,MODE=,BYPASSNEW,ADDMISSING,BEGIN=,BEEK=,BMONTH,  
BMEMBER=,SIMULATE
```

DATE=

A date other than the current date can be substituted for the current date. This allows users to run billing as if it had been run on a different date. The date can be in any accepted CA Disk format.

Note: Data sets with a creation date greater than this date will be bypassed.

MODE=

Two different techniques of billing are supported. They are selected through the MODE= parameter. When MODE=IMMEDIATE is specified, occupancy days are calculated each day a data set is found during the ACCUMULATE process; that is, if a data set is seen today, it will be billed for the day.

- When MODE=ELAPSED is specified, occupancy days are calculated based on the number of days the data set has existed since the last ACCUMULATE process.
- The primary difference between these two modes is that IMMEDIATE calculates occupancy days immediately, as soon as a data set is found on a volume, while ELAPSED requires that the data set exist across ACCUMULATE runs on two different days before charges will be accumulated.

BYPASSNEW

Valid only if MODE=IMMEDIATE is specified, the BYPASSNEW parameter indicates that if a data set's creation date is the same as the current date, no occupancy days are to be accumulated for it. The purpose of this command is to provide a way to avoid billing for transient data sets (those that are on the volume less than one day).

ADDMISSING

Valid only when MODE=IMMEDIATE is specified, the ADDMISSING parameter indicates that processing is to calculate occupancy days for all of the days the data set existed since the Accumulate processing was previously run.

- If Accumulate processing has not previously processed a data set in this billing cycle and the ADDMISSING parameter is specified, occupancy days will be determined by the date of the beginning of the billing cycle or the create date of the data set, whichever is later.
- If not specified, only one occupancy day is calculated each day the accumulate is run.

BEGIN=

This date is used to determine the beginning date of the billing cycle. No data sets will be billed for occupancy days before this date. The date can be in any accepted CA Disk format. (See BWEEK=, BMONTH, and BMEMBER= parameters also)

BWEEK=weekday

For users implementing a weekly billing cycle, the BWEEK= parameter is an alternative to the BEGIN= parameter for setting the begin date of the billing cycle. Specify the name of a day in the week, and the beginning of the billing cycle will automatically be set each week to the date corresponding to the specified weekday name; for example, BWEEK=MONDAY.

BMONTH

For users implementing a monthly billing cycle, the BMONTH parameter is an alternative to the BEGIN= parameter for setting the begin date of the billing cycle. When BMONTH is specified, the beginning of the billing cycle will automatically be set each month to the date corresponding to the first day of the month.

BMEMBER=

This parameter is an alternative to the BEGIN= parameter for setting the begin date of the billing cycle. The BMEMBER= specifies a member name in the parmlib that contains begin dates of billing cycles. When BMEMBER= is specified CA Disk scans the specified member name to determine the begin date of the active billing cycle.

The entries must be in an accepted CA Disk format. Sample entries in the parmlib member:

01/01/95

04/01/95

07/01/95

10/01/95

01/01/96

To determine the BEGIN date to use, CA Disk scans the member entries searching for the highest date that is equal to or prior to the current date. For example, using the previous entries, if the date of the accumulate run is 02/15/95, the Begin date used would be 01/01/95.

To use this parameter, first create a new member in the parmlib data set with the dates corresponding to your billing cycle begin dates. Then specify the member name in the BMEMBER= parameter.

SIMULATE

The presence of this parameter will cause the processing to take place, but no billing records will actually be updated.

Accumulate JCL for VSAM Clusters

The following section describes the JCL and control statements needed to run the BILLING functions against VSAM clusters. It is equivalent to the SCAN and ACCUMULATION process used for non-VSAM data sets. This is the only portion of the billing function that is unique to VSAM cluster billing.

Keep in mind that the EXTEND procedure will process both the VSAM and non-VSAM records.

```
//jobname JOB (acct,info),etc.
//DAILY EXEC VSAMBILL
//DMS.SYSIN DD *                               <-- NOTE use of STEP NAME
* Specify the following DSCL commands for VSAM billing. *
PASSWORD MASTER=,CATALOG=                      (optional)
SCAN CATALOGS=/                                (required)
    EXCLUDE CRITERIA=(DSORG,NE,VSAM)             (required)
    SELECT CRITERIA=,DSN=,VOL=,TABLE=            (optional)
    EXCLUDE CRITERIA=,DSN=,VOL=,TABLE=          (optional)
    BILLING                                       (required)
//ACCUM.SYSIN DD *                               <-- NOTE use of STEP NAME
* ACCUMULATE COMMAND as described for non-VSAM *
```

A sample job to accumulate VSAM billing records is as follows:

```
//jobname JOB (acct,info),etc.
//DAILY EXEC VSAMBILL
//DMS.SYSIN DD *
SCAN CATALOGS=/
    EXCLUDE CRITERIA=(DSORG,NE,VSAM)
    BILLING
//ACCUM.SYSIN DD *
    ACCUMULATE BMONTH,MODE=IMMEDIATE,BYPASSNEW
```

DASD Billing Total JCL

The following JCL is required to generate (extend) the billing totals at the end of a billing cycle:

```
//TOTALS EXEC EXTEND
```

```
//SYSIN DD *
```

EXTEND COMMAND and PARAMETERS as described below

Note: In order to run the EXTEND JCL, a restart data set must be allocated. If this has not been done, see Activating DASD Billing in the *Installation Guide*.

The EXTEND command causes billing records and reports to be produced and is used at the end of a billing cycle.

The EXTEND command syntax is:

```
EXTEND VOLUME=,DSNAME=,SIMULATE,REMOVE
```

A list of one or more volumes and volume prefixes to which processing is to be confined. Omission of this parameter results in no exclusion on the basis of volume. (A maximum of 50 entries is supported.)

DSNAME=

A list of one or more data set names and patterns to which the run is to be limited. (A maximum of 50 entries is supported.)

SIMULATE

Inclusion of this parameter will cause the reports to be produced without the updating of the billing records.

REMOVE

Will cause all selected entries in the DASD Billing Subfile (DASDSPCB) to be removed from the Subfile at the end of the processing. Entries that were not selected by use of the VOLUME= and DSNAME= parameters will not be removed.

Implementation - Suggested Run Procedures

The following sequence of steps is suggested as an approach to doing regular DASD BILLING on a monthly billing cycle:

1. Initialize a separate FDS just for DASD billing.
2. Specify the BMONTH parameter in the ACCUMULATE command to implement a monthly billing cycle that sets the begin date automatically to the first day of each month. Also, specify MODE=IMMEDIATE to calculate occupancy days for each day a data set is found during the ACCUMULATE processing.
3. Execute the SCAN and ACCUMULATE job once a day, every day, during the monthly billing cycle.

For example, August has 31 days so SCAN and ACCUMULATE should be run for 31 days.

4. At the end of the month (the last day):

Run the EXTEND job to generate the billing totals, print the reports, and zero the accumulation values. The billing subfile records can be deleted by adding the REMOVE subparameter to the EXTEND command for the EXTEND job. If the records are not erased all data sets that existed one month but not the next will continue to appear, but with zero billing amounts. The REMOVE parameter affects only records that have been selected via the DSNAME= and VOLUME= parameters.

Billing Processing Considerations

If a data set is scratched during a billing period and is later recreated or restored, the creation date of the new data set determines how charges are accumulated for it if `MODE=ELAPSED`, or `ADDMISSING` is being used.

When a data set is restored with the CA Disk Restore function or moved with the Move/Copy function, it is by default given its original create date. For the purpose of determining the data sets period of occupancy on DASD, this old date is misleading and can result in overcharges.

There are two options to help avoid this overcharging:

- Option 1 — Use the `CREATE` parameter in either Restore or Move/Copy functions. This parameter sets the creation date to the current date.
- Option 2 — Specify the `MODE=IMMEDIATE` without the `ADDMISSING` parameter in the command, and the data set will be charged for only one day no matter when it was restored. If the creation date is greater than the creation date of the original data set, charges will be accumulated for only the days the data set actually existed.

Note: If the creation date of the restored data set is the same as the creation date of the original data set before it was scratched and the `ADDMISSING` parameter is specified, the data set will be charged for all days in between the two creation dates, even if it did not exist on the volume during that time.

DASD Billing and Archival Functions

If both DASD Billing and archiving functions are being used, it is probably desirable to run the billing accumulations just prior to `ARCHIVING`. If this is not done, the *offending* data sets escape their last billing.

If an abnormal termination occurs at any point during the `SCAN` or `ACCUMULATE` functions (`BILLING JCL` procedure), you can restart processing by simply rerunning the entire procedure. This is possible because CA Disk will not accumulate charges for a data set that has already been processed on the same day. Some data sets will be reanalyzed, but no additional charges will be accumulated for them.

If an abnormal termination occurs at any time during the extend function (`EXTEND JCL` procedure), you can simply rerun it. This is possible because of the restart data set associated with extend processing.

Increased Billing Buffers

A default buffer size to hold entries in memory is controlled by the following sysparm. If this default value is not large enough due to the number of entries placed in the member, a diagnostic message is issued. The sysparm must be changed to increase the buffer size.

DSBCATSZxxxxx—The buffer size for the category codes. The default value is 00500.

Definitions of Fields in the DASD Billing Report

The following table is a list of definition of fields in the DASD Billing report:

USER CODE

The user code is the account code used in DASD Billing to group particular data sets for a billing report. It is specified by the user in the USERCODT member of parmlib or by use of format-1 DSCB displacement or index level processing.

DESCRIPTION

This is descriptive information for the user code. It is entered in the DSBUSERD member of parmlib, and is used for printing purposes only.

dataset NAME

The name of the data set for which billing is being done.

VOLUME

The volume on which the data set occupies space.

DEVICE TYPE

The type of device of the volume.

CREDIT

The creation date of the data set (Julian format).

USEDT

The date the data set was last used (Julian format).

LAST SCAN

The date that billing ACCUMULATE processing was last run for the data set.

ALLOC

The number of tracks that are allocated to the data set.

Note: The allocation figure represents the size of a data set at the time it was last examined. Unless it is a very static data set, its size can change from day to day. This circumstance will make it difficult for you to check the K/B DAYS or TRK DAYS field for accuracy through a manual calculation unless you already know that the data set has remained static in its allocation. The K/B DAYS or TRK DAYS fields, although correct in the report, can be misleading in such cases. keep this in mind when looking at your report.

MAX USED

The number of tracks that are actually being used by a data set as opposed to the number of tracks that are allocated.

K/B DAYS

The total number of kilobytes allocated to the data set multiplied by the number of days' occupancy. The algorithm is as follows:

$(\text{TRACKS} * \text{TRACK SIZE}) * \text{NUMBER OF DAYS OCCUPANCY} / 1000$

TRK DAYS

The total number of tracks allocated to the data set multiplied by the number of days' occupancy. The algorithm is as follows:

$\text{TRACKS} * \text{NUMBER OF DAYS OCCUPANCY}$

RATE

The rate of charge (per day) to be applied to a data set. The format of this field depends on which value is specified for sysparm DSBALTRT. If it is specified with a value of N, standard rate processing will be used and the format will be whole numbers. If it is specified with a value of 1, foreign currency rate processing will take effect and the format of the field will have a decimal point. (The last type of process is intended for currencies such as the Japanese yen or Italian lire.)

AMOUNT

The total amount that is charged for the data set during this billing cycle.

TOTAL data sets

The total number of data sets being billed in the billing cycle.

TOTAL ALLOCATED TRACKS

The total track allocation of all data sets that were billed in the billing cycle.

TOTAL K/B DAYS

The total number of K/B DAYS for all data sets in the billing cycle.

TOTAL TRK DAYS

The total number of TRACK DAYS for all the data sets in the billing cycle.

TOTAL AMOUNT

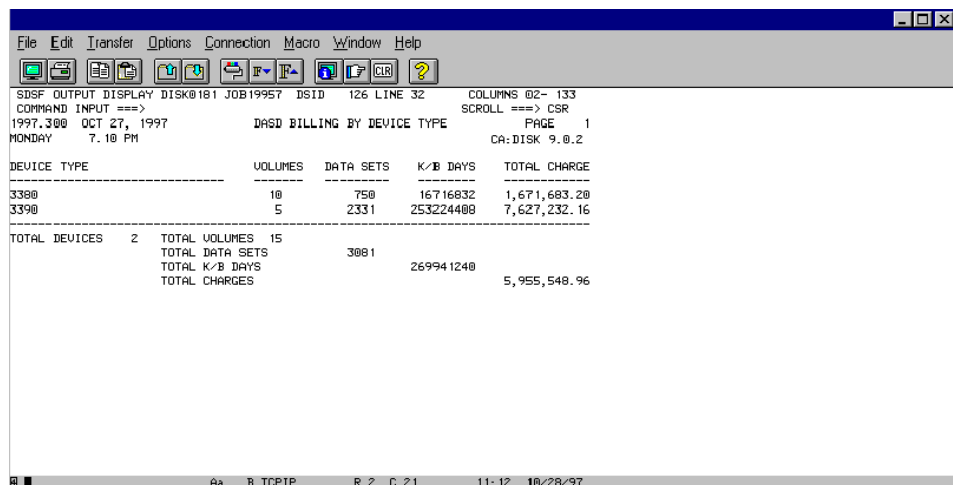
The total amount charged for all data sets in the billing cycle.

CA Disk Arithmetic Restriction

CA Disk DASD Billing uses fullword arithmetic in its calculations for totals. This restricts the highest number to hex 7FFFFFFF or decimal 2,147,483,647. If the number of K/B DAYS accumulated in a cycle is higher than this, the totals will be incorrect. However, the grand totals on the report will be correct.

DASD Billing Samples

The following pages contain sysout Billing samples including: Device Type, User Code, and Volume.

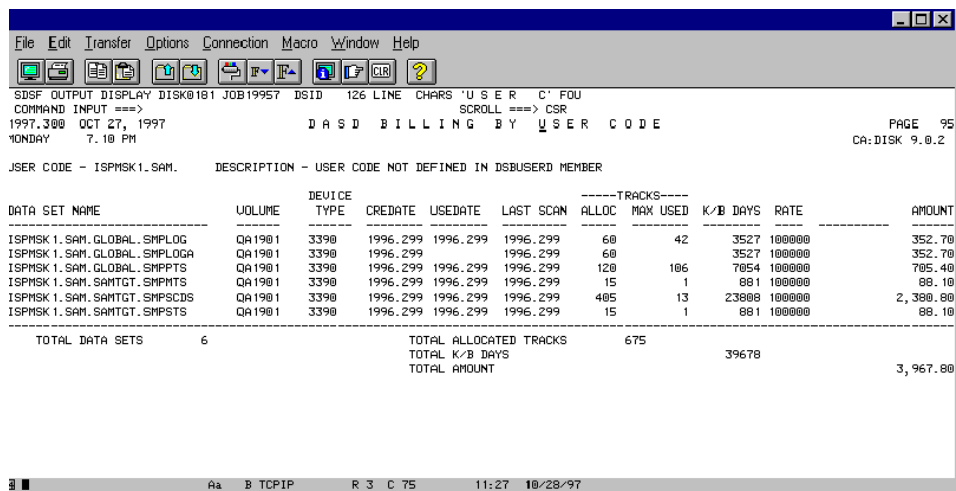


The screenshot shows a mainframe terminal window with a menu bar (File, Edit, Transfer, Options, Connection, Macro, Window, Help) and a toolbar. The main display area contains the following text:

```
SDSF OUTPUT DISPLAY DISK0101 JOB19957 DSID 126 LINE 32 COLUMNS 02- 133
COMMAND INPUT ==> SCROLL ==> CSR
1997.300 OCT 27, 1997 DASD BILLING BY DEVICE TYPE PAGE 1
MONDAY 7.10 PM CA:DISK 9.0.2
```

DEVICE TYPE	VOLUMES	DATA SETS	K/B DAYS	TOTAL CHARGE
3390	10	750	16716032	1,671,603.20
3390	5	2331	253224400	7,627,232.16
TOTAL DEVICES	2	TOTAL VOLUMES 15		
		TOTAL DATA SETS 3081		
		TOTAL K/B DAYS 269941240		
		TOTAL CHARGES		5,955,548.96

The status bar at the bottom shows: Aa B TCP/IP R 2 C 21 11:12 10/28/97.



The screenshot shows a mainframe terminal window with a menu bar (File, Edit, Transfer, Options, Connection, Macro, Window, Help) and a toolbar. The main display area contains the following text:

```
SDSF OUTPUT DISPLAY DISK0101 JOB19957 DSID 126 LINE CHARS 'U S E R C' FOU
COMMAND INPUT ==> SCROLL ==> CSR
1997.300 OCT 27, 1997 DASD BILLING BY USER CODE PAGE 95
MONDAY 7.10 PM CA:DISK 9.0.2
```

USER CODE - ISPMK1.SAM. DESCRIPTION - USER CODE NOT DEFINED IN DSUSERD MEMBER

DATA SET NAME	VOLUME	DEVICE TYPE	CREDATE	USEDATE	LAST SCAN	ALLOC	MAX USED	K/B DAYS	RATE	AMOUNT
ISPMK1.SAM.GLOBAL.SMPLOG	QA1901	3390	1996.299	1996.299	1996.299	60	42	3527	1000000	352.70
ISPMK1.SAM.GLOBAL.SMPLOGA	QA1901	3390	1996.299	1996.299	1996.299	60		3527	1000000	352.70
ISPMK1.SAM.GLOBAL.SMPPTS	QA1901	3390	1996.299	1996.299	1996.299	120	106	7054	1000000	705.40
ISPMK1.SAM.SAMTGT.SMPPTS	QA1901	3390	1996.299	1996.299	1996.299	15	1	881	1000000	88.10
ISPMK1.SAM.SAMTGT.SMPSCDS	QA1901	3390	1996.299	1996.299	1996.299	405	13	23008	1000000	2,300.80
ISPMK1.SAM.SAMTGT.SMPSTS	QA1901	3390	1996.299	1996.299	1996.299	15	1	881	1000000	88.10
TOTAL DATA SETS		6	TOTAL ALLOCATED TRACKS			675				
			TOTAL K/B DAYS				39678			
			TOTAL AMOUNT							3,967.80

The status bar at the bottom shows: Aa B TCP/IP R 3 C 75 11:27 10/28/97.

File Edit Transfer Options Connection Macro Window Help						
SDSF OUTPUT DISPLAY DISK0181 JOB19957 DSID 126 LINE 9,164 COLUMNS 02- 133						
COMMAND INPUT ==> SCROLL ==> CSR						
1997.300 OCT 27, 1997 DASD BILLING BY VOLUME PAGE 1						
MONDAY 7.10 PM CA:DISK 9.0.2						
VOLUME	DEVTYPE	NO SCANS	LAST SCAN	DATA SETS	K/B DAYS	TOTAL CHARGE
MUSS01	3380	12	97300	1	1284	128.40
QATEST	3380	161	97300	22	6713	671.30
QA1801	3380	161	97300	274	1717537	171,753.70
QA1901	3390	161	97300	1554	247953743	8,154,298.66
SMS80	3380	111	97300	34	4274834	427,483.40
SMS801	3380	161	97300	142	2128677	212,867.70
SMS802	3380	161	97300	125	3814167	381,416.70
SMS803	3380	161	97300	148	4774428	477,442.80
SMS901	3390	160	97300	256	1775912	177,591.20
SMS902	3390	160	97300	264	2895898	289,589.80
SMS903	3390	160	97300	251	598855	59,885.50
SSL803	3380	1	97118	1		.00
SSL804	3380	2	97253	2		.00
SSL806	3380	28	97196	1		.00
SSL807	3390	123	97300	6		.00
***** BOTTOM OF DATA *****						
TOTAL VOLUMES 15 TOTAL DATA SETS 3081						
TOTAL K/B DAYS 26994 1240						
TOTAL CHARGES 5,955,548.96						

Aa B TCP/IP R 2 C 21 11:35 10/28/97						

Chapter 17: ISPF

This section contains the following topics:

- [About ISPF Support](#) (see page 556)
- [Using the ISPF Support](#) (see page 557)
- [List of Available ISPF Functions](#) (see page 558)
- [ISPF Return Function Restriction](#) (see page 560)
- [ISPF Panel Examples](#) (see page 561)
- [Overview of ISPF Online Reporting Facility](#) (see page 580)
- [ISPF Online Reports](#) (see page 588)
- [Allocate Library for Report Definition](#) (see page 589)
- [Select Desired Report](#) (see page 590)
- [Display Report Information](#) (see page 590)
- [Specify New Report Information](#) (see page 591)
- [Select Report Field](#) (see page 592)
- [Specify Field Position on Report](#) (see page 593)
- [Select Sort Fields](#) (see page 595)
- [Specify Sort Field Information](#) (see page 597)
- [Select Any Sort Fields Where Subtotaling is Desired](#) (see page 598)
- [Select Subtotal Fields](#) (see page 600)
- [Specify Subtotal Information](#) (see page 601)
- [Select Grand Total Fields](#) (see page 603)
- [Specify Grand Total Information](#) (see page 603)
- [Online Reporting System \(Option 5\)](#) (see page 603)
- [Enter Data Set Name](#) (see page 604)
- [Specify Subset of Real DASD Volumes to Process](#) (see page 605)
- [Specify Selection Criteria You Desire](#) (see page 606)
- [Specify DSNAMES and/or Table](#) (see page 606)
- [Select Desired Reports](#) (see page 606)
- [Confirm DSCL Report Requests](#) (see page 607)
- [JCL for Batch Reports](#) (see page 608)
- [Browse Report](#) (see page 609)
- [Report Disposition](#) (see page 610)
- [Build Batch Commands](#) (see page 612)
- [Online Reporting - Expert Mode](#) (see page 614)
- [DSCL Command Edit](#) (see page 615)

About ISPF Support

The ISPF support of CA Disk provides an interface from the IBM System Productivity Facility, Dialog Management Services, to the CA Disk product. CA Disk uses ISPF dialog management service panels to give you an easy-to-use interface in an interactive environment.

This support can be used with the ISPF program product (5665-319) and subsequent releases.

The CA Disk dialog management functions either guide the user through the process of generating JCL to execute batch CA Disk facilities, or interface directly to the CA Disk foreground applications. No JCL is involved for foreground processing functions, as all requests are performed immediately in the TSO user's region.

All CA Disk dialog manager panels have associated HELP text. If at any time you need information on how to proceed or what to do, the <HELP> PF key can be pressed. The HELP information can be browsed until you are ready to continue. The <END> key will return you to the panel originally being processed.

Using the ISPF Support

For users authorized to use the CA Disk ISPF interface, the ISPF primary option panel will contain a selection option for CA Disk. Since this is an optional value, you can have to specify a different option. Examine your primary option menu for a line similar to the following example:

8 CA Disk - Perform Data Storage Management Functions

After option 8 is selected from the ISPF primary option menu, CA Disk will generate a menu of functions available. Since there are more options available than menu lines on a screen, it is sometimes necessary for CA Disk to implement two levels of menus. This occurs when more than 15 options are given to any user. If a user has fewer than 15 options, all options will be displayed on the same selection menu. If more than 15 options are specified, the first menu displayed will ask the user to select between DASD manager and user functions. After this has been specified, a second menu will be displayed with the appropriate functions listed.

Note: All users will not necessarily be presented a menu listing all available CA Disk ISPF functions. The DASD manager for each installation specifies what functions are presented to users. If you have a question about the functions you have been authorized to perform, contact your local CA Disk support personnel or your DASD manager.

Select the desired function from the ones listed on the menu. This will cause CA Disk to present a panel for the option specified. Use the ISPF HELP PF key to obtain HELP text on how to use the panel. To return to the function menu from the HELP menu, press the <END> key.

List of Available ISPF Functions

Following is a list of the CA Disk ISPF interface functions:

Explicitly Archive data sets

Generate JCL for batch job submission.

Queue Request to Archive data sets

Generate JCL for batch job submission.

Queue Request to Archive data sets ISPF

Foreground function that allows user to enter deferred archive commands.

Restore data sets (background)

Generate JCL for batch job submission.

Queue Request to Restore data sets

Generate JCL for batch job submission.

Queue Request to Restore data sets ISPF

Foreground function that allows user to enter deferred restore commands.

Restore data sets

Invoke TSO RESTORE command.

List data set Index ISPF

Foreground function that allows user to list entries only.

List and Update data set Index

ISPF foreground function that allows the user to list and delete entries for archived data sets.

List and Update data set Index

ISPF foreground function that allows the user to list and change expiration date values for archived data sets.

List and Update data set Index

ISPF foreground function that allows the user to list, change expiration dates, and delete entries for archived data sets.

List Queued Processing Requests

ISPF foreground function that allows the user to list deferred archive and restore requests previously entered.

List or Delete Queued Processing Requests

ISPF foreground function that allows the user to list and delete deferred archive and restore requests previously entered.

Move data sets From Disk to Disk

Generate JCL to invoke the Move/Copy using the COPY command.

Compress PDS data sets Generate JCL to invoke the PDS

Compress function for specific data sets.

Restart PDS Compression Process

Generate JCL to invoke the PDS Compress RELOAD function.

VTOC Scan — Generate CA Disk Reports

Generate JCL to invoke the REPORT function.

VTOC Scan - ARCHIVE, BACKUP, SCRATCH, or UNCATALOG

Generate JCL to invoke the IMPLICIT ARCHIVE (RETAIN) function.

VSAM Cluster Reporting

Generate JCL to invoke the EXPLICIT VSAM (FIND) function to generate VSAM reports only.

Catalog Scan — VSAM Cluster Reporting

Generate JCL to invoke the IMPLICIT VSAM function to generate VSAM reports only.

VSAM Cluster Maintenance

Generate JCL to invoke the EXPLICIT VSAM (FIND) function.

Catalog Scan — VSAM Cluster Maintenance

Generate JCL to invoke the IMPLICIT VSAM function.

Recover All or a Portion of a DASD Volume

Generate JCL to invoke the IMPLICIT .RECOVERY function

Data set Index Processing

Generate JCL to invoke the ARCHIVE INDEX maintenance procedure (IXMAINT).

Display Archive Volume Information

This function displays information about archive volumes in the foreground ISPF environment.

ARCHIVE/BACKUP Tapepool Management

Invoke the interactive TAPEPOOL management functions.

VTOC Scan — Release Idle Space

Generate JCL to invoke the Idle Space Release function.

VTOC Scan — Disk to Disk Migration

Generate JCL to invoke the IMPLICIT MOVE function.

VTOC Scan — Intelligent PDS Compression

Generate JCL to invoke the IMPLICIT PDS Compress function.

VTOC Scan — Restart PDS Compression Process

Generate JCL to restart the PDS Compress RELOAD function.

VTOC Scan — Move data sets from Disk to Tape

Generate JCL to invoke the IMPLICIT Sequential Migration to Tape function.

Files/Parameter Library Allocations

Displays a panel showing the names of the data sets that will be used for CA Disk requests.

Syntax Check Member of PARMLIB

Utility function to cause CA Disk to syntax-check any member of the parmlib data set. This is normally used after a user has modified some member in parmlib.

Online Reporting

Generate online reports using DSCL selection language. Reports can also be submitted as batch jobs.

HEX Dump FILES

Generate online hexadecimal dump of subfile records.

Generate DSCL Commands

Generate JCL for batch job submission.

ISPF Return Function Restriction

The CA Disk ISPF support does not fully support the ISPF *return* or *jump* facility. Users can use the facility to go from CA Disk panels to non-CA Disk panels, but cannot specify multiple menu level numbers when entering CA Disk functions. For example, =8 and the <Enter key can be used to get to the CA Disk ISPF menu but = 4 cannot be used to get to a specific CA Disk ISPF function.

However, the ISPF command delimiter character can be used in place of the period. When the command delimiter character is a ";", entering

=8;4;2

will select the fourth option of the condensed menu, then the second option from the selection menu.

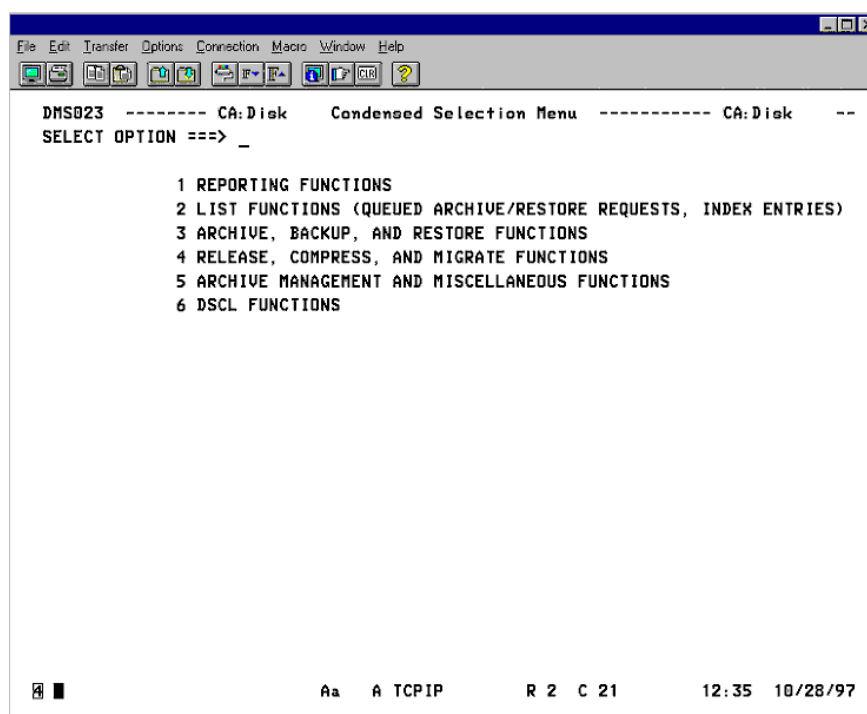
ISPF Panel Examples

Following are examples of CA Disk ISPF panels. This is not intended to be a complete set of examples or complete instructions on how to use the support. They are provided to give the user an idea of the function and format of CA Disk ISPF panels.

To determine exactly how to use the CA Disk ISPF interface, select the CA Disk option on your primary option menu, select the desired option, and press the <HELP> PF key.

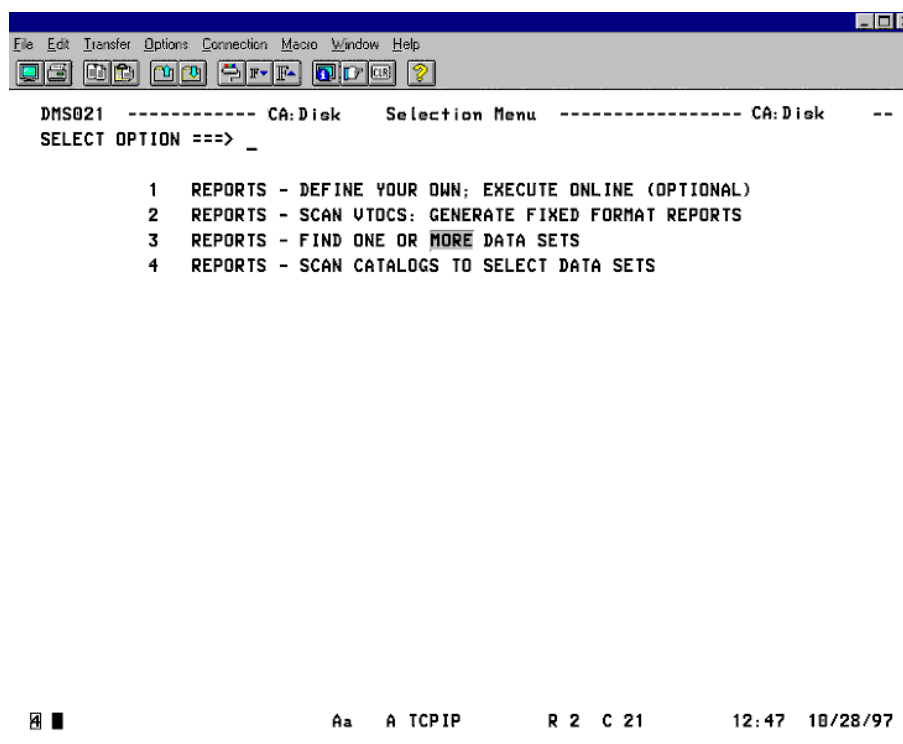
Condensed Selection Menu

Selecting the CA Disk option on the ISPF primary option menu enters this panel. It is displayed only when users have more than 15 CA Disk options available to them. This menu allows the user to select from various categories of CA Disk functions presented on the next menu level. The following illustration is a sample Condensed Selection Menu:



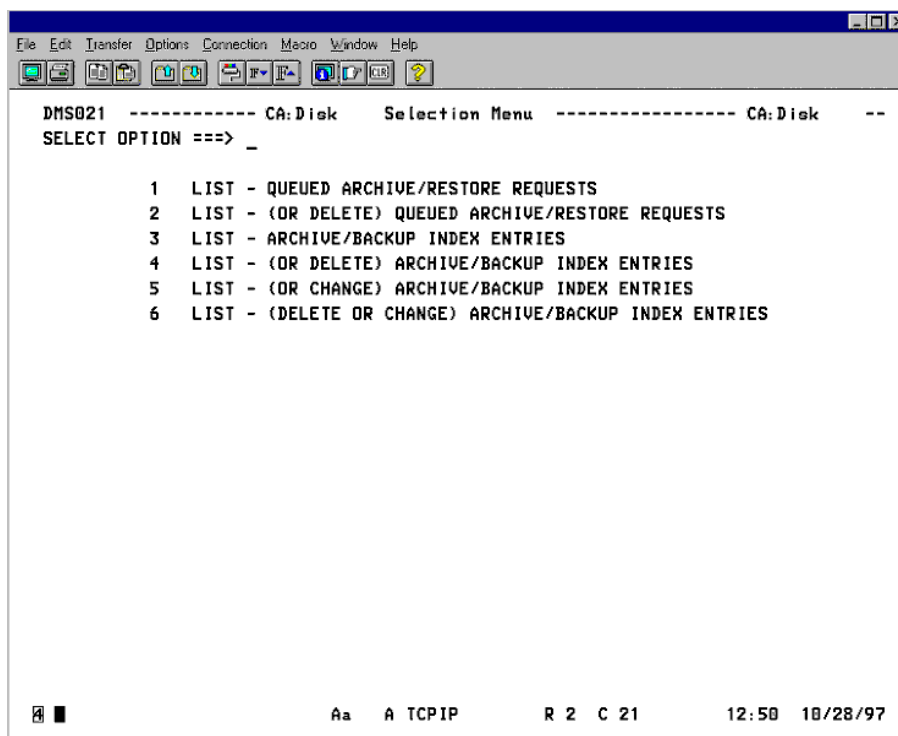
Selection Menus (Reports)

This panel is displayed from either the ISPF primary options menu or from the CA Disk CONDENSED MENU. It allows the user to pick the specific CA Disk ISPF interface function to use. The following illustration is a sample Selection Menus (Reports):



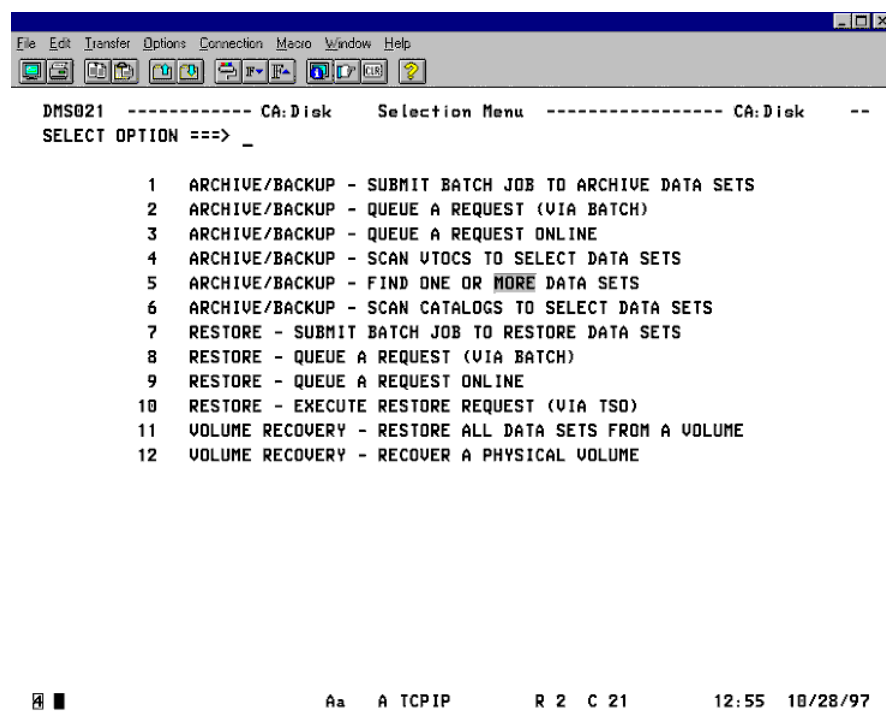
Selection Menu (List)

This panel is displayed from either the ISPF primary options menu or from the CA Disk CONDENSED MENU. It allows the user to pick the specific CA Disk ISPF interface function to use. The following illustration is a sample Selection Menus (List):



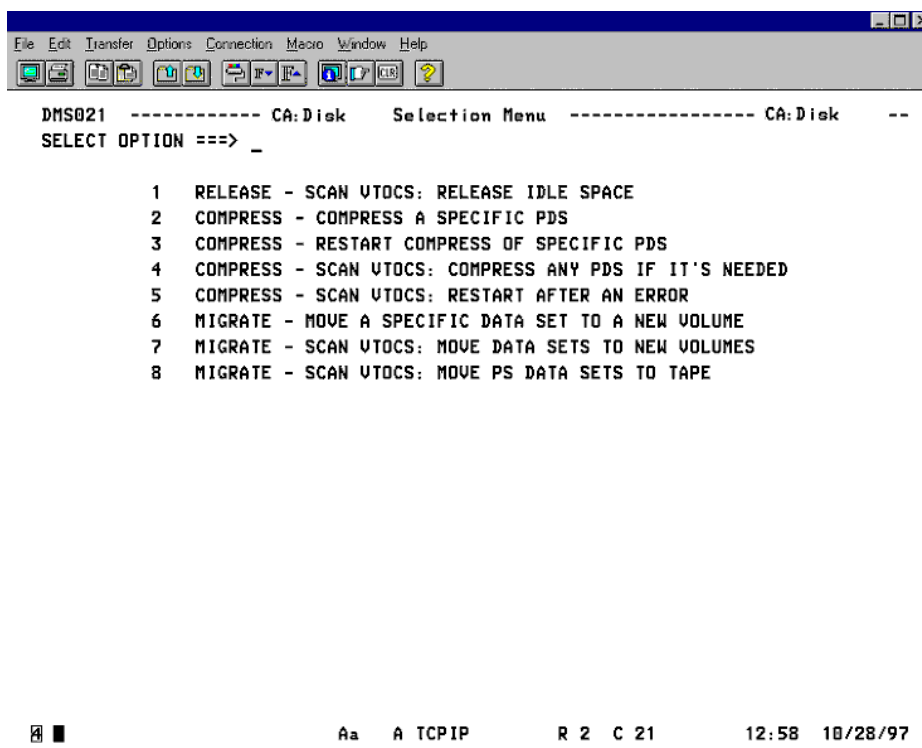
Selection Menu (Archive/Restore)

This panel is displayed from either the ISPF primary options menu or from the CA Disk CONDENSED MENU. It allows the user to pick the specific CA Disk ISPF interface function to use. The following illustration is a sample Selection Menu (Archive/Restore):



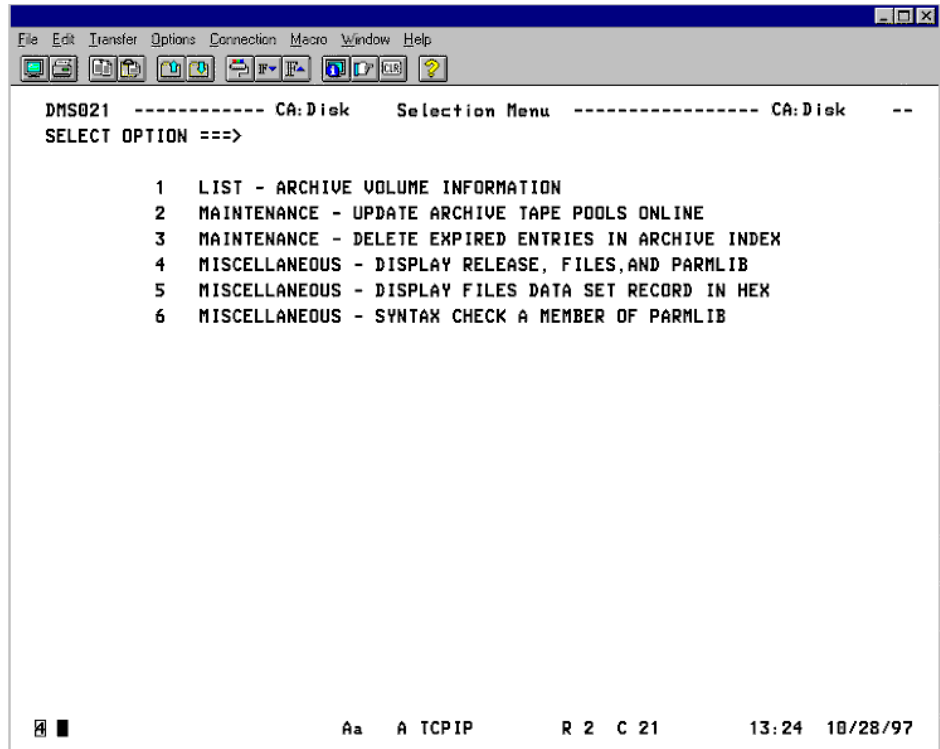
Selection Menu (Release/Compress/Migrate)

This panel is displayed from either the ISPF primary options menu or from the CA Disk CONDENSED MENU. It allows the user to pick the specific CA Disk ISPF interface function to use. The following illustration is a sample Selection Menu (Release/Compress/Migrate):



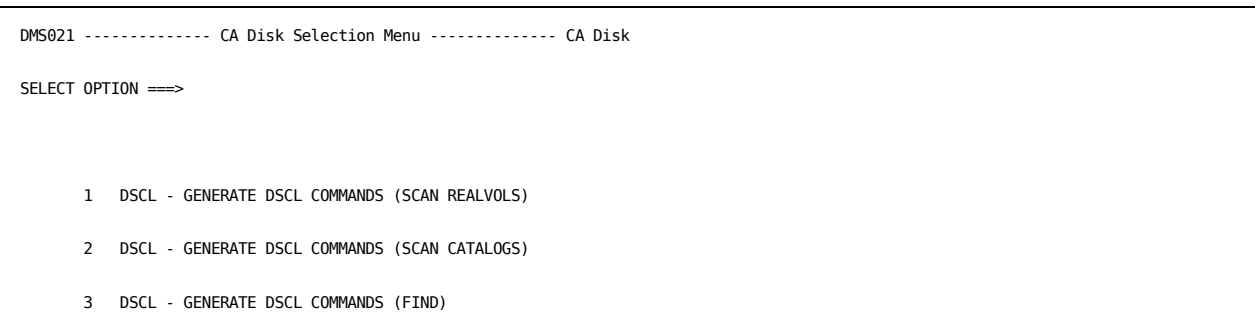
Archive Management and Miscellaneous Functions

This panel is displayed from either the ISPF primary options menu or from the CA Disk CONDENSED MENU. It allows the user to pick the specific CA Disk ISPF interface function to use. The following illustration is a sample Selection Menu (Archive Management):



DSCL Functions

This panel is displayed from either the ISPF primary options menu or from the CA Disk CONDENSED MENU. It allows the user to pick the specific CA Disk ISPF interface function to use. The following illustration is a sample Selection Menu (DSCL Functions):



Enter Request to Archive Data Set

This panel permits the user to enter a request to archive a specific data set. The user can either enter a specific data set name or a CA Disk pattern name. If a pattern name is supplied (as shown), the system catalog will be searched for names that match the pattern. These names will be presented on a selection screen and the user can then select the exact data set names to be archived (see the next panel example).

```

DMS001A ----- Enter Request to Archive Data Set ----- CA:Disk --
COMMAND ==> _

ISPF Library:
PROJECT ==>
LIBRARY ==>
TYPE ==>

Press END key to terminate processing.
Press ENTER to generate archive request.

Other data set, cluster name, or SAMS:Disk pattern name for select list:
Data Set Name ==> 'userid.DMS. /'
Volume Serial ==> (If not cataloged)
Expire Date ==>
Retention Period ==>
Delete ==>
Catalog Action ==>
Password ==> (Optional; VSAM cluster password only)

If desired, enter new name for data set in archives:
New Name ==>

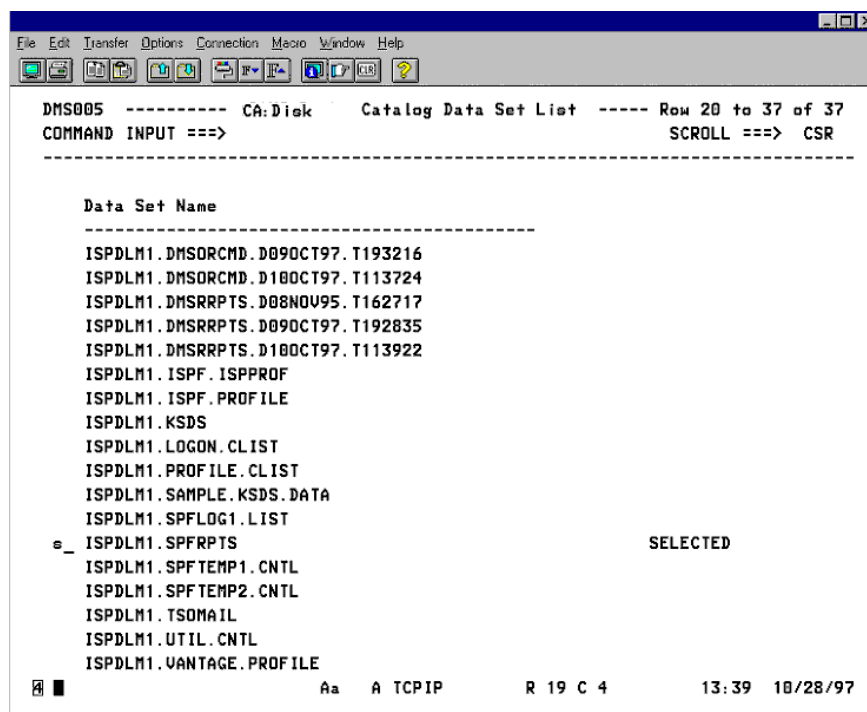
Jobcard information: (verify before proceeding)
==>
==>
==>
==>

Aa A TCP/IP R 2 C 15 13:33 10/28/97

```

CA Disk Catalog Data Set List

This panel is displayed when a pattern data set name is entered on an explicit data set processing screen. It is a list of data set names from the system catalog that match the specified pattern name. The user can enter an S by the name of the data set to be processed.



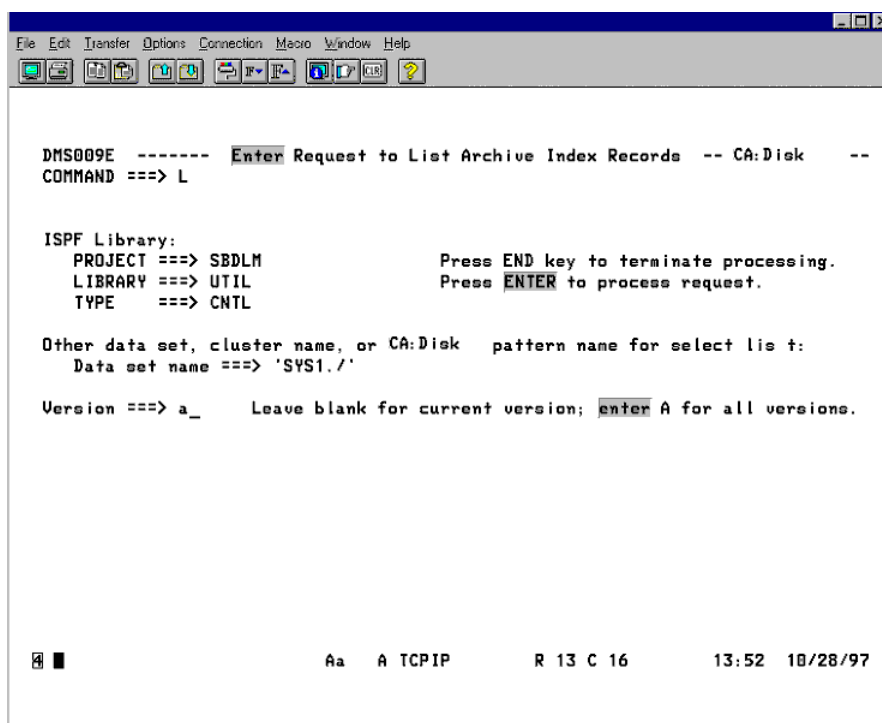
```
DMS005 ----- CA:Disk Catalog Data Set List ----- Row 20 to 37 of 37
COMMAND INPUT ==> SCROLL ==> CSR

-----
Data Set Name
-----
ISPDL1.DMSORCMD.D09OCT97.T193216
ISPDL1.DMSORCMD.D18OCT97.T113724
ISPDL1.DMSRRPTS.D08NOV95.T162717
ISPDL1.DMSRRPTS.D09OCT97.T192835
ISPDL1.DMSRRPTS.D18OCT97.T113922
ISPDL1.ISPF.ISPPROF
ISPDL1.ISPF.PROFILE
ISPDL1.KSDS
ISPDL1.LOGON.CLIST
ISPDL1.PROFILE.CLIST
ISPDL1.SAMPLE.KSDS.DATA
ISPDL1.SPFL0G1.LIST
S_ ISPDL1.SPFRRPTS                                SELECTED
ISPDL1.SPFTEMP1.CNTL
ISPDL1.SPFTEMP2.CNTL
ISPDL1.TSOMAIL
ISPDL1.UTIL.CNTL
ISPDL1.VANTAGE.PROFILE

Aa A TCPIP R 19 C 4 13:39 10/28/97
```


Enter Request to List/Process Archive Index

This panel allows the user to enter a complete or pattern data set name to be used to locate a record in the CA Disk archive index. If a pattern data set name is entered, a selection list of data sets will be presented and the user can select the data set to be listed, deleted or changed. If an explicit data set name is entered, the data set will be processed, based on the SELECT option specified.



The screenshot shows an ISPF panel titled "Enter Request to List Archive Index Records". The panel has a menu bar with options: File, Edit, Transfer, Options, Connection, Macro, Window, Help. Below the menu bar is a toolbar with icons for various functions. The main area of the panel contains the following text:

```
DMS009E ----- Enter Request to List Archive Index Records -- CA:Disk --  
COMMAND ==> L
```

ISPF Library:

```
PROJECT ==> SBDLM          Press END key to terminate processing.  
LIBRARY ==> UTIL          Press ENTER to process request.  
TYPE ==> CNTL
```

Other data set, cluster name, or CA:Disk pattern name for select list:

```
Data set name ==> 'SYS1.'
```

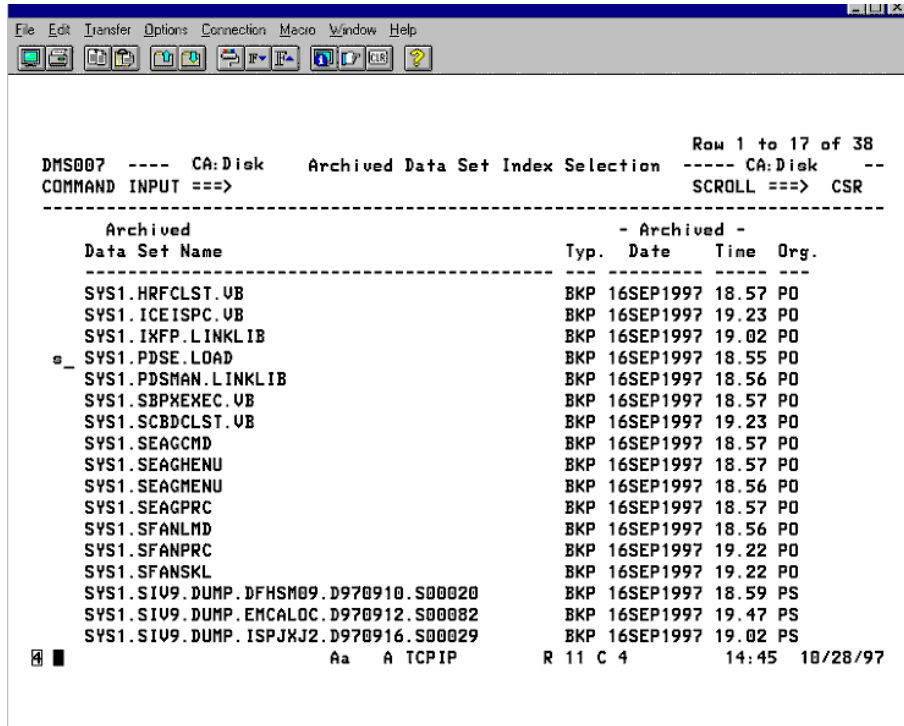
Version ==> a_ Leave blank for current version; enter A for all versions.

At the bottom of the panel, there is a status bar with the following information:

```
  Aa  A TCPIP  R 13 C 16  13:52  10/28/97
```

Archived Data Set Index Selection

This panel is displayed when the user enters a pattern data set name in response to an explicit restore request or an archive data set list request. It is a list of data set names from the archive index that match the specified pattern. The user can specify a ' next to any of the listed data sets to view additional information about the index record. By specifying an F, CA Disk displays the index records flag field information.



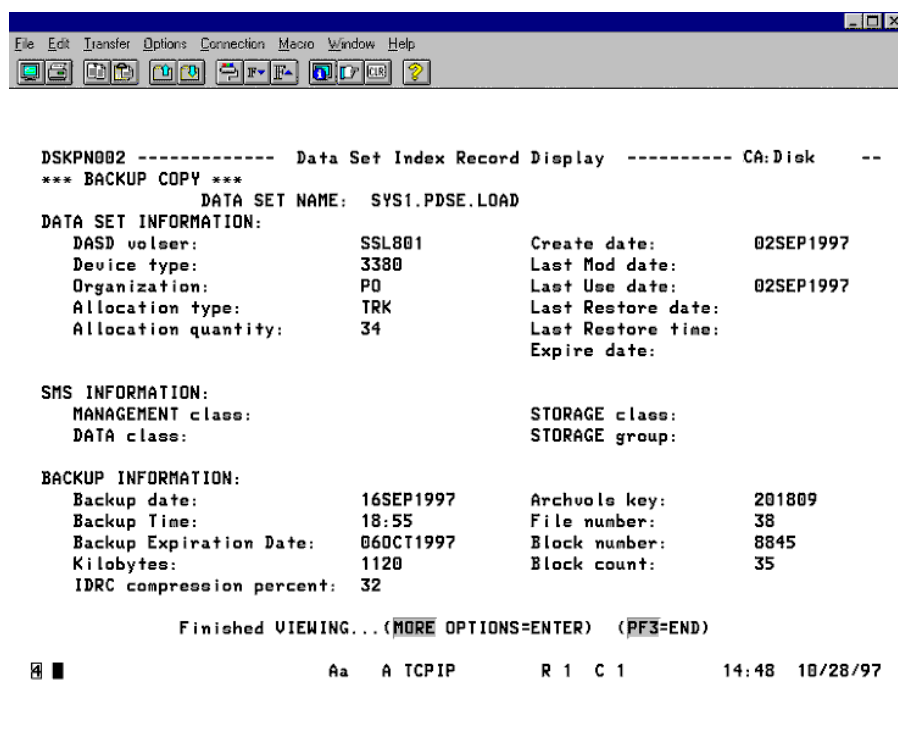
Archived Data Set Index Selection				
Archived Data Set Name	Typ.	Date	Time	Org.
SYS1.HRFCLST.VB	BKP	16SEP1997	18.57	PO
SYS1.ICEISPC.VB	BKP	16SEP1997	19.23	PO
SYS1.IXFP.LINKLIB	BKP	16SEP1997	19.02	PO
SYS1.PDSE.LOAD	BKP	16SEP1997	18.55	PO
SYS1.PDSMAN.LINKLIB	BKP	16SEP1997	18.56	PO
SYS1.SBPXEXEC.VB	BKP	16SEP1997	18.57	PO
SYS1.SCBDCST.VB	BKP	16SEP1997	19.23	PO
SYS1.SEAGCMD	BKP	16SEP1997	18.57	PO
SYS1.SEAGHENU	BKP	16SEP1997	18.57	PO
SYS1.SEAGMENU	BKP	16SEP1997	18.56	PO
SYS1.SEAGPRC	BKP	16SEP1997	18.57	PO
SYS1.SFANLMD	BKP	16SEP1997	18.56	PO
SYS1.SFANPRC	BKP	16SEP1997	19.22	PO
SYS1.SFANSKL	BKP	16SEP1997	19.22	PO
SYS1.SIU9.DUMP.DFHSM09.D970910.S00020	BKP	16SEP1997	18.59	PS
SYS1.SIU9.DUMP.EMCALOC.D970912.S00082	BKP	16SEP1997	19.47	PS
SYS1.SIU9.DUMP.ISPJXJ2.D970916.S00029	BKP	16SEP1997	19.02	PS

Row 1 to 17 of 38
 COMMAND INPUT ==> CSR
 Aa A TCP/IP R 11 C 4 14:45 10/28/97

When you specify an 'S', Data Set Index Record Display

Data Set Index Record Display

This display panel contains information about a data set in the CA Disk archives. It is displayed as a part of the data set index list function. The following illustration is a sample Data Set Index Record Display:



From this panel, you have 2 options:

You can press <ENTER> to view the flag field settings for an index record.

You can press PF3 to select another record in the list.

By pressing <ENTER>, DSNINDEX Flag Field Pop-Up Panel is displayed.

Options Menu Pop-Up Panel

This Pop-Up panel allows the user to display DSNINDEX Flag Field settings. By pressing <ENTER>, the following panel is displayed. The following illustration is a sample DSNINDEX Flag Field Pop-Up Panel:

```

DSKPN002 ----- Data Set Index Record Display ----- CA:Disk  --
*** BACKUP COPY ***
      DATA SET NAME:  SYS1.PDSE.LOAD
DATA SET INFORMATION:
  DASD volser:         |_>>>>>  OPTIONS <<<<<<| ate:          02SEP1997
  Device type:         |-----| date:
  Organization:         | -> VIEW Status Flags   | date:          02SEP1997
  Allocation type:      | (Press ENTER key)     | tore date:
  Allocation quantity:  | -> SELECT or VIEW a new | tore time:
                       | DSNINDEX record       | ate:
                       | (Press PF3 key)       |
SMS INFORMATION:
  MANAGEMENT class:    |-----| class:
  DATA class:         |-----| STORAGE group:

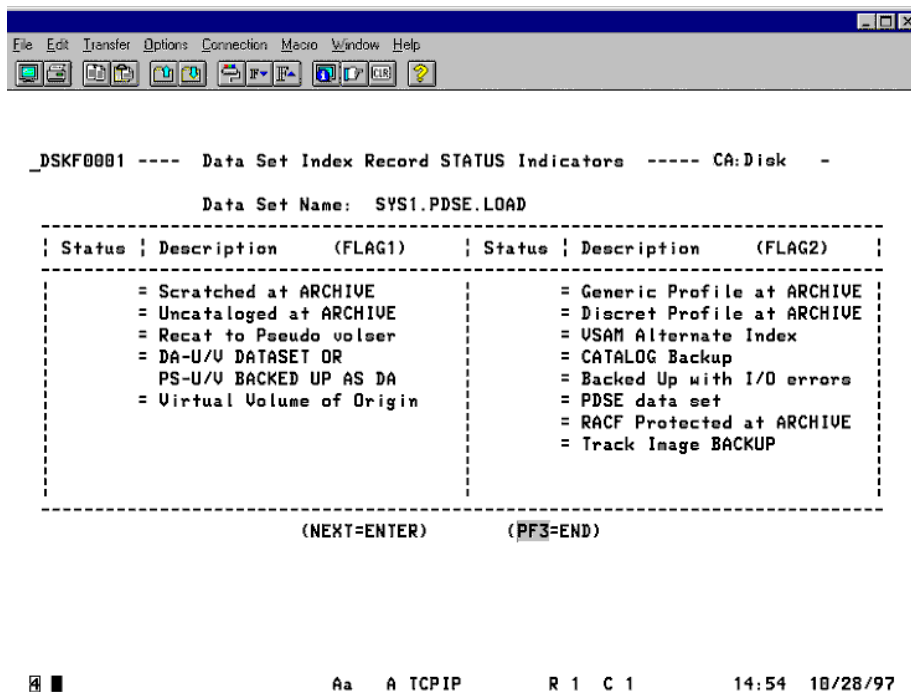
BACKUP INFORMATION:
  Backup date:          16SEP1997      Archive key:      201809
  Backup Time:          18:55          File number:      38
  Backup Expiration Date: 06OCT1997    Block number:     8845
  Kilobytes:           1120           Block count:      35
  IDRC compression percent: 32

      Finished VIEWING...(MORE OPTIONS=ENTER) (PF3=END)

Aa  A TCP/IP      R 5  C 27      14:52  10/28/97
  
```

DSNINDEX Flag Field Status Indicators — Panel 1

This is the first of three STATUS panels available while viewing DSNINDEX Flag Field settings. This is also the panel displayed when you specify an 'F' on Archived Data Set Index Selection.



```

_DSKF0001 ---- Data Set Index Record STATUS Indicators ----- CA:Disk -

Data Set Name: SYS1.PDSE.LOAD

| Status | Description (FLAG1) | Status | Description (FLAG2) |
|-----|-----|-----|-----|
| | = Scratched at ARCHIVE | | = Generic Profile at ARCHIVE |
| | = Uncataloged at ARCHIVE | | = Discret Profile at ARCHIVE |
| | = Recat to Pseudo volser | | = USAM Alternate Index |
| | = DA-U/V DATASET OR | | = CATALOG Backup |
| | PS-U/V BACKED UP AS DA | | = Backed Up with I/O errors |
| | = Virtual Volume of Origin | | = PDSE data set |
| | | | = RACF Protected at ARCHIVE |
| | | | = Track Image BACKUP |
|-----|-----|-----|-----|
| (NEXT=ENTER) | (PF3=END) |

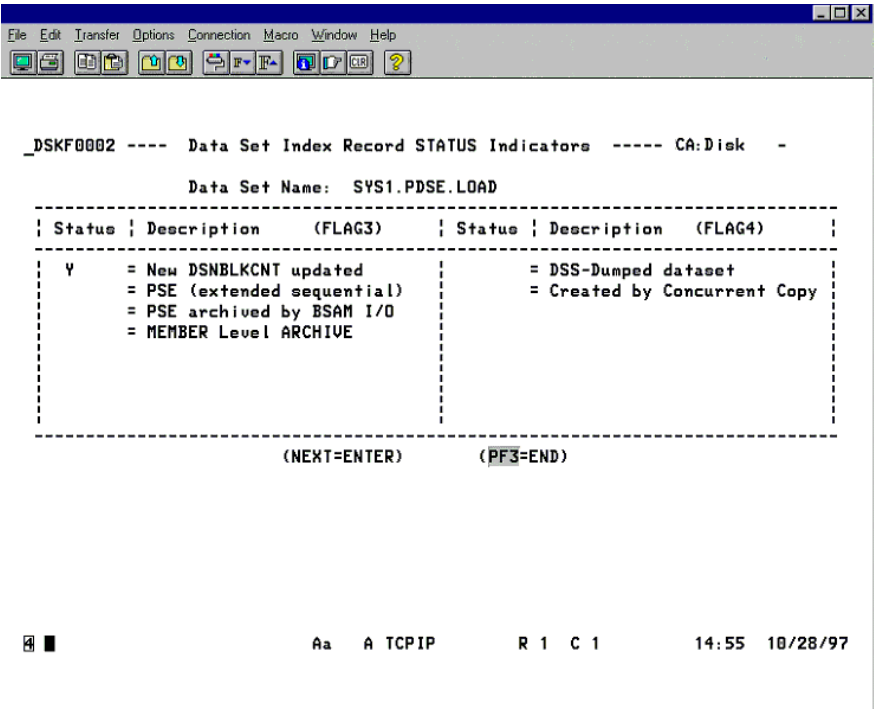
Aa A TCPIP R 1 C 1 14:54 10/28/97

```

By pressing <ENTER>, Flag is displayed.

DSNINDEX Flag Field Status Indicators – Panel 2

The following illustration is a sample Flag Field Status Indicators - Panel 2



By pressing <ENTER>, Flag Field Status Indicators ? Panel 3 appears.

DSNINDEX Flag Field Status Indicators – Panel 3

The following illustration is a sample DSNINDEX Flag Field Status Indicators – Panel 3:

_DSKF0003 ---- Data Set Index Record STATUS Indicators ---- CA:Disk --

Data Set Name: SYS1.PDSE.LOAD

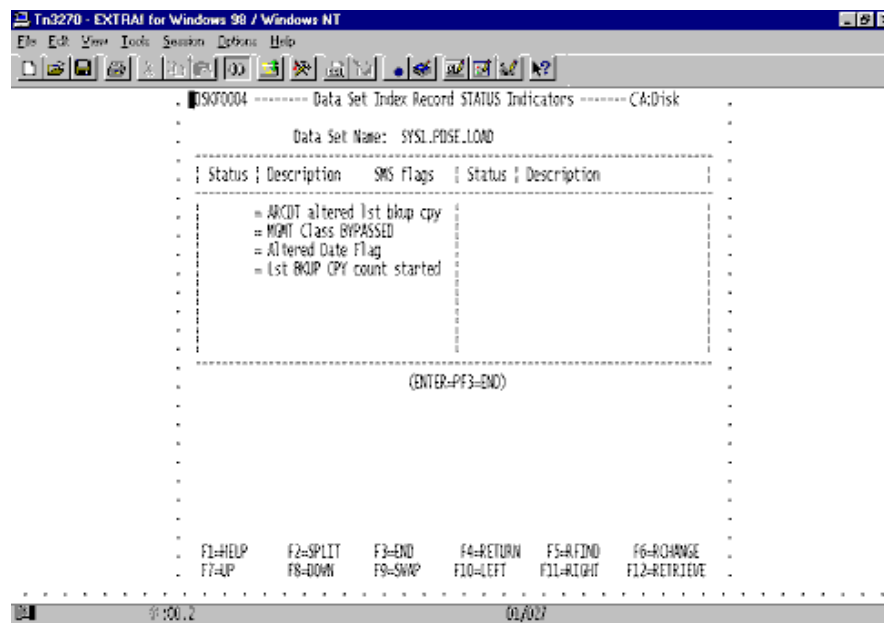
Status	Description	SMS flags	Status	Description
	= ARCDT altered 1st bkup cpy			
	= MGMT Class BYPASSED			
	= Altered Date Flag			
	= Lst BKUP CPY count started			

(ENTER=PF3=END)

Aa A TCP/IP R 1 C 1 14:56 10/28/97

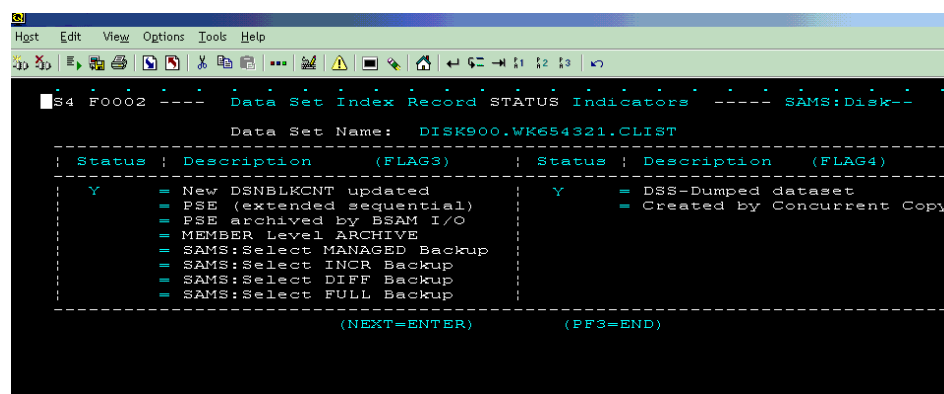
DSNINDEX Flag Field Status Indicators – Panel 4

The following illustration is a sample DSNINDEX Flag Field Status Indicators – Panel 4:



DSNINDEX Flag Field Status Indicators – Panel 5

The following illustration is a sample DSNINDEX Flag Field Status Indicators – Panel 5:



Compress PDS Data Set

This panel allows the user to specify parameters to generate JCL to compress one or more partitioned data sets. After this panel is entered, another will be displayed that will request that more command information be entered.

The screenshot shows an ISPF panel titled "Compress PDS Data Set". The panel has a menu bar at the top with options: File, Edit, Transfer, Options, Connection, Macro, Window, Help. Below the menu bar is a toolbar with icons for various functions. The main area of the panel displays the following information:

_DSKF0003 ---- Data Set Index Record STATUS Indicators ----- CA:Disk --

Data Set Name: SYS1.PDSE.LOAD

Status	Description	SMS flag	Status	Description
	= ARCDT altered 1st bkup cpy			
	= MGMT Class BYPASSED			
	= Altered Date Flag			
	= 1st BKUP CPY count started			

(ENTER=PF3=END)

At the bottom of the panel, there is a status bar with the following information: Aa A TCP/IP R 1 C 1 14:56 10/28/97

Generate JCL — Panel 2

This panel is displayed as a second input panel for explicit PDS Compress command generation. The user indicates additional information relating to the request to compress a data set.

The screenshot shows a terminal window with a menu bar (File, Edit, Transfer, Options, Connection, Macro, Window, Help) and a toolbar. The main text area contains the following content:

```
DMS030C ----- Generate JCL to Compress PDS ----- CA:Disk  --
COMMAND ==> _
Press END key to terminate request; press ENTER to generate command.

Data set name:

Threshold criteria for selecting data sets:
  Max. used  ==>      Maximum percentage of allocated space used.
  Min. used  ==>      Minimum percentage of allocated space used.
  Extents    ==>      Data set has more than nn extents.
  Max. dir.   ==>      Maximum percentage of directory space in use.
  Min. dir.   ==>      Minimum percentage of directory space in use.

Specify Y to select any of the following options:
  USE        ==>      Prevent reallocation of PDS.
  SHRINK     ==>      Allow primary allocation to reduce in size.
  CONVERT    ==>      Allocation type may be changed from CYL to TRK.
  PASSWORD   ==>      Process password-protected data sets.
  EMPTY      ==>      Process empty (no members) data sets.
```

At the bottom of the panel, there is a status line: `Aa A TCP/IP R 2 C 15 15:48 10/28/97`.

Generate JCL - Panel 3

This panel allows the user to specify allocation parameters to be used in building JCL and control statements to be used in the CA Disk implicit reporting function.

The screenshot shows an ISPF panel titled 'DMS030F ----- Generate JCL to Compress PDS ----- CA:Disk --'. The panel has a menu bar with 'File', 'Edit', 'Transfer', 'Options', 'Connection', 'Macro', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons. The main text area contains the following prompts:

```
DMS030F ----- Generate JCL to Compress PDS ----- CA:Disk --
COMMAND ==>

Press END key to terminate request; press ENTER to generate command.

Specify an allocation type to use the percentages as absolute values:
Space      ==> _      Allocation type (CYL, TRK, CYLCTG, etc.)

Specify values to be used to calculate new data set size:
Pri. space ==>      nnn percent primary space to be in use.
Extent size ==>     Extent size is to be nnn percent of total size.
Directory  ==>      nnn percent directory space to be in use.
```

At the bottom of the panel, there is a status bar with the following information: a small icon, 'Aa', 'A TCP/IP', 'R 8 C 22', '16:00', and '10/28/97'.

Overview of ISPF Online Reporting Facility

This facility provides the user with an automated means of generating reports that are tailored to each individual's needs. It also has the capability of remembering the data set names that it processes. This feature can be used to generate utility commands (either for CA Disk or some other utility) to perform some other function. We will now quickly go over the general flow of the facility, and later on will go into more detail on how all of the panels interact.

There are two very basic *paths* in the online reporting environment:

- Defining reports to CA Disk.
- Generating the actual reports after specifying the volumes and selection criteria to be used for the report(s).

The first path is executed by selecting options 0 through 4 on the primary option menu for online reporting (DMS042A), and the second path is selected with options 5 through 9. You must define your report before you can exercise the second path.

As you use this function of CA Disk, keep in mind that no matter where you are in processing, you can enter the command CANCEL to return to the primary option panel of the online reporting facility. This allows a graceful escape regardless of where you are in processing.

All of the report definitions that you generate must be saved in a special report definition data set. If the data set has not already been generated, or if you want to create your own, select option 0 from the main menu (DMS042A). This will take you to an ISPF panel where allocation information for the data set is specified. After this information is entered, CA Disk will allocate and format the data set. You are now ready to define reports.

Options 1 through 4 are used to create, modify, list and delete report definitions. Selecting option 1 allows you to define a new report. This will take you through a whole series of ISPF panels that prompt you for information about the report. After you finally make it to the end (defining grand total fields is the last step), you will be returned to the primary option menu. From here you are free to define more reports, modify the one you just created, or enter the report generation phase.

The ISPF online reports can include the SMS constructs and attributes. See the ISPF Custom Reports in the *Systems Guide* for information about adding these report fields.

Report generation begins by selecting option 5 (even though option 9, Generate Commands, is under this category on the menu, it's not really used to generate reports). By selecting this option, you are telling CA Disk to scan online volumes.

The first panel in report generation prompts you for the volumes you want to process. Here you can specify a number of volumes or volume patterns to scan. You should specify only those volumes that you are sure contain the data sets you are interested in looking at, in order to conserve computer resources.

From the volume selection panels you move to some selection test panels. These panels allow you restrict the reports to a subset of the data sets that are actually on the volume(s) being scanned. After entering selection tests, you are taken to a panel that has a list of all the reports that have been defined. You can select one or more reports to be included in the run.

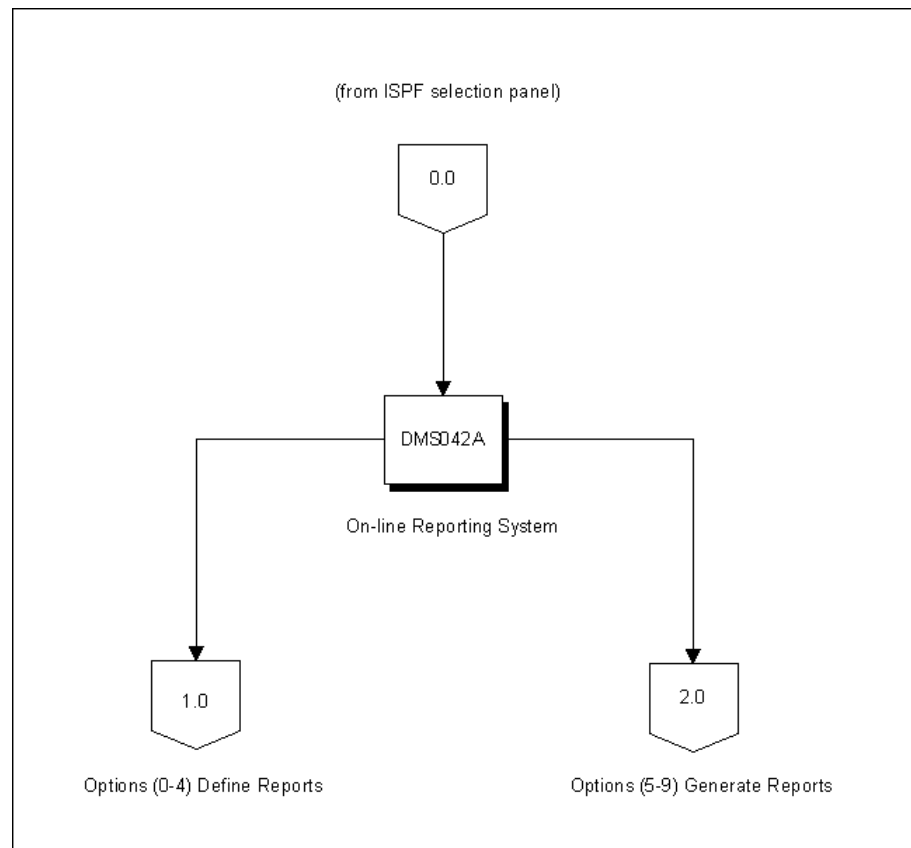
After selecting the report(s), you are taken to a panel that asks you to *confirm report requests*. The commands listed on the screen are the same ones you would enter to execute the reports in batch. At this point you can choose to run the reports in foreground (assuming you are authorized to), submit a batch job to generate them, or simply exit the reporting phase. If you elect to execute the job in batch, you will be prompted for the proper JCL and the job will be submitted.

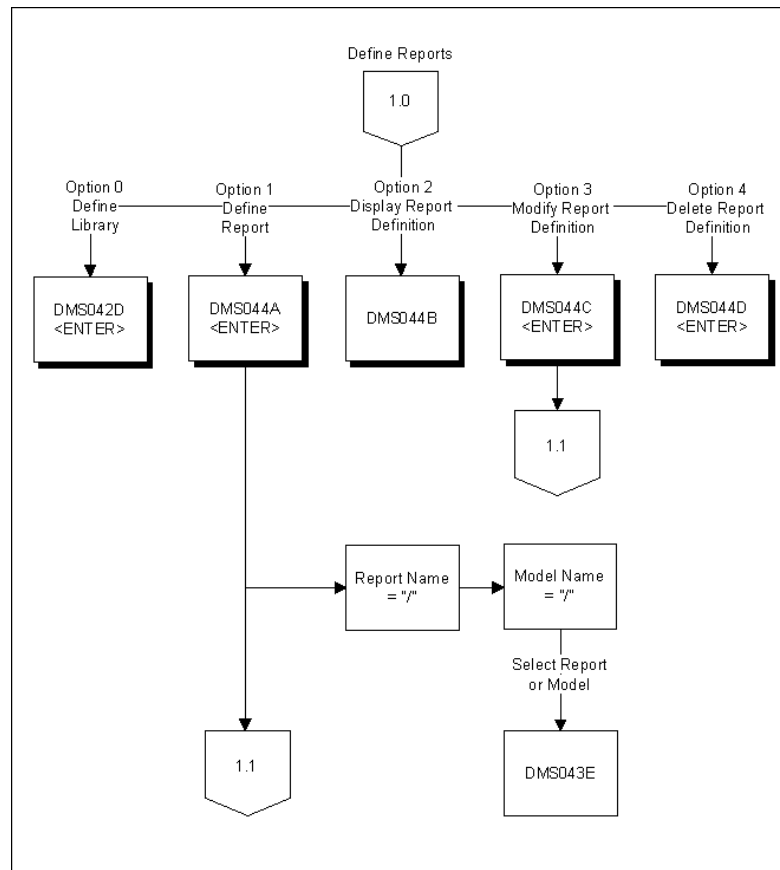
If you decided to run the reports in foreground, the screen will clear and a message will be displayed as each volume is scanned. After the report is formatted, it will be presented to you in ISPF browse mode (at this point it's merely a temporary data set). When you are finished looking at the report, press the <END> key. This will take you to a report disposition screen. From here you can choose from several options: rebrowse the report, save the report to a permanent data set, print the report with a batch job, generate utility commands, and/or save the report commands for later use. Pressing the <END> key at this point will return you to the top menu of the online reporting facility.

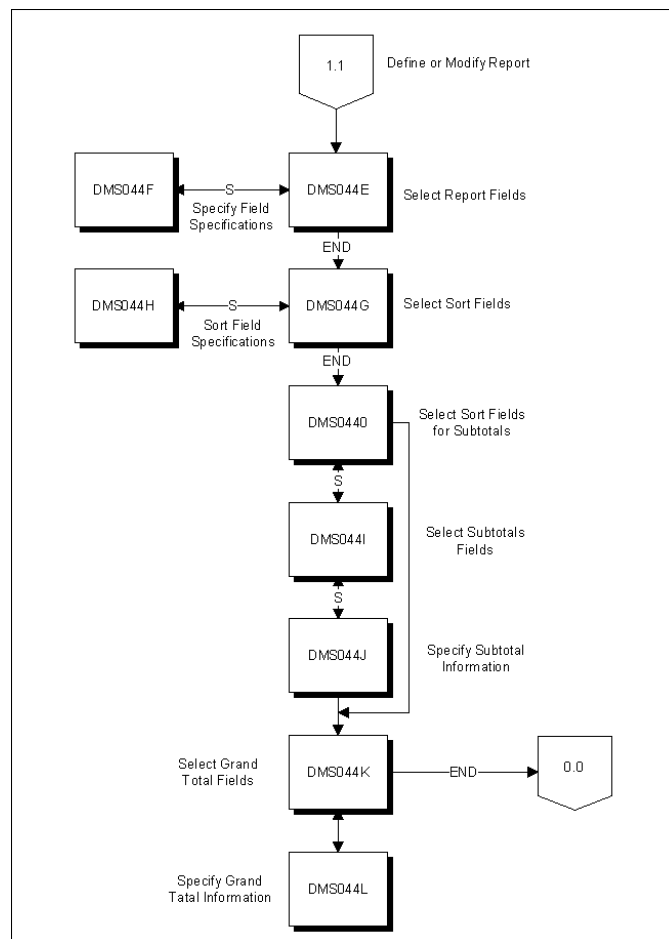
Hierarchy of ISPF Online Reporting Panels

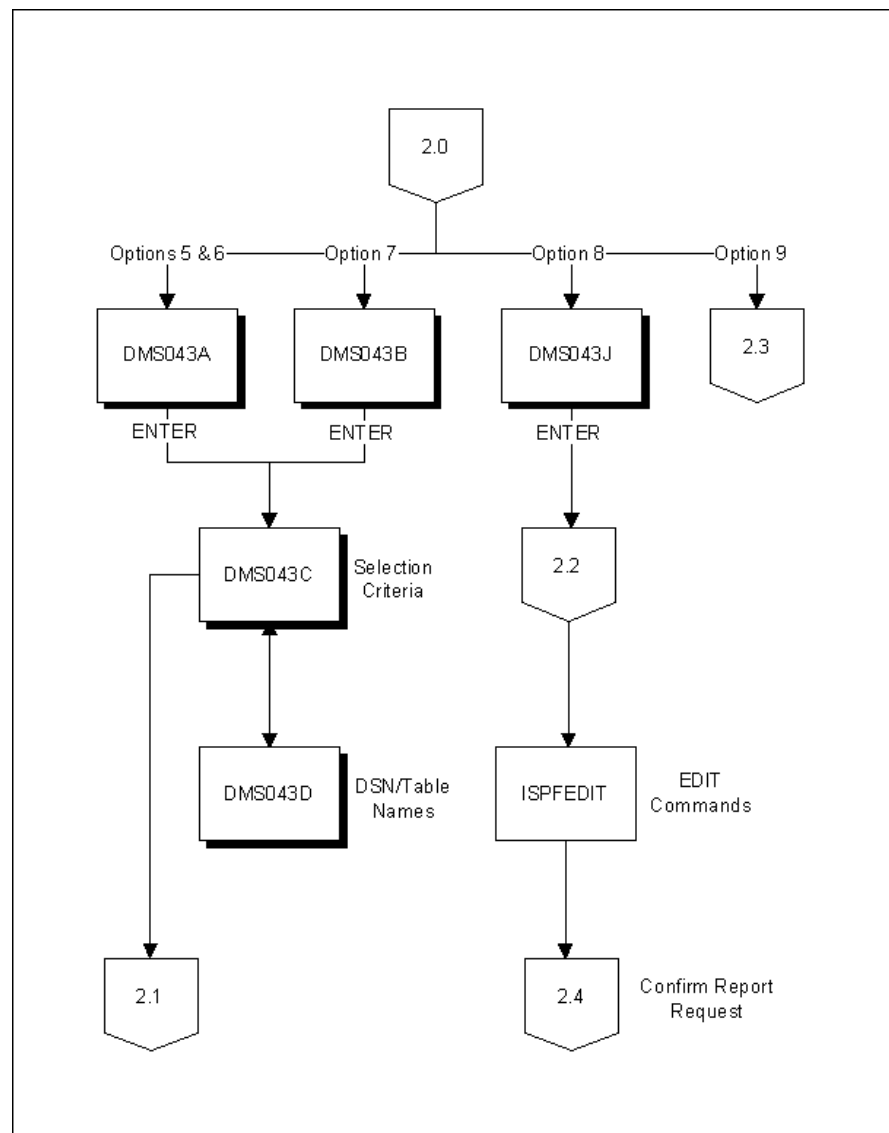
We will now take an in-depth look at the functions provided by the online reporting facility. First we present a hierarchical overview of the interaction of the various ISPF panels used in this facility. Then we will review most of the individual screens and give tips for usage. The screens will be presented in the order that a first-time user is most likely to see them:

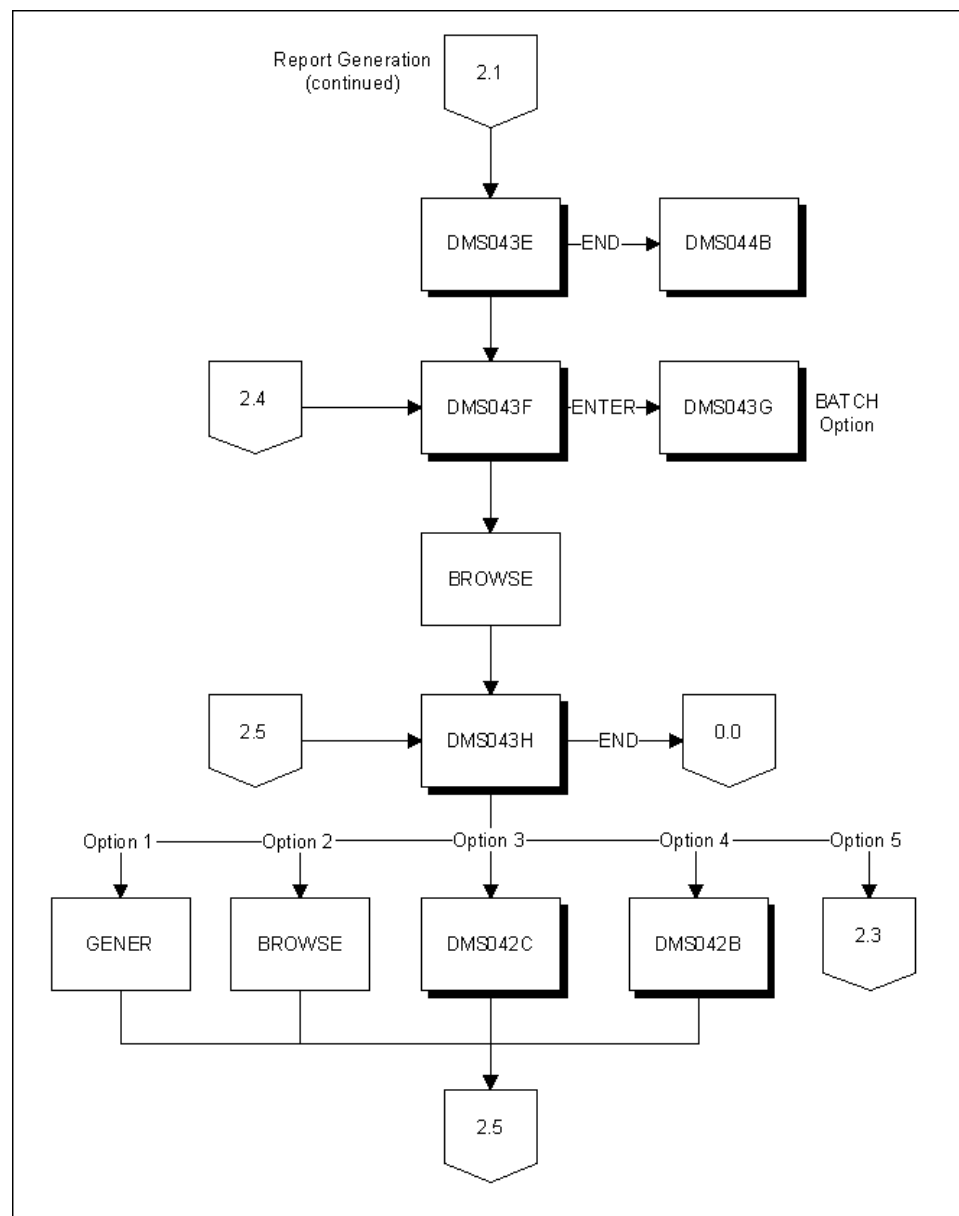
- Allocation of the report definition data set
- Defining an online report
- Executing the report and performing disposition processing

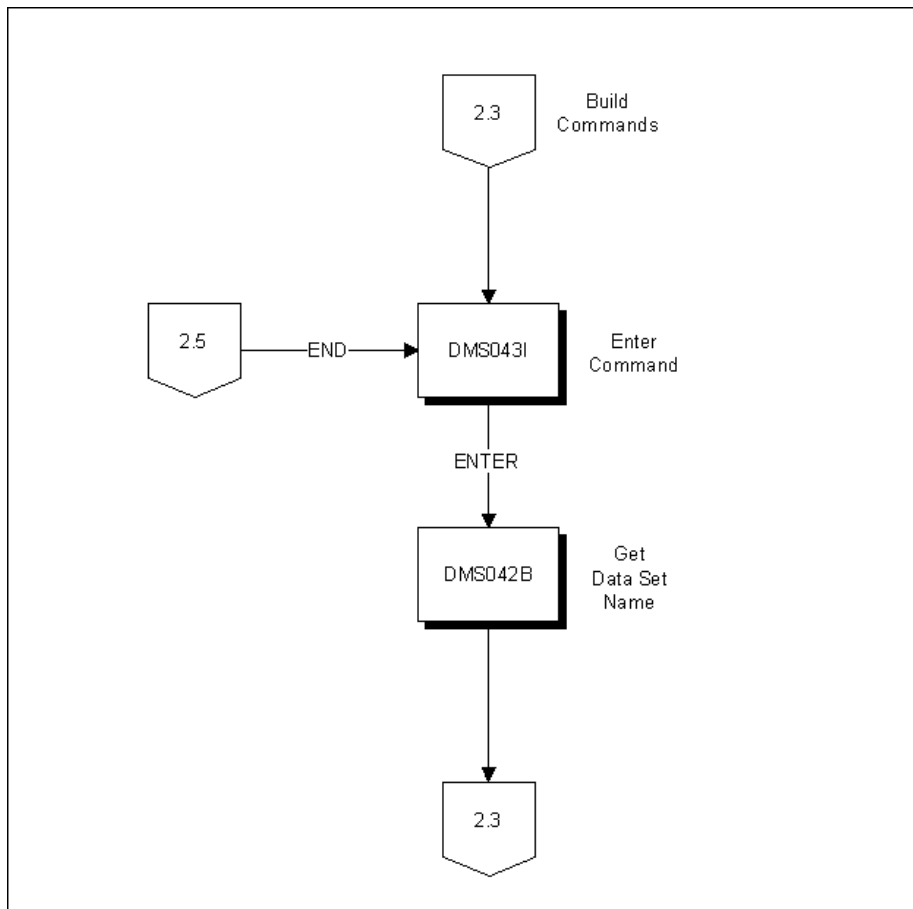












ISPF Online Reports

This is the initial panel presented when the online reporting facility is selected from the CA Disk selection menu. The following illustration is a sample Online Reporting System:

```

DMS042A  ----- CA:Disk   Online Reporting System ----- CA:Disk   --
SELECT OPTION ==>

Report Definition Library ==> SPFRPTS

----- Define Report -----
| 0 - Allocate new library |
| 1 - Define new report   |
| 2 - Display existing report |
| 3 - Modify existing report |
| 4 - Delete existing report |
-----

Enter report name, or /
for a list ==>
(options 2 thru 4 )

Note: Report(s) must be defined
before report generation options
are used.

----- Generate Report -----
| 5 - Scan real DASD volumes |
| 6 - Scan MSS volumes       |
| 7 - Scan MSS volume group  |
| 8 - Enter "Expert" mode    |
| 9 - Generate commands      |
-----

Save DSNs for later processing?
==> (yes or no)

Add COMPONENTS on REPORT command?
==> (yes or no)

Press ENTER to continue processing; press END to terminate processing.

Aa A TCP/IP R 2 C 21 17:37 10/28/97

```

This is a very *busy* panel that governs the path to be followed: report definition or report generation. The one field that is common to both facilities is the REPORT DEFINITION LIBRARY name. Be sure it is correct before selecting an option, because it cannot be changed on any other panel. Let's first concentrate on the report definition function.

The report definition phase appears in the box on the left of the screen, and it consists of five different options. Option 0 is used to allocate a new report definition library. This option is seldom if ever used by most users. Typically, most users will share a common library built by a data administrator or group leader. If you select this option and allocate a new library, be sure to put the new library's name in the REPORT DEFINITION LIBRARY field. The allocation panel is displayed on the following page.

Allocate Library for Report Definition

The information requested on this panel should be self-explanatory. After filling in the information, press the <ENTER> key to allocate the data set. If you decide to terminate the allocation, press <END>. The following illustration is a sample Allocation Library for Report Definition:

```

DMS042D ----- Allocate Library for Report Definition ----- CA:Disk
COMMAND ==>

Data set name:

Volume serial      ==>          (blank for default volume) *
Generic unit       ==> SYSDA    (generic group name or unit address) *
Space units        ==> TRKS     (BLKs, TRKs or CYLS)
Primary quantity   ==> 5        (in above units)
Secondary quantity ==> 5        (in above units)
Directory blocks   ==> 5
Blocksize          ==> 8000     (blocksize must be a multiple of 200)

( * Only one of these fields may be specified.)

Press ENTER to allocate data set or END to cancel allocation request.

Aa A TCP/IP R 2 C 15 17:42 10/28/97

```

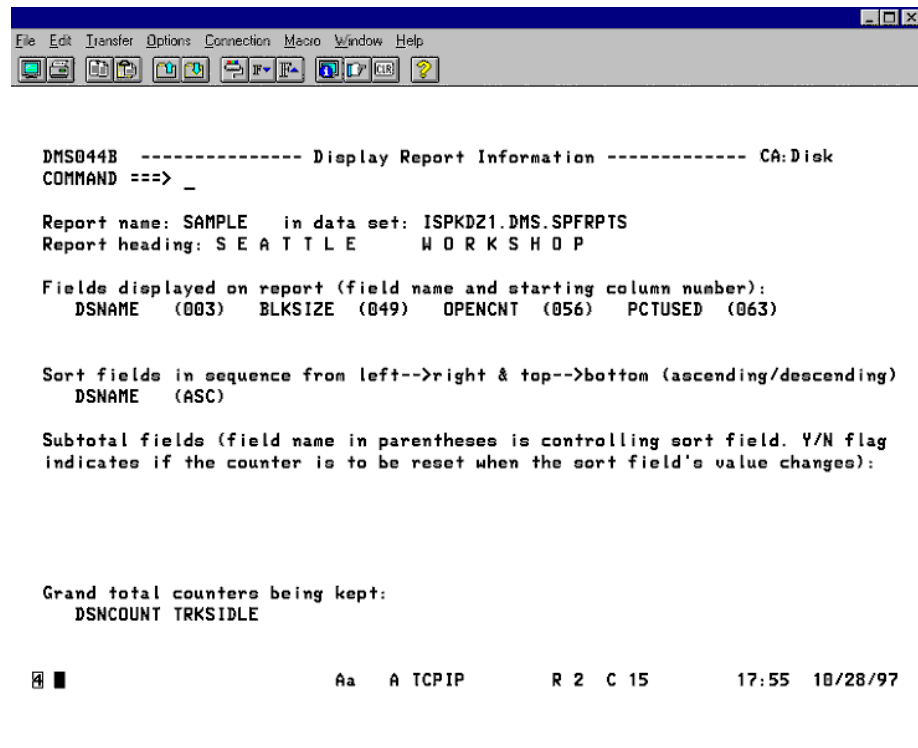
Now let's go back to the primary options menu. Before looking at option 1, which defines a new report, let's look at the other three options (2, 3 and 4). Option 2 is used to display a report's definition, and the panel that is used to display this information will be shown shortly. Option 3 is used to modify an existing report definition, and will look almost identical to option 1 (define new report) except that the first panel will be different, and on all subsequent panels that have field lists, the fields will already be marked as SELECTED if previously defined on the report. If option 4 is selected, you will be presented with another panel to confirm the delete request (you can press <ENTER> to delete the report or enter *CANCEL* in the command line to bypass it). Each of these options (2, 3 and 4) make use of the field on the panel labeled *ENTER REPORT NAME, OR / FOR A LIST*. At this point you must enter an exact report name that was previously defined. If you don't remember the name, enter a / and you will be presented with panel DMS043E, which is a scrollable table of all report names found in the definition library. The panel will look something like this:

Select Desired Report

If more than one page of report names has been presented, you can use the scroll keys to view the list. When you find the report name you are looking for, enter an *S* beside it. CA Disk will continue processing the option you specified on the reports primary option panel (DMS042A) as if you entered the report name from that panel. If you decide not to select one of the report names you must enter *CANCEL* on the command line and then press the <ENTER> key.

Display Report Information

If you select option 2 from the primary option panel, this is the panel that will be displayed. It gives a quick summary of the pertinent information in the report, such as display fields, sort fields, subtotal fields and grand total fields. the following illustration is a sample Display Report Information:



Specify New Report Information

If you select option 1 from the primary option menu, you will begin a new report definition with this panel. The following illustration is a sample Specify New Report Information:

```

DMS044A ----- Specify New Report Information ----- CA:Disk
SELECT OPTION ==> 1

      1 - Report on data sets (format-1 DSCBs)
      2 - Report on volume summary information only.

=====

Report's name ==> IDLTRK (6 characters)
Descriptive name ==> DISPLAY IDLE SPACE IN DATA SETS
Report heading ==> I D L E   T R A C K S _
Model (optional) ==>

Press ENTER to continue processing or END to return to primary options menu.

Aa  A TCPIP      R 13 C 48      18:00  10/28/97

```

There are two options that you can select from, and it is very important that you select the proper one. Option 1 is the most frequently used, and it causes CA Disk to report on data set names found on a DASD volume. Option 2 is used to provide summary information on DASD volumes, and therefore will only produce one line of output for each DASD volume processed.

The REPORT'S NAME field is used to give any name you desire to the report. Remember this name -- you will need it when you want to run the report later. Enter any description you want in the DESCRIPTIVE NAME field to help remember the purpose of the report. This field does not show up on the report itself, but is displayed alongside the report name whenever you ask for a list of report names defined in the report definition library. The REPORT HEADING field allows you to specify the heading that will appear when the report is generated.

The MODEL parameter is optional. This field would be used if you know another report exists that is already defined that is very similar to the new one you want to produce. Enter the name of the existing report, or enter a "/" to display a list of all the previously defined reports. When a model is used, the remaining portion of the report generation phase will actually look exactly like you had specified the modify option for an existing report from the primary option panel.

Select Report Field

This panel displays a scrollable list of all report fields that can be selected for reporting. In the columns next to the field name are FLD LTH and HDG LTH. These fields show you the number of characters required to display the actual field's data on the report, and also how many characters in width the field's heading requires. The larger of the two values for any given field is the number of bytes that it takes to display the field on the report.

Note: If you select the RACFALL, RACFDIS or RACFGEN fields for your report, you will not be able to run the report online. The selection line for these fields warns you of this fact. You will still be able to run the report in batch. However, if you have modified your TSO session to run APF-authorized (TSO normally does not run this way), you will be able to run your report online.

The remainder of the panels in the report definition phase uses an interactive approach to selecting fields. While each one will vary slightly, the overall processing tries to be consistent. As you select each field from the panel, you will be taken to another panel that prompts for more specific information about a field's selection. From this panel you can type in the requested information and press the <ENTER> key, or you can press <END> to cancel the request. Either response will return you to the field selection screen. You can then select another field, or you can press to proceed to the next level of report definition.

There are two line commands that can be entered from the field selection panel:

- An S to select a report field
- A D to delete it from the report (if it was previously selected)

To use these commands, you can use the next line (carriage return) key on the keyboard to position the cursor next to the field you want to select (or delete). Then enter the S or D and press the <ENTER> key. Your request will then be processed. If the field you desire is not on the screen, you can use the scroll keys to position the display to the proper page that contains the field name.

From the command line you can enter:

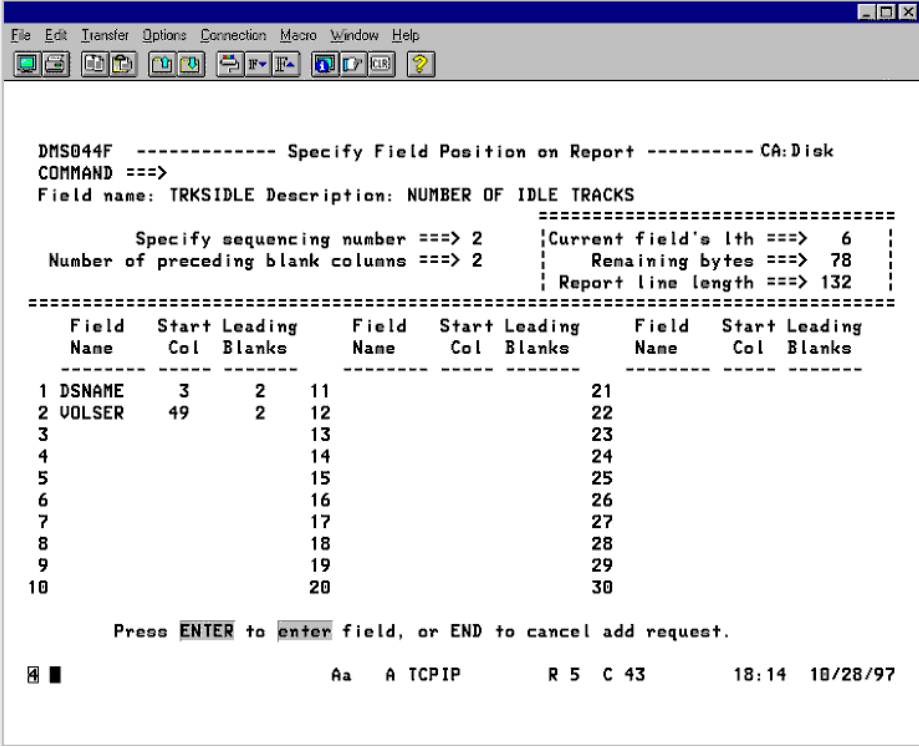
- S <fieldname>—to select a field for reporting without having to move the cursor to it,
- D <fieldname>—to delete the report field,
- L <string>—to locate or position the display to the first field name in the table that begins with the same character string,
- SHOW—to show the fields that have been selected to this point, or
- CANCEL—to cancel the report definition and return you to the online reports primary option menu.

Note: The fields shown in the previous examples with < > should be entered without the brackets.

Pressing the <END> key at this point signals CA Disk that you have completed the report field selection phase and are ready to proceed to the sort field selection phase. Before showing that panel, let's look at the panel displayed as a result of making a field selection or entering *SHOW* on the command line.

Specify Field Position on Report

The following illustration is a sample Specify Field Position on Report:



DMS044F ----- Specify Field Position on Report ----- CA:Disk
 COMMAND ==>
 Field name: TRKSIDE Description: NUMBER OF IDLE TRACKS

Specify sequencing number ==> 2 Current field's lth ==> 6
 Number of preceding blank columns ==> 2 Remaining bytes ==> 78
 Report line length ==> 132

Field Name	Start Col	Leading Blanks	Field Name	Start Col	Leading Blanks	Field Name	Start Col	Leading Blanks
1 DSNAME	3	2	11			21		
2 VOLSER	49	2	12			22		
3			13			23		
4			14			24		
5			15			25		
6			16			26		
7			17			27		
8			18			28		
9			19			29		
10			20			30		

Press ENTER to enter field, or END to cancel add request.

Aa A TCP/IP R 5 C 43 18:14 10/28/97

There are two field positioning values that you can enter:

- The field's sequencing number
- The number of preceding blanks

The sequencing number automatically defaults to the next available position on the report. If you want to insert the field in front of some other field already selected on the report, specify that field's number as the sequencing field. It will be inserted at that point and all of the other fields will be pushed down one position. If a field you previously selected is not at the position you desire, simply delete it from the report field selection screen and then reselect it.

The number of preceding blanks field allows you to select the number of blanks that will appear in front of this field on the report.

There are also several fields that are for display only. The field name and description are carried over from the field selection panel to remind you of the field you're currently working with. The numbers appearing in the box to the right are useful in formatting the report. The number shown for the field's length is the number of characters the field requires to print on the report, and is the larger of field width and heading width that was displayed on the previous panel. The remaining bytes field tells you how many bytes are still available on the report line. This does not include the current field you are defining, since the program has no way of knowing how many leading blanks you will specify (or if you will end up not selecting the report field).

Pressing the <ENTER> key adds the field to the report. If you decide that you don't want to add the field, press <END>. Either way you return to the report field selection panel. Suppose you just added a field and want to see what the new report looks like without selecting another report field. Remember, you can use the *SHOW* command from that screen to redisplay the field positioning screen without performing any field updates.

Select Sort Fields

Now that you have selected all of the fields to be printed on the report, it is time to specify the field(s) you want to sort on. The sort field selection process resembles the report field selection process. From this panel you will select, one at a time, the fields that are to be used for sorting. As you make each selection, you will be taken to panel DMS044H, which allows you to specify the details of the sort. The following illustration is a sample Select Sort Fields:

```

DMS044G ----- Select Sort Fields ----- Row 1 to 8 of 8
COMMAND ==> SCROLL ==> CSR

|== Enter S to select a sort field or D to delete field from sort criteria.
|      (Press END to continue on to subtotals selection screen.)
|
| Field Sort
| U Name Width Expanded Description Comment
|-----|-----|-----|-----|-----|
| DSNAME 44 DATA SET NAME (AS FOUND IN FORMAT 1 DSCB)
| e NODE1 8 FIRST LEVEL NODE OF DSNAME
| NODE2 8 SECOND LEVEL NODE OF DSNAME
| NODE3 8 THIRD LEVEL NODE OF DSNAME
| NODE4 8 FOURTH LEVEL NODE OF DSNAME
| NODE5 8 FIFTH LEVEL NODE OF DSNAME
| TRKSIDLE 2 NUMBER OF IDLE TRACKS
| VOLSER 6 DASD VOLUME DATASET FOUND ON
|***** Bottom of data *****

Aa A TCP/IP R 10 C 4 18:20 10/28/97

```

You will notice this field list is much shorter than the one presented for the field selection process. This was done intentionally so that you won't have to page through a lot of unnecessary fields. There are a few fields that CA Disk always displays on this screen, such as the nodes (index levels) of data set names. The only other fields that are displayed are those that were selected for the report. It's probably a safe assumption that if the field is not desired on the report, there isn't much reason to sort on it either.

The commands for this panel are identical to the report field selection panel, so we will just review them at this point. There are two line commands that can be used:

- An *S* to select a sort field, or
- A *D* to delete it from the sort field selection list
- From the command line you can enter:
 - *S* <fieldname>—to select a field for sorting without having to move the cursor to it,
 - *D* <fieldname>—to delete the sort field,
 - *L* <string>—to locate or position the display to the first field name in the table that begins with the same character string,
- *SHOW* to show the fields that have been selected to this point, or
- *CANCEL*, which cancels the report definition and returns you to the online reports primary option menu.

Note: The fields shown in the previous examples with < > should be entered without the brackets.

When you have finished selecting all of your sort fields press the <ENTER> key. This will take you to the subtotal selection screen where you can define the numeric fields that can be subtotaled.

Specify Sort Field Information

On this panel, you enter more information about the particular sort field you just selected. Again, this panel is very similar to the report field selection panel. The following illustration is a sample Specify Sort Field Information:

```

DMS044H ----- Specify Sort Field Information ----- CA:Disk
COMMAND ==>

Sort field name: NODE1   Description: FIRST LEVEL NODE OF DSNAME

Specify sequencing number ==> 1   (1=highest 9=lowest)
Ascending/Descending ==> ASC   (A=ascending D=descending)
Is this field considered a control break? ==> NO   (YES or NO)

If this field is a control break, answer the following:
Number of print lines to skip before next detail line ==> 0   (E=eject)

=====
|Previously selected fields: bytes remaining ==> 68   MAX length ==> 68   |
=====
1----- 2----- 3----- 4----- 5-----
6----- 7----- 8----- 9-----

Press ENTER to enter field, or END to cancel select request.

Aa A TCP/IP R 6 C 49 6:45 10/29/97

```

You can override the sequencing number if you want to change the sequence of the sort fields. You can also specify if the sort for this field is to be in ascending or descending sequence.

You are also asked if this field is a control break. If you are only sorting on this field and do not need to do any subtotalling or line spacing, then answer *no* to this question. If you answer *yes*, you should specify the number of lines to skip before printing the next detail line. For instance, if you are sorting on volume number and want to have the data set names for each new volume begin on a new page, specify control break as *yes* and the lines to skip as an *E* (page Eject). If you are sorting on data set name, you would probably respond *no* to the control break prompt, since it doesn't make much sense to do subtotalling at the individual data set name level.

Press the <ENTER> key to add the field as a sort criteria. Press the <END> key to terminate the request. Either key will return you to the sort field selection panel.

Select Any Sort Fields Where Subtotaling is Desired

Selecting the fields to perform subtotaling on is probably the most complex portion of the report definition process. This phase involves a three-tier hierarchy of panels, as opposed to the two-tier hierarchies found in the report field and sort field selection processes. The first panel shown here displays a list of the sort fields that you selected. You can select one or more of these fields as levels where subtotaling will be performed.

```

DMS0440 -- Select Any Sort Fields Where Subtotaling is Desired Row 1 to 3 of 3
COMMAND ==> SCROLL ==> CSR

|=== Enter S to start subtotal field selection process for this sort field
|      (or to modify a previously defined group of subtotals).
|
|      *** Press END key to continue on to grand total selections. ***
|
|      Name of                               Subtotals
|      U Sort Field Expanded Description      Defined?
|-----|-----|
s  NODE1   FIRST LEVEL NODE OF DSNAME
   TRKSIDLE NUMBER OF IDLE TRACKS
   VOLSER   DASD VOLUME DATASET FOUND ON
***** Bottom of data *****
  
```

To really understand the process, we need to look at a sample problem. Suppose your installation standard is that the third level node (or index) of every data set name must be the user ID. We want to produce a list of all data sets that belong to each user in volume sequence. We also need to show the total number of tracks allocated on each volume for the user, as well as total overall tracks allocated and tracks used for each user. Each user's data sets should begin on a new page. How should we define this report?

First of all, we would select NODE3 as the first-level sort criteria and define it as a control break with lines to skip equal to "E" (page eject). We would select VOLSER as the second-level sort field and specify it as a control break with skip lines equal to 2. We would probably also specify DSNAME as the third sort criteria, just so the data set names would show up in alphabetical order for each user (although this wouldn't be necessary). When we reach this panel (DMS0440), the three fields would show up in alphabetical order: DSNAME, NODE3, and VOLSER. Since there is no requirement for any totaling at the data set name level, it will be ignored.

The important idea to keep in mind is that the sort fields actually form a hierarchy of control levels in the reporting process. In this example we have the following levels:

- NODE1 (userid)
- TRKSIDLE
- VOLSER

This is important, because when the report is being generated, a comparison will be performed on the data to determine if any of the sort fields have changed value, from the highest level down. If a difference is found, all subtotaling from that level down will be performed and printed in reverse order (lowest level up to the level where the change was detected). Each subtotal is indented on the report according to the level of the sort field. This is mentioned here because it can be very confusing defining your subtotal fields if you don't understand how the levels work. In addition, the printed report can also be confusing if you use the standard default for subtotal print lines, as we'll see shortly.

Back to our example, we would first select NODE3. This can be done by either placing the cursor next to NODE3 on the display and keying an *S*, or by entering *S NODE3* on the command line. All we are doing at this point is telling CA Disk that we want to perform some type of subtotaling at the NODE3 (in our case, the user ID) level. This will take us to another panel (DMS044I) where you can specify the fields you want to total at this level. After you select the subtotal fields at this level, you will eventually be returned to this panel to make another selection. Next you would select the VOLSER field, which would take you back to the subtotal field selection screen. Although this panel is identical to the one shown at the NODE3 level, you are now selecting fields that will be totaled based on a change in volume within each user ID.

When you have finished specifying all subtotaling fields you desire, press the key at this panel to continue in the report definition process. If you change your mind about any of the subtotals being kept, you can reselect the sort field on this panel to modify your previous selections. If you decide that you want to terminate the report definition process, enter the command *CANCEL* in the command input area.

Select Subtotal Fields

This is the panel where you select which fields to subtotal. The sort field name is displayed at the top of the panel to show the level where the subtotalling is performed (labeled CONTROL FIELD on the panel). You will now interact with the field selection process much the same way as the report field and sort field selection panels did. From this screen you select each field, one at a time, to do subtotalling on. You will be guided to panel DMS044J to enter more information about the counter. When you finish selecting the field(s) you want totaled, press the <ENTER> key. This will return you to panel DMS044O, where you can select other sort fields on which to total.

Again, the commands for this panel are nearly identical to the report field selection panel. There are two line commands that can be used:

- An *S* to select a subtotal field, or
- A *D* to delete it from the subtotal field selection list

From the command line you can enter:

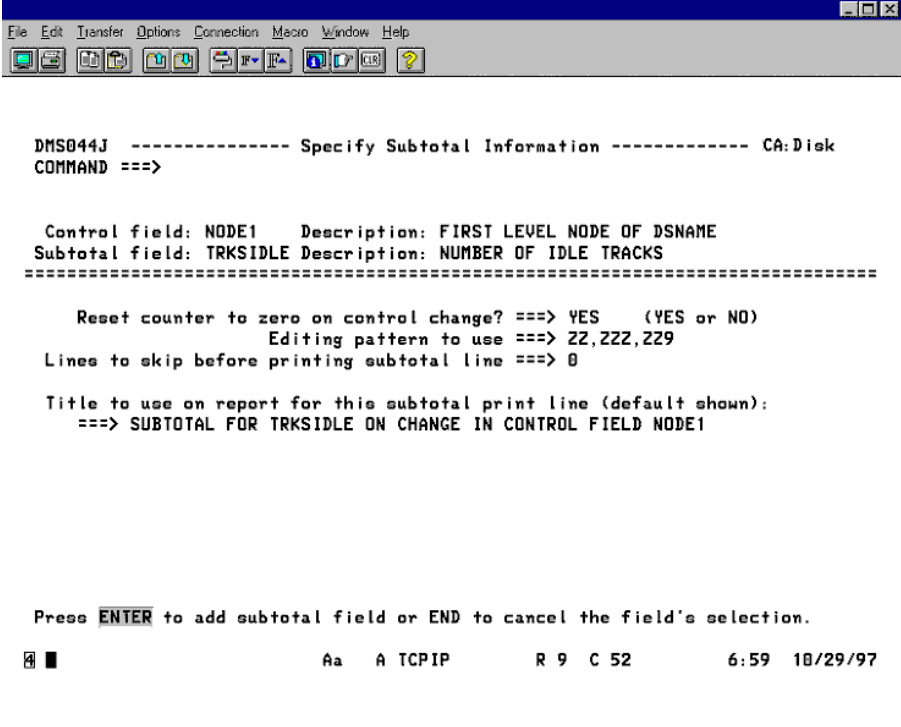
- *S* <fieldname>—to select a field for subtotalling without having to move the cursor to it,
- *D* <fieldname>—to delete the subtotal field,
- *L* <string>—to locate or position the display to the first field name in the table that begins with the same character string, or
- *CANCEL*, which cancels the report definition and returns you to the online reports primary option menu.

Note: The fields shown in the previous examples with < > should be entered without the brackets.

There is one important piece of information to keep in mind: the order in which you select the fields is the order in which they will be printed on the output report. Why is this important? Because the line spacing that you will specify on the next panel will influence the printing of subsequent subtotals. For instance, if you want all of the subtotals at a given control level to be printed on a separate page, then the first subtotal field you select must be specified with a line spacing value of *E*.

Specify Subtotal Information

On this panel, you enter information that controls the printing of the subtotal on the output report. You will notice that both the control field name and the subtotal field name have been carried over to the top of this screen for your convenience. The following illustration is a sample Specify Subtotal Information panel:



The screenshot shows a mainframe terminal window titled "Specify Subtotal Information". The window has a menu bar with "File", "Edit", "Transfer", "Options", "Connection", "Macro", "Window", and "Help". Below the menu bar is a toolbar with various icons. The main area of the window displays the following text:

```

DMS044J ----- Specify Subtotal Information ----- CA:Disk
COMMAND ==>

Control field: NODE1   Description: FIRST LEVEL NODE OF DSNAME
Subtotal field: TRKSIDLE Description: NUMBER OF IDLE TRACKS
=====

Reset counter to zero on control change? ==> YES   (YES or NO)
Editing pattern to use ==> ZZ,ZZZ,ZZ9
Lines to skip before printing subtotal line ==> 0

Title to use on report for this subtotal print line (default shown):
==> SUBTOTAL FOR TRKSIDLE ON CHANGE IN CONTROL FIELD NODE1

Press ENTER to add subtotal field or END to cancel the field's selection.

Aa  A TCP/IP      R 9  C 52      6:59  10/29/97

```

The RESET COUNTER field asks if you want the subtotal counter to be reset to zero when a control change is detected. The answer to this question should usually be *yes*. If you specify *no*, the value that is printed on the report will be a running total of the counter and as such will be incremented continually throughout the report. If you use the *no* option, you should probably include some reference to it being a running total in the title line that you supply.

The EDITING PATTERN TO USE field allows you to tailor the format used when printing the numeric value on the report. In most instances you can accept the default that is shown on the screen, so don't be concerned with this field if you don't understand how editing patterns work. If you are familiar with the editing patterns available in COBOL, this will be easy to recognize (although it really only provides a subset of COBOL's flexibility). The Zs in the pattern indicate suppression of leading zeros. This means if the digit being printed is a leading zero, print a blank instead of 0. The commas are used to help make the output more readable when printing large values, such as 21,882,041. If you don't want the commas to appear in the output, remove them from the edit pattern. The final character, a 9, is used to start what's known as the significance indicator. What this really does is cause the digit to be printed even if it is a leading zero.

If you want to print a blank value instead of a 0 when a counter contains zero, replace the 9 with a Z in the editing pattern. On the other hand, if you want leading zeros to be printed, you can replace the Zs in the pattern with 9s.

The LINES TO SKIP value allows you to specify the number of print lines to skip before printing this subtotal value. The default value of zero will cause the subtotal line to be printed immediately after the preceding print line. You can specify a value from 0 to 9, or an "E" to cause a page eject before printing the value.

The TITLE line is shown with the default that would be printed on the report. You should notice that both the control break field and the counter are mentioned in the default line. Although this is adequate in some reports, it is much nicer to supply a more meaningful title. In the example mentioned earlier, we were sorting by NODE3 primary and VOLSER secondary. We were doing subtotalling of tracks allocated at both levels, and also tracks used at the NODE3 level. If we were now defining the TRKS counter at the NODE3 level, the default title would show: SUBTOTAL FOR TRKS ON CHANGE IN CONTROL FIELD NODE3. A much more meaningful title is: TOTAL TRACKS ALLOCATED TO THIS USER ON ALL VOLUMES. By the same token, when defining the title for TRKS at the VOLSER level, it would be better to override the default with something like: TOTAL NUMBER OF TRACKS ALLOCATED TO THIS USER ON THIS VOLUME.

When you have completed entering the information on the panel, press the <ENTER> key to record the information. If you decide that you don't want to do subtotalling on the field, you can press <END>. In either case you will be returned to panel DMS0441.

Select Grand Total Fields

The grand total field selection process is identical to the subtotals process, so we won't go into the details again. Basically, you select the grand total fields one at a time and are taken to panel DMS044L to specify printing information for each field.

Keep in mind that the order you select the fields in is the order in which they will be printed on your report. When you have completed this panel and press <ENTER>, your report will be saved in the report definition library and you will be returned to the online reports primary options menu (DMS042A).

Specify Grand Total Information

Online Reporting System (Option 5)

At this point, you should have a report defined. You can now execute the report in either an online or batch mode. The following illustration is a sample Online Reporting System (Option 5) panel:

```

DMS042A ----- CA:Disk   Online Reporting System ----- CA:Disk   --
SELECT OPTION ==>

Report Definition Library ==> SPFRPTS

----- Define Report -----
| 0 - Allocate new library |
| 1 - Define new report   |
| 2 - Display existing report |
| 3 - Modify existing report |
| 4 - Delete existing report |
|-----|
Enter report name, or /
for a list ==>
(options 2 thru 4 )

Note: Report(s) must be defined
before report generation options
are used.

----- Generate Report -----
| 5 - Scan real DASD volumes |
| 6 - Scan MSS volumes      |
| 7 - Scan MSS volume group |
| 8 - Enter "Expert" mode   |
| 9 - Generate commands     |
|-----|

Save DSNs for later processing?
==> (yes or no)

Add COMPONENTS on REPORT command?
==> (yes or no)

Press ENTER to continue processing; press END to terminate processing.

Aa A TCPIP R 2 C 21 17:37 18/28/97

```

Since expert mode (option 8) and command generation (option 9) are more advanced options, we will leave them until later. After we have gone through the full report generation cycle, it will be much easier to explain these two functions. The one input field we will be concerned with now is the one labeled *SAVE DSNAMES FOR LATER PROCESSING?* Reply to this message either yes or no. If you answer *yes*, you are taken to panel DMS042C, which is shown on the next page. If you reply *no*, you will be taken directly to the volume selection panel.

Enter Data Set Name

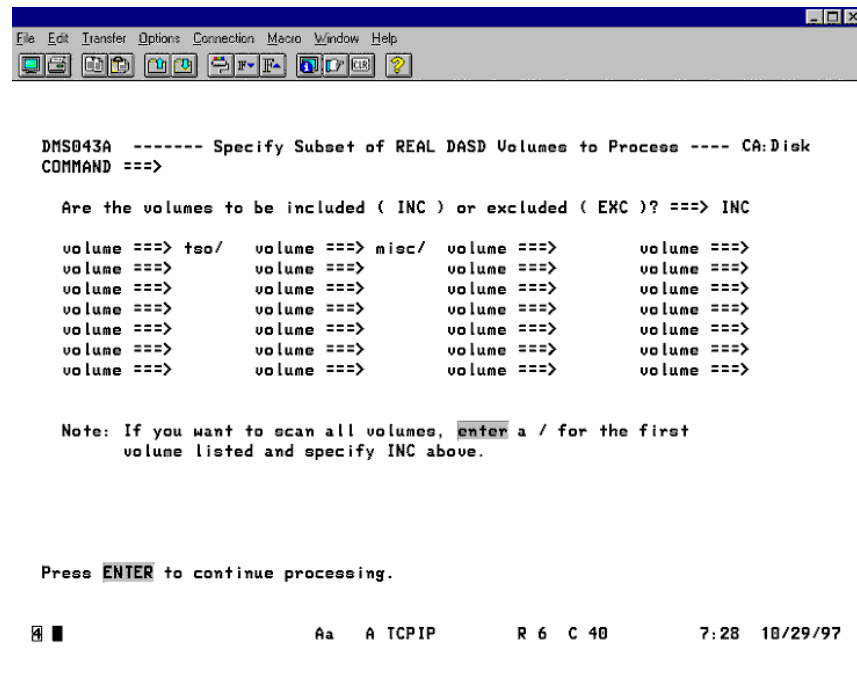
On this panel you are prompted to enter the name of an already-existing data set in which the names of processed data sets can be saved. The record length of this data set must be at least 50 and it must have a sequential organization.

The purpose of this facility is to allow the user to automatically generate utility commands, either for CA Disk or some other program, for the data set names that appear in the report. Although this panel does not actually generate the commands, you must specify the data set name here in order to produce the commands later on. It is always a good idea to save the data set names that appear on the report if there is even a remote chance that you will want to generate utility commands for them. After this panel has been bypassed, it is impossible to respecify this value until the next online report request.

For more information, see the section CMDGENDSN= in the chapter "Reports."

Specify Subset of Real DASD Volumes to Process

On this panel you specify which online volumes (depending on the option specified on the primary options menu) are to be scanned. You must specify at least one value, whether it is an explicit or pattern value. The following illustration is a sample Specify a Subset of DASD Volumes to Process panel:



It must be specified in the upper left input prompt. Any other volumes can be specified in any of the remaining prompt areas. The INC/EXC prompt tells CA Disk if the volume list specified is for inclusion or exclusion.

After all volumes or patterns have been specified, press <ENTER>. If you decide not to execute the reporting function, you can press <END> or type *CANCEL* in the command input area.

Specify Selection Criteria You Desire

This panel allows you to enter selection or exclusion tests in order to limit the data sets reported on to a subset of those actually on the volume(s) being scanned. For detailed information on the selection tests, see the section SELECT and EXCLUDE Rules in the chapter "Basic System – DMS PROC." This information is also discussed in the help text provided for this particular panel.

Each time you press the <ENTER> key from this panel, a SELECT or EXCLUDE command will be generated in DSCL format. When you have finished specifying the last test (or if you don't want to specify any selection tests), press <END> to continue. If you reply that you want to specify a list of data set names or a table name, you will be taken to the panel DMS043D, which is displayed on the next page.

Specify DSNAMES and/or Table

On this panel, you are instructed to specify a list of data set names to process. If the data set name or pattern is not enclosed in quotes, the user ID will be appended to the front of the data set name. Press <ENTER> to save the data set names and to return to the selection/exclusion specification panel.

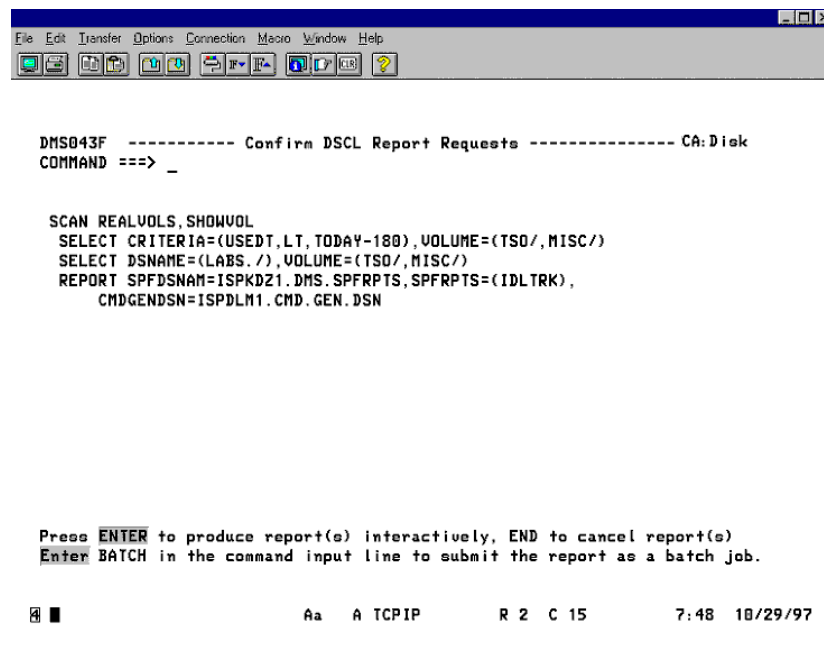
Select Desired Reports

This panel presents a scrollable list of all report names that are defined in the report definition library. To select a report, position the cursor next to the desired report name and enter an S.

Repeat this process for as many reports as you would like to produce. When you have completed the report selection process, press <END> to continue to the next panel.

Confirm DSCL Report Requests

Displayed on the screen are the actual DSCL commands that have been generated. You can recognize these as the same commands that you would submit in a batch job. In fact, you'll have a chance to save these commands later on so that you can either execute these again in a batch job, or execute them again online using the expert mode from the online reporting primary options panel. The following illustration is a sample Confirm DSCL Report Requests panel:



```

DMS043F ----- Confirm DSCL Report Requests ----- CA:Disk
COMMAND ==> _

SCAN REALVOLS,SHOWVOL
SELECT CRITERIA=(USED,T,LT,TODAY-180),VOLUME=(TSO/,MISC/)
SELECT DSNNAME=(LABS./),VOLUME=(TSO/,MISC/)
REPORT SPFDSNAM=ISPKDZ1.DMS.SPFRPTS,SPFRPTS=(IDLTRK),
      CMDGENDSN=ISPDLM1.CMD.GEN.DSN

Press ENTER to produce report(s) interactively, END to cancel report(s)
Enter BATCH in the command input line to submit the report as a batch job.

Aa  A TCPIP      R 2  C 15      7:48 10/29/97

```

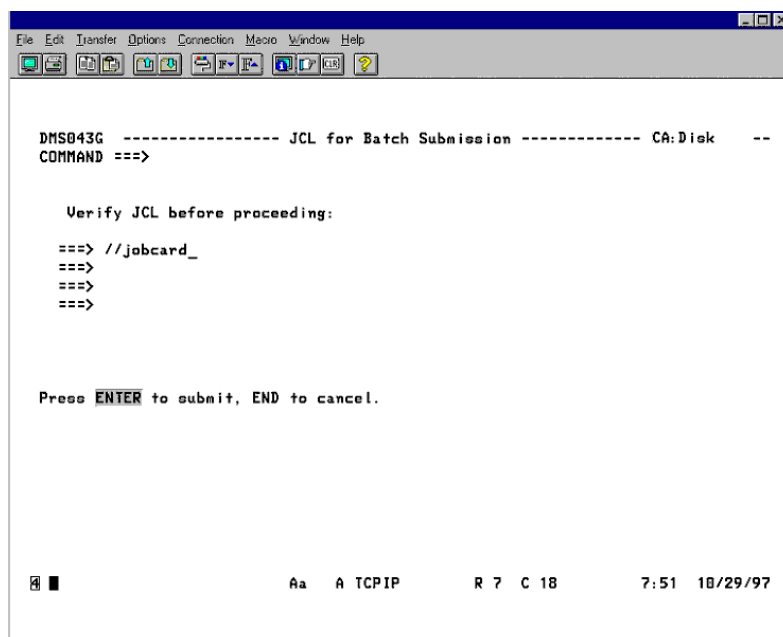
At this point, you can choose from the following options:

- Enter *BATCH* on the options line, which takes you to panel DMS043G to submit the job in batch
- Enter *CANCEL* or press <END>, which returns you to the primary options menu without executing the report
- Just press <ENTER> to produce the report online (if you are authorized to do so).

First let's look at what happens if you enter "BATCH" in the command line. You will be presented with the following screen:

JCL for Batch Reports

This panel gives you an opportunity to modify job card information prior to submitting the job for execution. You can either submit the job by pressing the <ENTER> key, or you can cancel the request by pressing <END>. The following illustration is a sample JCL for BATCH Reports:



If you do submit the job, you will be taken to the report disposition screen (DMS043H). Although you cannot browse the report (since it is a batch job), this allows you to save the DSCL commands so you can use them again in either a batch job or in expert mode from the primary options menu.

Now let's go back and see what happens when you run the reports online. Let's assume you're back at panel DMS043F to confirm the report requests and have just hit <ENTER>. The screen will clear and a message will appear saying *BEGINNING REPORT GENERATION*. Next, you will see the DSCL commands echoed back as they are read and interpreted by the command processor. You will soon be receiving a 2860 message for each volume that is processed, just to keep you informed as to where CA Disk is in processing. When the final volume is processed, you will receive one last 2860 message, which will give a count of the number of volumes processed -- the extraction phase has completed.

There are two more messages that you should receive before acquiring the report. The first is a message indicating when the sort phase begins, and the last one tells you that report formatting has begun. When the report is fully formatted, it will be presented to you in ISPF browse mode. A sample panel appears on the following page.

Browse Report

At this point, the report has been put in a temporary data set. You can use all of the facilities of browse, such as FIND, scrolling keys (up/down and left/right), and so on. The following illustration is a sample Browse Report:

```

Menu Utilities Compilers Help
-----
BROWSE  ISPDLM1.DMSRRPTS.D29OCT97.T082419      Line 00000000 Col 001 080
Command ==>                                     Scroll ==> PAGE
***** Top of Data *****
1997.302 OCT 29, 1997                          I D L E   S P A C E
WEDNESDAY 8.25 AM

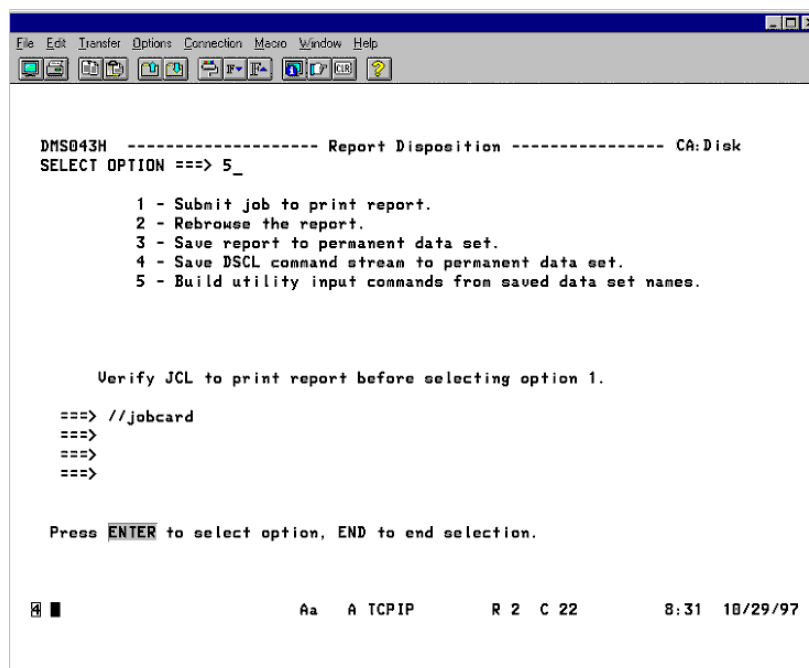
DATA SET NAME                                     TRACKS
-----
LABS.CUST.LOADLIB                                379   SSL802
LABS.DLR.MTFTABLE                                4     SSL802
LABS.DLR.P162810.USAM.INDEX                      5     SSL801
LABS.DLR.TEST.BACKUP                             1     SSL801
LABS.DMSRTR81.DSETR020.TESTCOPY.DATA.A002        42     SSL801
LABS.DMSRTR81.DSETR020.USAMCOPY.DATA.A002        28     SSL801
LABS.FM11.LOADLIB                                45     SSL801
LABS.GLH.BSCDS                                   25     SSL801
LABS.GLH.LOADLIB                                521     SSL803
LABS.GLH.PARMLIB                                 40     SSL803
LABS.GLH.R817.LOADLIB                            350     SSL803
LABS.ISPFLMF.TABLES                             74     SSL802
LABS.KCH.LOADLIB                                112     SSL803
Aa A TCPIP R 4 C 15 8:27 10/29/97

```

After you have finished browsing the report, press <END> to exit browse and continue to the report disposition panel.

Report Disposition

At this point in processing, the report that was generated is still in a temporary data set. The purpose of this screen is to allow you to perform several options, depending on the results of the report. When you have completed processing on this panel, press <END> to return to the primary options panel. The following illustration is a sample Report Disposition panel:

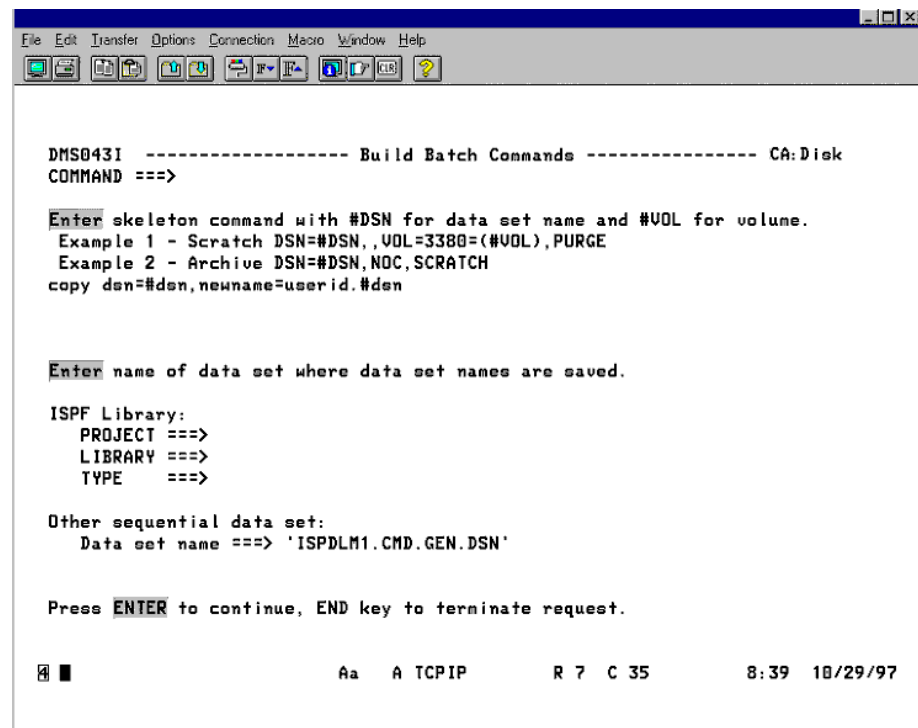


- Option 1 - SUBMIT JOB TO PRINT. This option submits a batch job to print the report that was just produced. Before selecting this option, be sure the job card information at the bottom of the panel is correct. The other function of the batch job is to delete the temporary report data set. To be sure that you have completed processing of the report, CA Disk will cause the temporary data set to be locked out until you leave the report disposition panel. Because of this, the batch job will not begin executing until you return to the reports primary options panel.
- Option 2 - REBROWSE THE REPORT. Use this option to enter browse again to review the report. This option is valid even if you already submitted the job to print.
- Option 3 - SAVE REPORT TO PERMANENT data set. If for some reason you would like to save the report to a permanent data set, select this option. You will be directed to another panel where you can enter the permanent data set's name.
- Option 4 - SAVE DSCL COMMAND STREAM TO PERMANENT data set. Remember those DSCL commands that were shown on the panel to confirm the report request? Those commands can reproduce the same results for reports in either the online or batch environment. If the report you just produced will be required again in the future, it can benefit you to save the commands in a permanent data set, particularly if you had a lot of complex criteria commands specified. This option would also be handy if you need to run the same report again with slight changes in the selection criteria. In this case you would save the commands and then return back to the primary options menu. From there you would select "expert" mode and you can merely modify the previous commands in ISPF edit mode. This would save a lot of energy and also prevent possible errors when respecifying all of the panel information again. If you want to submit the commands in a batch job, execute the procedure CA Disk and point the SYSIN dd card to the permanent data set.
- Option 5 - BUILD UTILITY INPUT COMMANDS FROM SAVED data set NAMES. This is a handy utility for generating utility commands -- either for CA Disk or some other utility -- based upon the data set names shown in the report. For instance, if you wanted to run the Move/Copy function on only a subset of the data sets on a volume, you can use the selection capability to limit the data sets to the proper subset. Then you would use this option to generate COPY commands for all of the data sets, which can subsequently be fed into a batch Move/Copy run. Note that to use this option you must have replied yes to the SAVE DSNAMES FOR LATER PROCESSING prompt on the primary options menu of this facility. On the next page you can see the panel that is displayed if you select this option.

Build Batch Commands

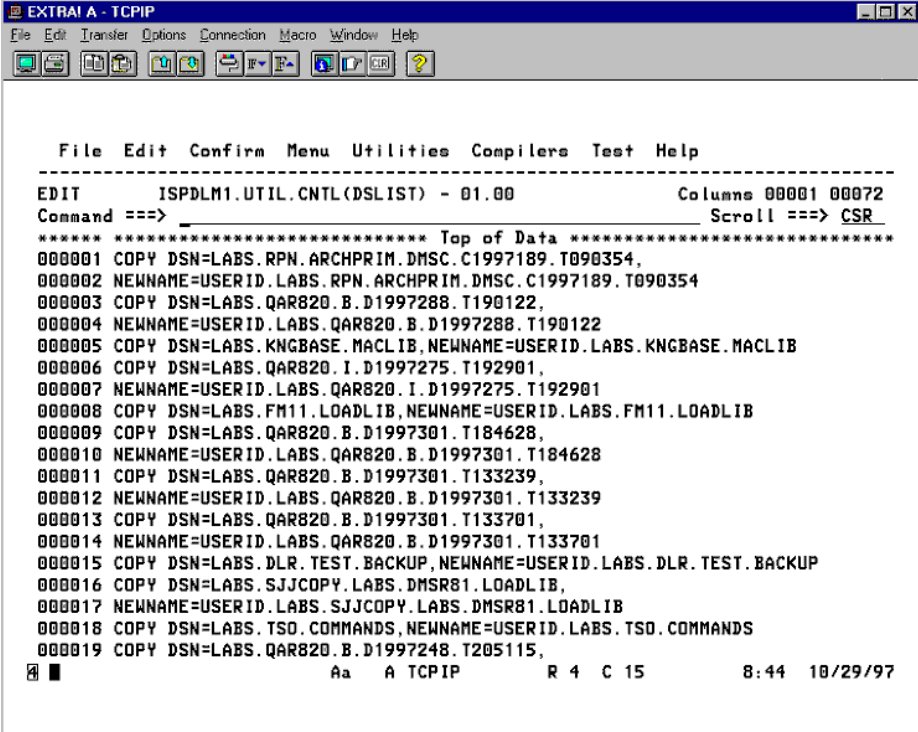
The build BATCH command panel serves two functions:

- Specify the pattern command to be generated
- Supply the name of the data set where the list of data set names shown on the report actually resides (this is the data set name that you specified on Enter Data Set Name). The following illustration is a sample Build BATCH Commands panel:



For every data set name found in this sequential data set, the pattern command will be generated, substituting the #DSN value with the actual data set name.

For example, if the pattern command specified is COPY
DSN=#DSN,NEWNAME=userid.#DSN, then the following EDIT session will be displayed:



```

EXTRACT A - TCP/IP
File Edit Transfer Options Connection Macro Window Help

-----
File Edit Confirm Menu Utilities Compilers Test Help
-----
EDIT      ISPDLM1.UTIL.CNTL(DSLIST) - 01.00      Columns 00001 00072
Command ==> _____ Scroll ==> CSR
***** ***** Top of Data *****
000001 COPY DSN=LABS.RPN.ARCHPRIM.DMSC.C1997189.T090354,
000002 NEWNAME=USERID.LABS.RPN.ARCHPRIM.DMSC.C1997189.T090354
000003 COPY DSN=LABS.QAR820.B.D1997288.T190122,
000004 NEWNAME=USERID.LABS.QAR820.B.D1997288.T190122
000005 COPY DSN=LABS.KNGBASE.MACLIB,NEWNAME=USERID.LABS.KNGBASE.MACLIB
000006 COPY DSN=LABS.QAR820.I.D1997275.T192901,
000007 NEWNAME=USERID.LABS.QAR820.I.D1997275.T192901
000008 COPY DSN=LABS.FM11.LOADLIB,NEWNAME=USERID.LABS.FM11.LOADLIB
000009 COPY DSN=LABS.QAR820.B.D1997301.T184628,
000010 NEWNAME=USERID.LABS.QAR820.B.D1997301.T184628
000011 COPY DSN=LABS.QAR820.B.D1997301.T133239,
000012 NEWNAME=USERID.LABS.QAR820.B.D1997301.T133239
000013 COPY DSN=LABS.QAR820.B.D1997301.T133701,
000014 NEWNAME=USERID.LABS.QAR820.B.D1997301.T133701
000015 COPY DSN=LABS.DLR.TEST.BACKUP,NEWNAME=USERID.LABS.DLR.TEST.BACKUP
000016 COPY DSN=LABS.SJJCOPY.LABS.DMSR81.LOADLIB,
000017 NEWNAME=USERID.LABS.SJJCOPY.LABS.DMSR81.LOADLIB
000018 COPY DSN=LABS.TSD.COMMANDS,NEWNAME=USERID.LABS.TSD.COMMANDS
000019 COPY DSN=LABS.QAR820.B.D1997248.T205115,
Aa A TCP/IP R 4 C 15 8:44 10/29/97

```

Online Reporting - Expert Mode

The expert mode allows you to *free form* the DSCL commands yourself, bypassing the multiple ISPF panel prompting that normally occurs. This allows for much quicker entry of the selection data, but it does require the user to know the language constructs of DSCL. Another useful feature is that previously generated DSCL commands can be recalled and executed from here, without having to re-specify all of the selection criteria. The following illustration is a sample Online Reporting - Expert Mode panel:

```

DMS043J ----- CA:Disk   Online Reporting--Expert Mode ----- CA:Disk   --
COMMAND ==>

Do you wish to use the system skeleton commands?
( YES or NO ) ==> y

If NO, enter the data set name where your report command is stored,
so it can be copied.
ISPF Library:
PROJECT ==> ISPDLM1
LIBRARY ==> UTIL
TYPE ==> CNTL
MEMBER ==>

Other partitioned or sequential data set:
Data set name ==>

Press ENTER to continue processing.

Aa  A TCP/IP  R 5 C 30  8:51 10/29/97

```

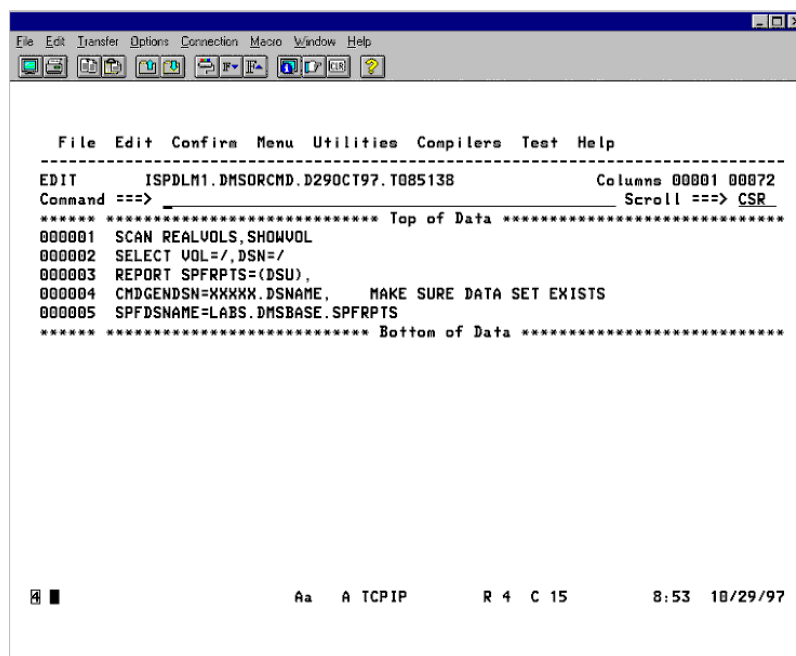
There are two different sources of input for this function. You tell CA Disk which one to use at the prompt *DO YOU want TO USE SKELETON COMMANDS?* If you answer *yes*, a basic outline of some DSCL commands will be copied from PARMLIB and you will be taken to edit mode of ISPF so you can modify the commands as desired. This option requires the greater amount of knowledge on the user's part and should only be used if you are knowledgeable about DSCL.

The second option is to answer *yes* or *no* to the previous prompt and supply the name of a data set that has DSCL commands already generated. Typically this would be the commands you saved at report disposition time on a previous run. CA Disk will copy these commands to a temporary data set and pass control to ISPF edit so you can modify the commands if you want.

As you can see, either response to the prompt will take you into ISPF edit mode. The only function the prompt served was to tell CA Disk the source of the DSCL command statements. After in edit mode, your screen should look something like this:

DSCL Command Edit

You are now in the standard edit facility of ISPF. If you want to modify the commands, you are free to do so. Keep in mind that you are actually editing a temporary data set, so the changes you make will not be saved at this time. If you want to save the changes, you can do so at the report disposition screen once you have generated your report(s). The following illustration is a sample DSCL Command EDIT Panel:



After you have modified the commands (or if you choose not to), press <END>. You will be passed to the report confirmation panel (DMS043F). From there the report generation process will proceed as previously described.

Chapter 18: TSO Command Processor

This section contains the following topics:

[Commands Available](#) (see page 617)

[General Information](#) (see page 618)

[DARCHIVE — Deferred Archive](#) (see page 618)

[DRESTORE — Deferred Restore](#) (see page 621)

[RESTORE — Immediate \(Dynamic\) Restore](#) (see page 623)

[LISTDMS — Listing the Archive Index](#) (see page 625)

[LISTREQ — List Status of Deferred Requests](#) (see page 628)

[DERASE — Erase a Deferred Request](#) (see page 631)

[Restrictions](#) (see page 631)

[User Exits](#) (see page 632)

Commands Available

The commands available for the TSO Command Processor are as follows:

- DARCHIVE
- DRESTORE
- DERASE
- LISTDMS
- LISTREQ
- RESTORE

General Information

Specialized support has been provided for the TSO environment in the form of command processors. To minimize the amount of familiarization necessary for TSO users to master the commands, the parse service has been used in the command processors for command and parameter formatting and validation.

Use of the TSO command processors requires some special installation tasks to be accomplished. Among the steps are the optional changing of the default names for the CA Disk files and parmlib data sets, discussed in *Identify Files and Parmlib in the Installation Guide*.

A data set to be archived must be found on the volume at the time the request is made. Requests to process data sets on offline volumes are allowed only through sysparm TSOVOLOF.

The commands for these TSO command processors are described in detail in this section. TSO HELP text is available for all command processors.

DARCHIVE — Deferred Archive

A TSO command processor is provided to enable a user to queue an archive request for later batch processing. This command processor interfaces to the same subcontrol module used when a batch user makes a deferred archive request.

Successful use of this command results in the placement of a record containing all of the specified parameters into the ARCHCMDS file. All pending command records are processed and appropriately flagged at the next execution of the program that disposes of the deferred archive requests.

The status of such requests can be interrogated by use of the LISTREQ command after submission of the request. The processed command records are kept on file for an installation-specified number of days for possible inquiries. After this period has elapsed, they are automatically deleted. The default period is five days. Sysparms DARCKEEP and DRESKEEP can be specified to alter the default for either or both.

Restrictions

Deferred archiving does not support a subset of a PDS as specified by a member list.

DARCHIVE Command

The DARCHIVE command requests the deferred archival of a data set.

The DARCHIVE command syntax is:

```
DARCHIVE {DSN(data set names)}  
        {VOLUME(volser)}  
        {NEWNAME(dsname)}  
        {EXPDT(date)}  
        {RETPD(nnnnn)}  
        [UNCATALOG]  
        [NOCATALOG]  
        [RECATALOG]  
        [SCRATCH]  
        [NOSCRATCH]  
        [RESET]  
        [NORESET]  
        {PASSWORD(password)}
```

DSN

A list of one to ten names of data sets to be archived. The DSN can be eliminated from the command line and a prompt for a data-set name list will be issued. In the case of a GDG, either the relative or absolute name can be used.

VOLUME

This parameter must be supplied if either no entry exists in the system catalog for the data set, or if the catalog points to a data set with the same name on a different volume.

NEWNAME

A data set can be indexed in the archives under a different name, supplied as the value of this parameter. In the case of a GDG, either the relative or absolute name can be specified.

EXPDT

This parameter can be used to provide the date on which the archived data set can be deleted from the archives. It must be in an accepted CA Disk format. For more information, see the section Year 2000 Considerations in the chapter "General Information."

RETPD

An alternative means of specifying the date on which the archived data set can be deleted from the archives is to specify a number of days it is to be retained in the archives. The number of days must be a one- to four-digit number. For more information, see the section Year 2000 Considerations in the chapter "General Information."

UNCATALOG

The normal default catalog action after a data set is archived, but it can also be specified.

NOCATALOG

This parameter can be used to preclude the default action of uncataloging at the completion of data set archival.

RECATALOG

The presence of this parameter results in the recataloging of a data set to the CA Disk pseudo- volume upon successful archival.

SCRATCH

The normal default scratch action after a data set is archived, but it can also be specified.

NOSCRATCH

The scratching of a data set that normally occurs after archival is suppressed by the inclusion of this parameter.

RESET

By default, the change bit for the data set is reset (turned off) if the data set is NOT scratched. Although this is the default, it can still be specified.

NORESET

The resetting of the change bit that normally occurs when a data set is backed up (archived, but not scratched) is suppressed by the inclusion of this parameter.

PASSWORD

The password parameter provides the password value required to access a protected VSAM cluster. If the cluster is to be archived but not scratched, the read level or higher is required. If the cluster is to be scratched, the master-level password is required.

DRESTORE — Deferred Restore

A TSO command processor is provided to enable a user to queue a restore request for later batch processing. The same interface is used by the TSO foreground as the batch background services for processing deferred restore requests.

Execution of this command causes a record containing the specified parameters to be placed in the RESTCMDS file. These records are processed during the next batch execution of the program that disposes of the deferred restore requests.

At the time the queued restore requests are processed, they are sorted into sequence by the archive volumes on which the data sets reside, and block number sequence within volumes. This is done to minimize tape mounts and positioning time.

The RESTCMDS file can be interrogated by use of the LISTREQ command to determine if a request is still pending or if the restore has been completed. The queued request records will automatically be deleted five days (the default) after the restore processing has actually been done. This period to keep the records available for display can be changed by sysparm DRESKEEP.

In the event the data set being queued is cataloged to the pseudo volser, the SYSCTLG parameter will be automatically generated for the request. If sysparm ARSECURE is set to either 'R' or 'N', the user submitting the DRESTORE request will only require INPUT authority. When the DISPOSE is run, if the data set is no longer cataloged to the pseudo volser, the request will be bypassed. This allows the TSO DRESTORE to function exactly like DMSAR under these circumstances.

DRESTORE Command

The DRESTORE command requests the deferred restoring of a data set.

The DRESTORE command syntax is:

```
DRESTORE {DSN(data set names)}  
        {NEWNAME(newname)}  
        {VERSION(-nn)}  
        {TIMEDATE(hhmmdate)}  
        {VOLUME(volser)}  
        {POOL(name)}  
        [NOCATALOG]  
        [RECATALOG]  
        [SCRATCH]  
        [ERASE]  
        {PASSWORD(password)}
```

DSN

A list of one to ten names of data sets to be restored. The DSN can be eliminated from the command line and a prompt for a data-set name list will be issued. The VERSION or TIMEDATE parameter can be used to designate other than the most recently archived version of a data set.

NEWNAME

To change the name of the data set being restored, this parameter must be specified.

VERSION

Specify -1 to -99 to restore the progressively older version of the data set to be restored. The absence of this parameter and the TIMEDATE parameter will default to restoring the most current version of the data set.

TIMEDATE

This parameter provides a more precise technique for designating a version of a data set to be restored. The format is hhmmdate, where *hhmm* is 24 hour time and *date* is any acceptable CA Disk date format, described in the section Date Formats in the chapter "General Information." The values can be obtained from either the batch LISTD or TSO LISTDMS index listing for the data set.

VOLUME

This parameter can be specified to restore to a volume other than the original source volume.

POOL

To cause the data set to be restored to a volume belonging to a predefined POOL of volumes, this parameter is required.

NOCATALOG

The inclusion of this parameter prevents the data set from being cataloged after it has been successfully restored.

RECATALOG

Default processing attempts to catalog all data sets, but will not change an existing catalog entry. Specify this parameter only when you want to force the data set to be cataloged to the target volume.

SCRATCH

If the data set already exists on the target volume, this parameter causes it to be scratched and then reallocated.

ERASE

This parameter will cause the directory of a preallocated partitioned data set to be formatted before being used.

PASSWORD

The password parameter provides the password value required to access a protected VSAM cluster. If the cluster is to be restored but not scratched, the update-level or higher is required. If the cluster is to be scratched, the master-level password is required.

RESTORE — Immediate (Dynamic) Restore

Data sets can be dynamically restored in the TSO user region by way of the RESTORE command. The command processor, if needed, dynamically allocates a tape drive, for the period of time necessary to restore the requested data set.

Note: Use of the dynamic (immediate) restore command requires that the RESTORE command processor be designated as privileged. This is necessary due to the use of the ENQ and ALLOCATE SVCs.

A facility is available to allow the installation to limit the number of tape units that can be concurrently allocated in support of TSO RESTORE commands. When the limit has been reached, execution of the command processor results in a response to the user that the command will have to be reissued at a later time.

This support does not require the dedication of tape units for its use. At the time of a restore request, an attempt is made to allocate a tape drive, if needed, according to the protocol observed by the system allocate function. Deallocation occurs immediately after the data set is restored.

RESTORE Command

The RESTORE command dynamically restores data sets. There is no restriction on the number of RESTORE commands that a user can execute during a session.

The RESTORE command syntax is:

```
RESTORE [DSNAME-LIST]  
      {MEMBER(mbrlist)}  
      {NEWNAME(newname)}  
      {VERSION( -nn)}  
      {TIMEDATE(hhmmdate)}  
      {VOLUME(volume)}  
      {POOL(name)}  
      [NOCATALOG]  
      [RECATALOG]  
      [SCRATCH]  
      [ERASE]  
      {PASSWORD(password)}
```

All parameter definitions for DRESTORE also apply to this command and are not repeated here.

Restoring Selected PDS Members

An additional capability in the immediate restore is to restore only selected members of a PDS. The members to restore are specified as follows:

```
MEMBER (mbrlist)
```

Where *mbrlist* specifies one or more members to be restored from the PDS.

RESTORE Message

If a tape drive is needed and no tape unit is available for dynamic allocation at the time the command processor is executed, a message will be returned to the user and processing will terminate.

LISTDMS — Listing the Archive Index

The LISTDMS command selectively lists the contents of the archive index.

The LISTDMS command syntax is:

```
LISTDMS {DA(dsname-list)}  
        {L(prefix-list)}  
        [STA]  
        [ALL]  
        [ARC]  
        [DUPLICATES]
```

DATA SET

A list of one or more data set names or patterns for which index entries are to be listed. The user ID will be prefixed to data set names not enclosed in quotes.

DA('A.B.C') will list only the most current entry. DA('A.B.C/') will list all occurrences.

Acceptable abbreviations for this keyword are DSN and DA.

LEVEL

Letter *L* is an acceptable abbreviation for level. A list of levels or data set name patterns to be listed. As with other TSO commands, the user ID is not prefixed to level entries. A prefix does not have to coincide with an index level. The entire index file can be listed by specifying *L(/)*. This keyword can be used instead of the DSNNAME parameter.

STATUS

This default parameter can also be specified to cause data set status information to be listed. An alternative keyword is STA. This parameter is mutually exclusive with the ALLOCATION and ARCHIVES parameters.

ALLOCATION

The presence of this parameter causes data set allocation information to be displayed. An alternate keyword is ALL. This parameter is mutually exclusive with the STATUS and ARCHIVES parameters.

ARCHIVES

The presence of this parameter causes archive volume information for the data set to be listed. An alternate keyword is ARC. This parameter is mutually exclusive with the STATUS and ALLOCATION parameters.

DUPLICATES

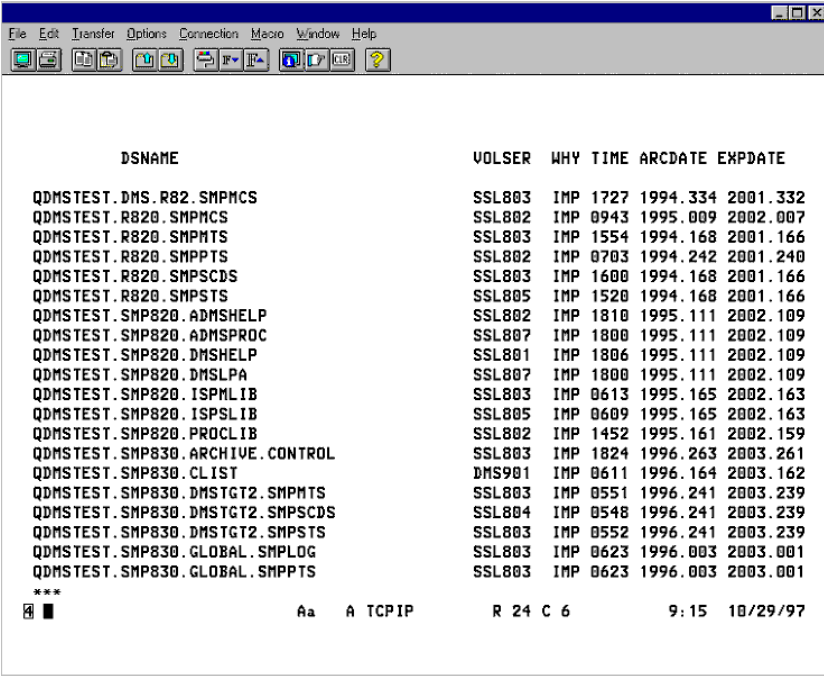
This optional parameter indicates that all versions of the specified data sets are to be listed. An alternate keyword is DUP. If this parameter is not specified, only the most recent copy of the data set will be displayed. If neither the data set nor the level parameter is specified, all entries under the user ID node will be listed. The status display is the default when no selection keyword has been specified.

LISTDMS — Sample Output

This section shows samples of the LISTDMS STATUS, ALLOCATTION, and ARCHIVES options.

LISTDMS STATUS Option

The following illustration is a sample Output from the LISTDMS STATUS Option panel:



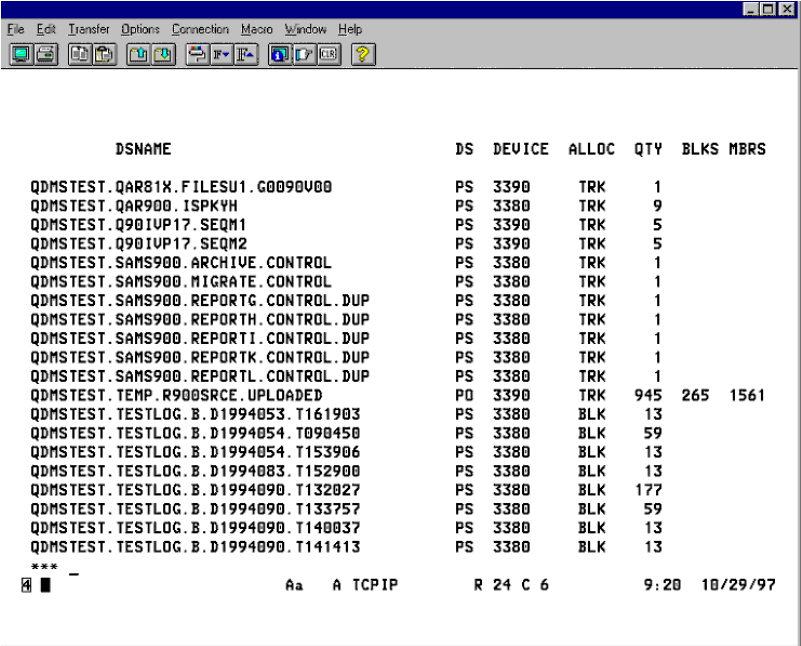
The screenshot shows a window titled "LISTDMS STATUS" with a menu bar (File, Edit, Transfer, Options, Connection, Macro, Window, Help) and a toolbar. The main area displays a table of archive data with columns: DSNAME, VOLSER, WHY, TIME, ARCDATE, and EXPDATE. The table lists various archive entries for QDMTEST, including SMPMCS, SMPMCS, SMPPTS, SMPPTS, SMPSCDS, SMPSTS, ADMSHELP, ADMSPROC, DMSHELP, DMSLPA, ISPLIB, ISPLIB, PROCLIB, ARCHIVE.CONTROL, CLIST, DMSTGT2.SMPPTS, DMSTGT2.SMPSCDS, DMSTGT2.SMPSTS, GLOBAL.SMPLOG, and GLOBAL.SMPPTS. The table ends with a line of asterisks and a status line showing "Aa A TCP/IP R 24 C 6 9:15 10/29/97".

DSNAME	VOLSER	WHY	TIME	ARCDATE	EXPDATE
QDMTEST.DMS.R82.SMPMCS	SSL803	IMP	1727	1994.334	2001.332
QDMTEST.R820.SMPMCS	SSL802	IMP	0943	1995.009	2002.007
QDMTEST.R820.SMPPTS	SSL803	IMP	1554	1994.168	2001.166
QDMTEST.R820.SMPPTS	SSL802	IMP	0703	1994.242	2001.240
QDMTEST.R820.SMPSCDS	SSL803	IMP	1600	1994.168	2001.166
QDMTEST.R820.SMPSTS	SSL805	IMP	1520	1994.168	2001.166
QDMTEST.SMP820.ADMSHELP	SSL802	IMP	1810	1995.111	2002.109
QDMTEST.SMP820.ADMSPROC	SSL807	IMP	1800	1995.111	2002.109
QDMTEST.SMP820.DMSHELP	SSL801	IMP	1806	1995.111	2002.109
QDMTEST.SMP820.DMSLPA	SSL807	IMP	1800	1995.111	2002.109
QDMTEST.SMP820.ISPLIB	SSL803	IMP	0613	1995.165	2002.163
QDMTEST.SMP820.ISPLIB	SSL805	IMP	0609	1995.165	2002.163
QDMTEST.SMP820.PROCLIB	SSL802	IMP	1452	1995.161	2002.159
QDMTEST.SMP830.ARCHIVE.CONTROL	SSL803	IMP	1824	1996.263	2003.261
QDMTEST.SMP830.CLIST	DMS901	IMP	0611	1996.164	2003.162
QDMTEST.SMP830.DMSTGT2.SMPPTS	SSL803	IMP	0551	1996.241	2003.239
QDMTEST.SMP830.DMSTGT2.SMPSCDS	SSL804	IMP	0548	1996.241	2003.239
QDMTEST.SMP830.DMSTGT2.SMPSTS	SSL803	IMP	0552	1996.241	2003.239
QDMTEST.SMP830.GLOBAL.SMPLOG	SSL803	IMP	0623	1996.003	2003.001
QDMTEST.SMP830.GLOBAL.SMPPTS	SSL803	IMP	0623	1996.003	2003.001

Aa A TCP/IP R 24 C 6 9:15 10/29/97

LISTDMS ALLOCATION Option

The following illustration is a sample Output from the LISTDMS ALLOCATION Option panel:

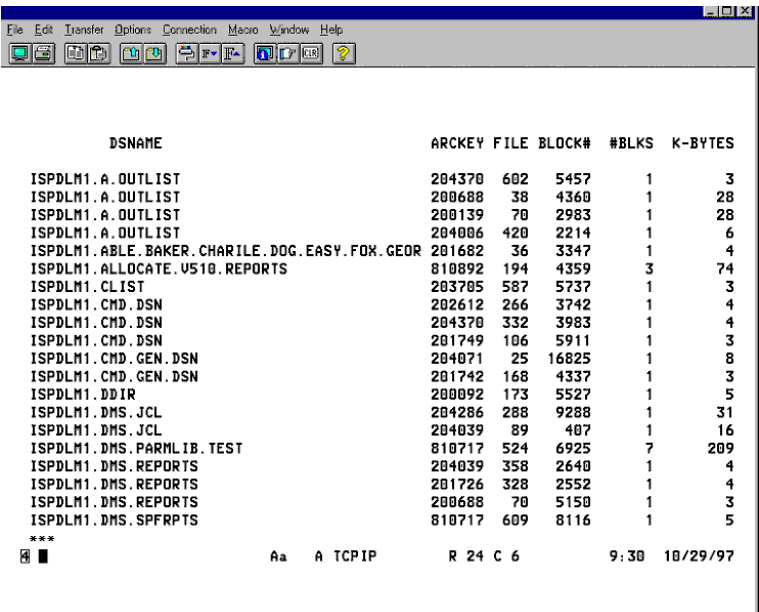


DSNAME	DS	DEVICE	ALLOC	QTY	BLKS	MBRS
QDMSTEST.QAR81X.FILESU1.G0090V00	PS	3390	TRK	1		
QDMSTEST.QAR900.ISPKYH	PS	3380	TRK	9		
QDMSTEST.Q90IUP17.SEQM1	PS	3390	TRK	5		
QDMSTEST.Q90IUP17.SEQM2	PS	3390	TRK	5		
QDMSTEST.SAMS900.ARCHIVE.CONTROL	PS	3380	TRK	1		
QDMSTEST.SAMS900.MIGRATE.CONTROL	PS	3380	TRK	1		
QDMSTEST.SAMS900.REPORTG.CONTROL.DUP	PS	3380	TRK	1		
QDMSTEST.SAMS900.REPORTH.CONTROL.DUP	PS	3380	TRK	1		
QDMSTEST.SAMS900.REPORTI.CONTROL.DUP	PS	3380	TRK	1		
QDMSTEST.SAMS900.REPORTK.CONTROL.DUP	PS	3380	TRK	1		
QDMSTEST.SAMS900.REPORTL.CONTROL.DUP	PS	3380	TRK	1		
QDMSTEST.TEMP.R900SRCE.UPLOADED	PD	3390	TRK	945	265	1561
QDMSTEST.TESTLOG.B.D1994053.T161903	PS	3380	BLK	13		
QDMSTEST.TESTLOG.B.D1994054.T090450	PS	3380	BLK	59		
QDMSTEST.TESTLOG.B.D1994054.T153906	PS	3380	BLK	13		
QDMSTEST.TESTLOG.B.D1994083.T152900	PS	3380	BLK	13		
QDMSTEST.TESTLOG.B.D1994090.T132027	PS	3380	BLK	177		
QDMSTEST.TESTLOG.B.D1994090.T133757	PS	3380	BLK	59		
QDMSTEST.TESTLOG.B.D1994090.T140037	PS	3380	BLK	13		
QDMSTEST.TESTLOG.B.D1994090.T141413	PS	3380	BLK	13		

Aa	A	TCPIP	R 24 C 6	9:20	10/29/97	

LISTDMS ARCHIVES Option

The following illustration is a sample Output from the LISTDMS ARCHIVES Option panel:



DSNAME	ARCKEY	FILE	BLOCK#	#BLKS	K-BYTES
ISPDLM1.A.OUTLIST	204370	602	5457	1	3
ISPDLM1.A.OUTLIST	200688	38	4360	1	28
ISPDLM1.A.OUTLIST	200139	70	2983	1	28
ISPDLM1.A.OUTLIST	204006	420	2214	1	6
ISPDLM1.ABLE.BAKER.CHARILE.DOG.EASY.FOX.GEOR	201682	36	3347	1	4
ISPDLM1.ALLOCATE.VS10.REPORTS	810892	194	4359	3	74
ISPDLM1.CLIST	203705	587	5737	1	3
ISPDLM1.CMD.DSN	202612	266	3742	1	4
ISPDLM1.CMD.DSN	204370	332	3983	1	4
ISPDLM1.CMD.DSN	201749	106	5911	1	3
ISPDLM1.CMD.GEN.DSN	204071	25	16825	1	8
ISPDLM1.CMD.GEN.DSN	201742	168	4337	1	3
ISPDLM1.DDIR	200092	173	5527	1	5
ISPDLM1.DMS.JCL	204286	288	9288	1	31
ISPDLM1.DMS.JCL	204039	89	407	1	16
ISPDLM1.DMS.PARMLIB.TEST	810717	524	6925	7	209
ISPDLM1.DMS.REPORTS	204039	358	2640	1	4
ISPDLM1.DMS.REPORTS	201726	328	2552	1	4
ISPDLM1.DMS.REPORTS	200688	70	5150	1	3
ISPDLM1.DMS.SPFRTS	810717	609	8116	1	5

Aa	A	TCPIP	R 24 C 6	9:30	10/29/97

LISTREQ — List Status of Deferred Requests

A TSO command processor is available for displaying the status of requests for deferred archiving and restoring of data sets. The display will include the requests submitted in both TSO and batch environments.

The status indicated for each request will be displayed as one of the following values:

PEND

Processing of the request is pending

COMP

Processing of the request has completed successfully

FAIL

Processing of the request has failed

ACTV

Processing of the request is currently active

The request records will normally remain in their respective command files for five days subsequent to processing.

LISTREQ command

The command and parameters necessary to list the status of queued archive and restore requests are described below.

Note: Batch functions are also available for listing the status of deferred archive and restore requests. For detail information, see the section Listing of Deferred Archive Request (LISTREQ) in the chapter "Backup/Archive" and the section Listing of Deferred Restore Requests in the chapter "RESTORE/RECOVER."

The LISTREQ command syntax is:

```
LISTREQ {DA(dsname-list)}  
        {L(prefix-list)}  
        [ARCHIVE]  
        [RESTORE]
```

DATA SET

A list of one or more data set names for which index entries are to be listed. The user ID will be prefixed to data set names not enclosed in quotes. Acceptable abbreviations for this keyword are DSN and DA.

LEVEL

Letter *L* is an acceptable abbreviation for LEVEL.

A list of levels or prefixes of data set names to be listed. As with other TSO commands, the user ID is not prefixed to level entries. A prefix does not have to coincide with an index level. The entire index file can be listed by specifying L(/).

This keyword can be used instead of the DSNAME parameter.

ARCHIVE

The presence of this parameter results in queued archive request records being selected. It can be abbreviated as ARC.

RESTORE

The presence of this parameter results in queued restore request records being selected. It can be abbreviated as RES.

Note: If neither the DATASET nor the LEVEL parameter is specified, all entries under the user ID will be listed. If neither the ARCHIVE nor the RESTORE keyword is specified, RESTORE is assumed.

LISTREQ — Sample Output

The following illustration is a sample LISTREQ Report panel:

RESTORE / DSNAME	HHMM REQDATE	STAT	HHMM RUNDATE
COM, U41XPROD, SOURCE	0857 1993.342	FAIL	0903 1997.287
CSLUL2, DOC, P163085, DUMP	1714 1997.225	ACTU	0000 1997.287
C279305, PLIB	1447 1996.283	ACTU	0000 1997.287
DIM, U2R1M0, ASM	1303 1996.352	ACTU	0000 1997.287
DISK, RHH, RHHPOS	1448 1996.338	ACTU	0000 1997.287
DISK820, \$\$\$, LOAD	1340 1995.272	ACTU	0000 1997.287
DISK820, DLH, LANGLIST	1832 1995.170	FAIL	0903 1997.287
DISK820, DLH, LANGLIST	1831 1995.170	FAIL	0903 1997.287
DISK820, ISPFLMF, CONTROLS	0802 1995.116	ACTU	0000 1997.287
DISK820, P1609540, MXREF	1144 1995.138	ACTU	0000 1997.287
DISK820, SCCS, FILES	0959 1995.284	ACTU	0000 1997.287
DISK820, SCCS, FILES	1028 1995.256	PEND	
DISK820, TEMP, ASM	1506 1994.186	ACTU	0000
DISK820, WK131958, LANGLIST	1456 1995.271	ACTU	0000
DISK820, WK611309, PROCLIB	2354 1995.151	ACTU	0000
DISK900, DLH, LNKT	1249 1997.197	ACTU	0000
DISK900, DLH, OBJ	1434 1997.301	PEND	
DISK900, GLH, OBJ	1049 1997.002	ACTU	0000
DISK900, P1135526, LOAD	1646 1997.218	ACTU	0000
DISK910, BASE, MXREF	1409 1997.126	ACTU	0000

Aa A TCP/IP	R 24 C 6	9:56	10/29/97

DERASE — Erase a Deferred Request

The DERASE command deletes deferred archive and deferred restore requests.

Note: The DERASE command can also be used in batch runs. For detail information, see the section Erasing of Deferred ARCHIVE Requests in the chapter "Backup/Archive" and the section Erase a Deferred Restore Request from the Queue in the chapter "RESTORE/RECOVER."

The DERASE command syntax is:

```
DERASE {DSNAME(S)}  
      [ARCHIVE]  
      [RESTORE]
```

DSNAME(S)

A list of one or more data set names for which deferred requests are to be deleted. A maximum of 20 data set names is allowed. The user ID will be prefixed to data set names not enclosed in quotes.

ARCHIVE

The presence of this parameter indicates that the deferred requests to be deleted are archive transactions.

RESTORE

The presence of this parameter indicates that the deferred requests to be deleted are restore transactions.

Note: If neither the ARCHIVE nor RESTORE keyword has been specified, RESTORE will be the default.

Restrictions

- Under MVS, the use of the dynamic (immediate) restore command requires that the restore command processor be designated as privileged. See Activating TSO Support in the *Installation Guide* for more information on how to authorize a command processor.
- Since the ISPF program product does not support the execution of command processors requiring authorization under the TSO function menu screen, the dynamic restore function cannot be executed from within the ISPF environment.
- The multitude of parameters unique to VSAM for altering cluster attributes are not supported in either the DRESTORE or the RESTORE command processors. Batch requests must be made whenever attributes are to be changed when restoring a cluster.

User Exits

User exits are provided that allow the installation to screen all TSO requests. Consult User Exits in the *Systems Guide* for details regarding their use.

Chapter 19: Testing CA Disk after Installing

This section contains the following topics:

[Test the CA Disk System](#) (see page 633)

[Testing the CA Disk Auto-Restore Feature](#) (see page 638)

[Special Testing Environment](#) (see page 639)

Test the CA Disk System

The CA Disk Installation Verification Process (IVP) is a set of jobs that demonstrate the functions of CA Disk. Go to the CCUWJCL data set to see the members that are shown. The PROCLIB customization must be completed before executing these jobs.

This introductory approach to testing lets you become familiar with the basic functions, commands, and coding conventions. After this testing, you can implement a solution for your installation. Many functions and options are not covered, but this preliminary testing introduces techniques that you can use to deal with data storage management problems.

Run the tests in the order listed.

For more information, see [Basic System DMS PROC](#) (see page 59). The added information helps you understand the following examples and provides ideas for customizing them.

Test Various Volume and Data Set Level Reports (IVP01RPT)

The first function of evaluating CA-Disk is to produce many reports. The reports include:

- [CLDATA report](#) (see page 687)
- CLDATA report with options
- [Cluster internal analysis report](#) (see page 688)
- Data set level reports
- PDS reports
- Cluster internal analysis with options
- Volume level reports

For each volume found online, the following JCL produces summary reports including information such as:

- Volume free space
- Types of data sets
- Data set size distributions

Since these reports are summary reports, each volume produces only one report line with 24 volumes on one page.

If your configuration has many DASD devices, limit the report to a selected few volumes by adjusting the [SELECT command](#) (see page 82) to the input stream.

Note: Most of the parameters can be abbreviated to the first three characters. A CA-Disk command can start in any column, including column one. If the volume list contains only one entry, no parentheses are needed.

Test Move and Copy Functions (IVP02MOV)

Copy and move is the second function when evaluating CA-Disk.

Follow these steps:

1. Copy an IVP data to a new volume with a new name.
2. Move the new data set from the new volume to the copied data set volume.
3. Compare the moved data set with the test data set.

Test Idle Space Release Function (IVP03ISR)

If data sets are over-allocated and wasting space, CA Disk provides a flexible and convenient means to free this space. You can establish thresholds which allow extra space but, if exceeded, the unused space is released.

This JCL has the following steps:

1. Releases IVP data sets to 50 percent used threshold.
2. Releases IVP data sets to 100 percent used.
3. Converts cylinder-allocated non-VSAM to tracks.
4. Converts secondary allocation of non-VSAM to 50 percent of the primary.

Test Backup and Batch Restore Functions (IVP03ISR)

The JCL backs up the IVP data sets then restores selected data sets. Consider the test backup and batch restore function an automated archival. The operation scans all the data sets on a volume or group of volumes. Then, based on the criteria you supply, the operation archives certain data sets and retains others on the volume. Because test backup and batch restore is a powerful function, a SIMULATE mode is provided. We recommend that you run only simulations until you achieve the results you want.

Specification of the retention criteria is flexible and complex. **See the information in the Basic System—DMS PROC and Backup/Archive sections and the PARMLIB section of the Systems Guide when you set up your own DSCL selection criteria.**

The evaluation JCL allows backup of an empty file. The ARCEMPTY parameter is specified under //SYSPARMS DD to allow backup operations to archive empty data sets.

After backup, several data sets are in the archives. To restore the archived data sets, use the RESTORE function.

If you want the data set restored to a volume other than the one it was archived from, include the VOL= parameter to direct CA Disk to a particular volume. Default disposition for a restore operation is to catalog the data set following the restore. Because the original data set still exists on DASD, use the PREALLOCY parameter to allow Restore operations to overwrite pre-existing non-VSAM data sets.

Test Expire Function (IVP05EXP)

Use this JCL to evaluate the expire function. If a data set satisfies DSCL selection and CA Disk determines that it is expired, the data set is deleted from DASD.

This JCL expires using the current date, and then expires using a future date. The last step reports on remaining data sets.

Test Various PDS compress Functions (IVP06CMP)

CA Disk provides a PDS management facility that includes PDS reports and a PDS compression utility. Run the PDS reports using the standard CA Disk report procedure to include:

- A PDS Status Report which includes the data set and directory statistics
- A PDS Member Report that is a 12-up listing of the member names that reside in the PDS

Run the compression utility explicitly or in a scan mode. The scan mode evaluates the status of a PDS to determine whether to compress the data set. You specify certain threshold values that the evaluation module uses to make its determination.

The JCL runs the compress function in two steps:

1. Compresses the PDS leaving free space.
2. Compresses the PDS without free space.

Test XCOPY Function (IVP07XCP)

XCOPY copies data from an archive and backup ARCHVOL onto a Disaster Recovery ARCHVOL.

The JCL runs these steps:

1. Creates an output FILES data set.
2. Copies data from one FILES data set to another based on data set name.
3. Executes LISTD and LISTV to show the input FILES data set after XCOPY and the output FILES data set after XCOPY.

Test Merge Functions (IVP08MRG)

Each time an archive or backup run is made, another archive tape is created (or an archive data set on disk). As time passes, consolidate archive tapes (or move the archives on disk off to tape) and remove the expired data sets.

CA Disk provides an archive MERGE function that successively processes each archive tape volume, or an archive data set on disk, as input. It copies and distributes only the unexpired data sets.

The simple JCL merges data from FILES data set to new volumes.

Test Recover Function (IVP09DRC)

After CA Disk has been installed and running for some time, the archive contains a copy of the data sets. If you experience the loss of a DASD volume due to a hardware or software failure, CA Disk can help in recovering the lost data.

The JCL recovers all data sets previously backed up and scratches them if they have been restored.

Test Delete Function (IVP10DEL)

The JCL deletes IVP data sets.

The DELETE command deletes the non-VSAM data sets and VSAM clusters that satisfy the selection criteria without taking a backup copy of the data set. To use this command, specify sysparm DELETEDOK with a value of Y.

Important: Use this command with care. Extreme and unrecoverable DATA LOSS can occur if used improperly.

Test Files Data Set Management Functions (IVP11IXM)

The final evaluation JCL is purge. CA Disk purges entries from the archive index regardless of the expiration date. Purge applies to both archived data sets (DSNINDEX records) and archive volumes (ARCHVOLS records).

1. Displays the archive index.

The DSNINDEX is keyed on data set name and has one entry for each data set or copy of a data set that resides in the archives.

The ARCHVOLS is keyed on volume serial and contains one entry for each archive tape volume.

2. Purges all data sets specified in DSN=.

Testing the CA Disk Auto-Restore Feature

There are several good reasons to perform thorough system testing of the auto-restore feature. Some of the benefits that can be derived are:

- You become familiar with the characteristics of the system.
- You experience what the system does under varying external conditions.
- The system operators have a chance to get used to the messages that the system issues. They also have a chance to respond to its operator prompts where they are required.
- If a user screening exit has been written, this provides an excellent opportunity for testing its effectiveness.
- You can formulate some ideas of your own for implementation guidelines.

Forcing an Auto-Restore to be Initiated

To force invocation of the catalog management support, you must first recatalog an archived data set to the CA Disk pseudo-volume. The easiest way to do this is to run an explicit CA Disk archive job with the parameter DISP=RECAT specified. This causes CA Disk to recatalog the data set after the scratch has taken place.

After you have the data set recataloged, you can invoke the support by either running a batch job that references the data set (such as IEBGENER), or by attempting to edit the data set from a TSO/ISPF environment. If you do this from an online environment, you are prompted by several messages:

- Do you want to restore the data set?
- Do you want an immediate or deferred restore?
and, if an immediate restore is requested,
- Do you want to wait for the restore to complete?

Each of these questions is explained more thoroughly in the Customizing the TSO/ISPF Auto-Restore Environment section.

Testing Auto-Restore in a Production Environment

Many possible job combinations can be run to simulate production environments. The following list details possible tests that can be run. The output reports created by the system task DMSAR should be routed to hard copy so you can review. All data set references in the examples are to non-existent data sets; that is, no format-1 DSCBs exist for the data set.

1. Run a test specifying a data set that can be found in the archives index, and one that cannot.
2. Execute two auto-restore tasks simultaneously by forcing two jobs to execute at the same time (their job names must be different), each job requesting a different archived data set.
3. Try running the test as in 2 above, but this time reference different data set names that reside on the same archive tape volume (you will have to do a LISTD on an archive volume to find a suitable pair of data sets).
4. If you are supplying an ARESPREX screening exit, or using other options to tailor the function to your environment, try running a few conditions that test your implementation. Make sure the restore requests are processed as expected.

Special Testing Environment

Testing and implementing CA Disk functions that make use of a data set's last used date (for example, archival), or perhaps its last modified date (for example, special backup option), requires some special consideration if the CA Disk SVC (or MVS SU 60) has not been installed long enough to have these fields properly updated on all data sets.

The first option is to simply wait an appropriate period before implementing a dependent function. The second is to use the DSCB Update Utility (see the DSCB Update Utility section) to initialize these fields to an appropriate starting value. The third is to use the following sysparms for testing, and then follow either option one or two.

Sysparms DSCBLUSD and DSCBLMOD normally provide the decimal offset into the format-1 DSCB where the last used and last modified dates are being recorded. For testing, however, you specify them with an offset of either the creation date (decimal offset 53) or the expiration date (decimal offset 56). For your test purposes, these values cause CA Disk to treat the creation date (or expiration date) as the last used and last modified dates as well. In the example here, the sysparms have been specified with the offset of the creation date:

```
DSCBLUSD053  
DSCBLMOD053
```

Note: Be sure to remove these entries after the CA Disk SVC or SU 60 has been active long enough to have recorded proper values. Be sure that you do not have these testing values set if you execute the DSCB Update Utility to initialize the fields.

If a local modification to OPEN is already being used to record either or both of these fields in the DSCB, the sysparm entries can be used to direct CA Disk to the fields maintained by your own local modification. See the descriptions of these system parameters in the *Systems Guide*.

Appendix A: Health Checks

This Appendix describes health checks for CA Disk. The check owner for all CA Disk Health Checks is CA_DISK.

This section contains the following topics:

[DISK_SYSPARM_CARTCALC_N](#) (see page 642)

[DISK_SYSPARM_DSNDUPVL_N](#) (see page 643)

[DISK_SYSPARM_IOCHNBK_1](#) (see page 644)

[DISK_SYSPARM_IOTRACKS_1](#) (see page 645)

[DISK_SYSPARM_MERGSORT_S](#) (see page 646)

[DISK_SYSPARM_IXMQTIME_0](#) (see page 647)

DISK_SYSPARM_CARTCALC_N

Description

Setting the sysparm CARTCALC to the value N or allowing it to use the default value N, turns off the query to the device for the device capacity. This causes the value in the sysparm CARTFEET to be used to set the cartridge capacity. Having sysparm CARTCALC set to N can cause under-utilization of tape cartridges. It may also cause unnecessary I/Os to be performed on a tape if the calculated capacity is greater than the actual capacity, because an attempt may be made to fit a data set that is too large for the remaining tape space by wasting time and resources. This exception will be cleared when this check runs again after the job causing the exception is changed and runs, or after the job statistics are older than the number of days specified in the AGE parameter for this health check. The default value for the sysparm CARTCALC will be changed to Y with the next release of CA Disk.

Best Practice

Set the sysparm CARTCALC to Y to get the best utilization of the tape device by using the least number of resources. This means a higher utilization for each cartridge regardless of type. A tape library with mixed capacity tapes or with high capacity tapes must have CARTCALC specified as Y, else cartridges will be underutilized.

Parameters Accepted

AGE(7)

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see the *Systems Guide*.

Messages

ADSH0011E SYSPARM CARTCALC is set to N while using a tape cartridge device.

DISK_SYSPARM_DSNDUPVL_N

Description

DSNDUPVL provides a way to handle data sets with the same name that reside on different volumes.

Setting DSNDUPLV to N CA Disk considers the DSNINDEX records for the data sets to be different data sets. These records are handled separately for version checking and for retrieval of DSNINDEX information.

Setting DSNDUPLV to Y CA Disk considers the DSNINDEX records for the data sets to be the same data set. This setting allows CA Disk to process data sets as they are moved from volume to volume.

DSNDUPVL is used by the following processes:

- REPORTS
- BACKUP/ARCHIVE
- BACKUPCC
- IXMAINT

IXMAINT processing uses it when the COPIES= parameter is specified.

- DSCL

DSCL selection processing uses DSNDUPVL when information from DSNINDEX (ARCDATE) is needed.

- RECOVER

RECOVER processing uses the DSNDUPVL when information from DSNINDEX (ARCDATE) is needed.

- Fast ARCHIVE and MERGE

Fast ARCHIVE and MERGE processing use this SYSPARM for selecting the DSNINDEX record that will be reused.

Best Practice

To allow a data set to move to different volumes and still be recognized as the same data set, set the sysparm DSNDUPVL to Y.

Parameters Accepted

AGE(7)

Debug Support

Yes

Verbose Support

Yes

Reference

See "System Parameter Definitions" in the *Systems Guide*.

Messages

ADSH0061E SYSPARM DSNDUPVL is set to N which does not allow volume pooling for data sets.

DISK_SYSPARM_IOCHNBLK_1

Description

Setting the sysparm IOCHNBLK to the value 1 or allowing it to use the default value 1, causes Backup and Archive jobs to write only one block to a tape at a time. This causes the job to run slower and to use more processing time. Sysparm IOCHNBLK can accept a value from 1 to 9, which is the number of tape blocks to be written with one I/O. Setting a value of 9 generally improves the performance significantly. This exception will be cleared when this check runs again after the job causing the exception is changed and runs, or after the job statistics are older than the number of days specified in the AGE parameter for this health check. The default for the sysparm IOCHNBLK will be changed to 9 with the next release of CA Disk.

Best Practice

Set the sysparm IOCHNBLK to 9 to get the best performance on tape I/O activity during a Backup or Archive job. Getting the best performance on tape I/O results in reduced processing time for your required Backup and Archive jobs, which will shrink the Backup/Archive processing window.

Parameters Accepted

AGE(7)

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see the *Systems Guide*.

Messages

ADSH002IE Sysparm IOCHNBLK is set to 1 while using a tape device.

DISK_SYSPARM_IOTRACKS_1

Description

Setting the sysparm IOTRACKS to a value less than the number of tracks in a cylinder causes Backup, Archive, and Restore jobs to issue more read/write requests to the DASD device. This causes the job to run slower and use more processing time and CPU resources. This exception will be cleared when this check runs again after the job causing the exception is changed and runs, or after the job statistics are older than the number of days specified in the AGE parameter for this health check.

Best Practice

Set the sysparm IOTRACKS to 30 to get the best performance on disk I/O activity during a Backup or Archive job. Getting the best performance on DASD I/O results in reduced processing time for your required Backup and Archive jobs.

Parameters Accepted

AGE(7),MIN(15)

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see the *Systems Guide*.

Messages

ADSH0031E Sysparm IOTRACKS is set to less than the number of tracks in a cylinder.

DISK_SYSPARM_MERGSORT_S

Description

Setting the sysparm MERGSORT to a value other than S causes Merge to sort the tapes to be processed by the volume serial number (VOLSER). Setting the sysparm MERGSORT to S allows Merge to process the ARCHVOLs according to the least space used rather than the alphanumeric VOLSER order and allows Merge to process more efficiently when smaller processing windows are needed. This exception will be cleared when this check runs again after the job causing the exception is changed and runs, or after the job statistics are older than the number of days specified in the AGE parameter for this health check. The default for the sysparm MERGSORT will be changed to S with the next release of CA Disk.

Best Practice

Set the sysparm MERGSORT to S to allow MERGE to process the ARCHVOLs according to the least space used rather than alphanumeric order. This setting allows MERGE to process more efficiently when smaller processing windows are needed.

Parameters Accepted

AGE(7)

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see the *Systems Guide*.

Messages

ADSH0041E Sysparm MERGSORT is not set to S.

DISK_SYSPARM_IXMQTIME_0

Description

The CA Disk utility IXMAINT issues a RESERVE-type enqueue to prevent concurrent updates to the Files Data Set (FDS) so that the integrity of the data set can be maintained. The value of the sysparm IXMQTIME defines how long the RESERVE will be held before it is released to allow other jobs to access the FDS. Setting the sysparm IXMQTIME to zero causes IXMAINT to retain the RESERVE for the entire length of the job. This will prevent other jobs, such as DMSAR Auto-Restores, from being able to access the FDS while the RESERVE is held. Also, if the RESERVE was not converted to a global enqueue, it will prevent jobs on systems other than the one on which the RESERVE was issued from accessing the volume on which the FDS resides. This can suspend any production work attempting to access the volume. This exception will be cleared when this check runs again after the job causing the exception is changed and runs, or after the job statistics are older than the number of days specified in the AGE parameter for this health check. The default value for the sysparm IXMQTIME will be changed to 0300 with the next release of CA Disk.

Best Practice

Set IXMQTIME to 0300 (30 seconds) to allow the needed data sets to be restored in a timely manner. When using a Files Data Set, integrity is maintained using ENQs which can prevent other CA Disk jobs from processing. Because IXMAINT is typically a longer running job, this may delay critically needed data sets from being restored.

Parameters Accepted

AGE(7),MIN(1)

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see the *Systems Guide*.

Messages

ADSH0051E Sysparm IXMQTIME is set to zero.

Appendix B: Examples

The following examples of various CA Disk executions are grouped by the function being performed.

This section contains the following topics:

[REPORT Examples](#) (see page 649)
[VREPORT Examples](#) (see page 651)
[ARCHIVE Examples](#) (see page 651)
[RESTORE Examples](#) (see page 653)
[Volume—Level BACKUP Examples](#) (see page 657)
[Volume—Level RECOVER Examples](#) (see page 658)
[Volume Defragmentation Examples](#) (see page 659)
[RECOVER Examples—FMS PROC](#) (see page 661)
[BACKUP Examples](#) (see page 671)
[Incremental BACKUP Examples](#) (see page 672)
[LISTD/LISTV Examples](#) (see page 673)
[IXMAINT Examples](#) (see page 674)
[MERGE Examples](#) (see page 674)
[REBUILD Examples](#) (see page 676)
[IXUPDATE Examples](#) (see page 676)
[Idle Space Release Examples](#) (see page 678)
[Sequential Migrate Examples](#) (see page 679)
[PDS Compression Examples](#) (see page 679)
[Move/Copy Examples](#) (see page 681)
[DASD BILLING Examples](#) (see page 682)
[XCOPY Example](#) (see page 684)

REPORT Examples

The REPORT group consists of the following four examples.

Example 1—Multiple Volume Data Set

Produce the multiple volume data set **DICTIONARY** for all volumes beginning with the letters **TSO**.

```
//jobname JOB (acct,info),etc.  
//          EXEC DMS  
//SYSIN    DD *  
SCAN REALVOLS  
    SELECT VOL=TSO/  
    REPORT MVD
```

Example 2—Use of DSNNAME Parameter

Another example illustrates the use of the DSNNAME keyword parameter where the Single Volume Dictionary is desired for all data sets beginning with the letters PERS on volume PRV002. The names of all the data sets appearing in the report are also to be saved in sequential data set LABS.MJB.DSNLIST.

```
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD *
SCAN REALVOLS
          SELECT VOL=PRV002,DSN=PERS/
          REPORT SVD,CMDGENDSN=LABS.MJB.DSNLIST
```

Example 3—Find Selected Proclib Data Sets

Scan all of the online volumes to find all proclib data sets, as identified by their data set names. Produce both the MVD report data for these data sets and a cross-reference of the procs that are duplicates (same names) in two or more of the libraries.

```
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD *
SCAN REALVOLS
          SELECT VOL=/,DSN=(!PROCLIB!)
          REPORT MVD,P02,COMPONENTS
```

This list indicates that any data set name that ends with *PROCLIB* is to be processed. Since only non-VSAM data sets are being searched for, the COMPONENTS parameter was added to speed up processing.

Example 4—Compare Member Names

Compare the member names from MASTER.SOURCE and MASTER.LOADLIB and identify any member found in one library but not the other.

```
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD *
FIND DSN=MASTER.SOURCE
FIND DSN=MASTER.LOADLIB
REPORT P01
      or
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD *
FIND DSN=MASTER.SOURCE
REPORT P01
FIND DSN=MASTER.LOADLIB
REPORT P01
```

VREPORT Examples

The VREPORT group consists of the following example.

Example—Produce a Volume Allocation Summary

Produce the volume allocation summary for all volumes beginning with letters TSO.

```
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD *
SCAN REALVOLS
  SELECT VOL=TSO/
  VREPORT ALL
```

ARCHIVE Examples

The ARCHIVE group consists of the following six examples.

Example 1—Archive Selected Data Sets

Archive data sets JAN.STATS and FEB.STATS. They should be scratched and uncataloged at the time of archival. The archive copies are to be kept for 60 days.

```
//jobname JOB (acct,info),etc.  
//          EXEC DMS  
//SYSIN     DD *  
FIND DSN=FEB.STATS  
FIND DSN=JAN.STATS  
        ARCHIVE RETPD=60
```

Scratch and uncatalog are default actions. Note that these commands will result in a tape being mounted and the two named data sets being written to it (and indexed in the CA Disk files).

Example 2—Defer the Archive of Data Sets

Data sets JAN.STATS and FEB.STATS are to be archived, but they should be placed in the archive queue to be processed together with other archive requests. They are to be scratched and uncataloged at the time of archival. The archive copies are to be kept for 60 days.

```
//jobname JOB (acct,info),etc.  
//          EXEC ARCHIVE  
//SYSIN     DD *  
        DARCHIVE DSN=JAN.STATS,RETPD=60  
        DARCHIVE DSN=FEB.STATS,RETPD=60
```

Scratch and uncatalog are default actions. Note that these commands will result in no tape mount. The requests are merely placed in the archive queue in the CA Disk FILES.

Example 3—Create Backup Copies

Create backup copies of DAILY.RECS and WEEKLY.SUMMARY. Place the requests in the queue to be processed with other requests. The backup copies need to be kept only 10 days. NOScratch and NOCatalog are both specified since backup copies are desired.

```
//jobname JOB (acct,info),etc.  
//          EXEC ARCHIVE  
//SYSIN     DD *  
        DARCHIVE DSN=DAILY.RECS,NOS,NOC,RETPD=10  
        DARCHIVE DSN=WEEKLY.SUMMARY,NOS,NOC,RETPD=10
```

Example 4—Dispose of Queued Requests

Process all of the requests that have been placed in the archive queue.

```
//jobname JOB (acct,info),etc.  
//          EXEC ARCHIVE,PARM.ARCHIVE=ADSDM276
```

No input stream is used, since the requests come from the archive queue in the CA Disk FILES.

Example 5—List the ARCHIVE Request Queue

Find the current status for all archive requests that were made previously for data sets that begin with the name of DEPT20.

```
//jobname JOB (acct,info),etc.  
//          EXEC LISTREQ  
//SYSIN     DD *  
            LISTREQ DSN=DEPT201/,ARCHIVE
```

A status of PEND (pending), COMP (complete) or FAIL (failed) will be displayed, along with other related data, for each Archive/Backup request found in the queue.

Example 6—Delete Entry from ARCHIVE Queue

An archive request was placed in the archive queue, but the wrong data set name was specified. Delete the entry from the queue.

```
//jobname JOB (acct,info),etc.  
//          EXEC DERASE  
//SYSIN     DD *  
            DERASE DSN=THE.WRONG.NAME,ARCHIVE
```

The ARCHIVE parameter instructs the delete to be made from the archive queue. (The same command, but specifying RESTORE instead of ARCHIVE, directs the delete to the restore queue.)

RESTORE Examples

The RESTORE group consists of the following ten examples.

Example 1—Restore a Data Set

Data set MUST.HAVE was accidentally destroyed. Restore the most current copy of the data set to the volume where the damaged copy now resides, deleting the damaged copy.

```
//jobname JOB (acct,info),etc.
//          EXEC RESTORE
//SYSIN     DD *
           RESTORE DSN=MUST.HAVE,SCR
```

The SCRATCH parameter instructs CA Disk to scratch the existing copy of the data set. By default, the data set will be restored to the volume to which it is cataloged, or if not cataloged, to its original volume. This request is not placed in the restore queue, but is executed immediately. To cause the request to be placed in the queue to be processed with other requests, simply change the command from RESTORE to DRESTORE.

Example 2—Restore a Data Set from a Previous Copy

Restore data set A.B. Do not use the most current backup version, but the previous version. Give the restored data set the name OLD.A.B, and restore it to any of the volumes defined to the pool of volumes known as WORKPOOL.

```
//jobname JOB (acct,info),etc.
//          EXEC RESTORE
//SYSIN     DD *
           RESTORE DSN=A.B,VER=-1,NEW=OLD.A.B,POOL=WORKPOOL
```

The WORKPOOL of volumes must have been defined previously to CA Disk. A sample definition for this pool and another named PRIVPOOL is as follows. (The pool volumes will be used only if the new data set name is not currently cataloged. If it is, the cataloged volume will be used.)

Entries in PARMLIB member DASDPOOL to define the two pools:

```
'WORK01WORKPOOL '
'WORK02WORKPOOL '
'WORK03WORKPOOL '
'WORK04WORKPOOL '
'WORK05WORKPOOL '
'PRIV01PRIVPOOL '
'PRIV02PRIVPOOL '
```

Example 3—Restore Two VSAM Clusters

Restore two VSAM clusters. The catalog being restored to is update password-protected, as is one of the clusters. Assign a new name to the other cluster and use the current date as its creation date. The cluster that is not being renamed is currently defined in the catalog, so instruct CA Disk to delete its old version prior to restore.

```
//jobname JOB (acct,info),etc.
//          EXEC RESTORE
//SYSIN     DD  *
PASSWORD CATALOG=VOL05 USERCAT,MASTER=UPDPASS
RESTORE DSN=DMS.TJP.ICFTEST.CLUSTER,PASSWORD=ICFTEST,SCR
RESTORE DSN=DMS.TJP.VSAM,NEWNAME=DMS.TJP.NEWSAM,CREATE
```

Example 4—Restore to a Different Catalog

Restore a VSAM cluster that was originally defined in a VSAM catalog into an ICF catalog. Change the volume to which it is being restored.

If the high level index in the master catalog has already been updated to point to the new ICFCAT1 catalog, all that is needed is the following command.

```
//jobname JOB (acct,info),etc.
//          EXEC RESTORE
//SYSIN     DD  *
RESTORE DSN=DMS.TJP.VSAM,VOLUME=VOL075
```

If the master catalog still relates the index to the VSAM catalog, you must supply the STEPCAT dd statement for the new catalog when you restore the cluster. Specifying the CATALOG= parameter on the RESTORE command is optional, since the presence of the STEPCAT will direct it there anyway.

```
//jobname JOB (acct,info),etc.
//          EXEC RESTORE
//STPCAT    DD  DISP=SHR,DSN=ICFCAT1
//SYSIN     DD  *
RESTORE DSN=DMS.TJP.VSAM,VOLUME=VOL075,CATALOG=ICFCAT1
```

Example 5—Restore a VSAM Cluster and Alternate Indexes

Restore a VSAM cluster and all of its alternate indexes. Rebuild the alternate indexes.

```
//jobname JOB (acct,info),etc.
//          EXEC RESTORE
//SYSIN     DD  *
RESTORE DSN=DMS.TJP.BASE.CLUSTER,AIXDEF,BLDINX
```

Example 6—Restore a VSAM Cluster and Change Control Interval

Restore a VSAM cluster and change its control interval size. Also, restore its two alternate indexes with different space allocations than they had originally.

```
//jobname JOB (acct,info),etc.
//          EXEC RESTORE
//SYSIN    DD *
RESTORE DSN=DMS.TJP.BASE.CLUSTER,CISZ=4096,
        NOAIXDEF                                don't define the alternate indexes
RESTORE DSN=DMS.TJP.BASE.CLUSTER,
        AIXNAME=DMS.TJP.AIX.NAME1,              AIX's name
        ALLOC=(TRK,5,2)                          new space allocation
RESTORE DSN=DMS.TJP.BASE.CLUSTER,
        AIXNAME=DMS.TJP.AIX.NAME2,              AIX's name
        ALLOC=(CYL,2,0)                          new space allocation
```

Example 7—Restore a Base Cluster to a New Name and Password

Restore a base cluster that was archived with a master password of ORIGMSTR. Change the base cluster name and assign a new master password of NEWMSTR. Restore its alternate index with a new name also. Also change the data and index component names for the alternate index.

```
RESTORE DSN=DMS.TJP.BASE.CLUSTER,
PASSWORD=ORIGMSTR,                                needed to read archive tape
        NEW=DMS.TJPNEW.BASE.CLUSTER,             new base name
        MASTERPW=NEWMSTR,                         new master password for base
        NOAIXDEF                                  don't define the alternate index
RESTORE DSN=DMS.TJP.BASE.CLUSTER,
PASSWORD=ORIGMSTR,                                needed to read archive tape
        RELATE=DMS.TJPNEW.BASE.CLUSTER,          new base name
        RELPSWD=NEWMSTR,                         new master password for base
        AIXNAME=DMS.TJP.AIX.NAME1,               AIX's original name
        NEW=DMS.TJPNEW.AIX.NAME1,                new AIX name
        DNAME=$.DATA,                             new data component name
        INAME=$.INX                               new index component name
```

Example 8—Process Queued Requests

Process all of the requests that have been placed in the restore queue.

```
//jobname JOB (acct,info),etc.
//          EXEC RESTORE,PARM=RESTORE=ADSDM279
```

No input stream is used, since the requests come from the restore queue in the CA Disk FILES.

Example 9—List the RESTORE Request Queue

Find the current status for all restore requests that were made previously for data sets that begin with the name of DEPT20

```
//jobname JOB (acct,info),etc.  
//          EXEC LISTREQ  
//SYSIN     DD *  
            LISTREQ DSN=DEPT201/,RESTORE
```

A status of PEND (pending), COMP (complete) or FAIL (failed) will be displayed, along with other related data, for each restore request found in the queue.

Example 10—Delete Entry from RESTORE Queue

A restore request was placed in the restore queue but the wrong data set name was specified. Delete the entry from the queue.

```
//jobname JOB (acct,info),etc.  
//          EXEC DERASE  
//SYSIN     DD *  
            DERASE DSN=THE.WRONG.NAME,RESTORE
```

The RESTORE parameter instructs the delete to be made from the restore queue. (The same command, but specifying ARCHIVE instead of RESTORE, directs the delete to the archive queue.)

Volume—Level BACKUP Examples

The Volume-Level BACKUP group consists of the following two examples.

Example 1—Volume-Level Backup with shared ENQs

Select volume "PROD01" for volume-level backup processing. Perform shared ENQs on all data sets in the volume but do not perform a RESERVE on the volume itself. Request a retention period of 30 days and create a MAP report of all tracks backed up in data set name sequence.

```
//jobname JOB (acct,info),etc.  
//          EXEC DMS  
//SYSIN     DD *  
SCAN REALVOLS  
  SELECT VOL=(PROD01)  
  VBACKUP DSNENQ=SHR,RESERVE=NO,RETPD=00030,RPT=MAP
```

Example 2—Volume-Level Backup with no ENQ Processing

Select volume *PROD01* for volume-level backup processing. Request no ENQ processing and put a RESERVE on the volume selected. Request an expiration date of 1999365 and do not create a MAP report. Also, backup only those tracks that are actually used, not all tracks that are allocated.

```
//jobname JOB (acct,info),etc.  
//          EXEC DMS  
//SYSIN     DD *  
SCAN REALVOLS  
          SELECT VOL=(PROD01)  
          VBACKUP DSNENQ=NONE,RESERVE=YES,EXPDT=1999365,RPT=NO,LSTAR
```

Volume—Level RECOVER Examples

The Volume-Level RECOVER group consists of the following two examples.

Example 1—Volume-Level Recovery, No Enqueues or Reserve

Selects volume *TS0006* for volume-level recovery processing. Perform no enqueue or reserve during volume-level recovery. After volume-level processing has completed successfully, all data set modified (or created) after the last volume-level backup are recovered using data set-level processing.

```
//jobname JOB (acct,info),etc.  
//          EXEC RECOVER  
//SYSIN     DD *  
VRECOVER VOL=TS0006,DSNENQ=NONE,RESERVE=NO,UPDATE
```

Example 2—Volume-Level Recovery from Specified Tracks

Select volume *PROD01* for volume-level recovery processing. Recover the tracks starting with cylinder X'0115', head X'0005' and ending with cylinder X'0115', head X'001D'.

```
//jobname JOB (acct,info),etc.  
//          EXEC RECOVER  
          VRECOVER VOL=PROD01,ABS=(01150005-0115001D)
```

Volume Defragmentation Examples

Disk volumes can become fragmented to the degree that even though there are many free tracks available, requests to allocate new data sets will still fail. This is caused by the free space being in such small extents that it is unusable; that is, a track here, another there, but 15 tracks together can't be found.

Due to the lack of disk management tools in the past, the only recourse seemed to be to *reorganize* the pack, now commonly known as *defragging* it. Although there are times when this is needed, CA Disk provides other alternatives that can solve the real problem with much less system impact.

Why Space Allocations Fail

The real problem is that space requests are failing. The term *fragmented* just explains why the requests are failing at a point in time. Why not just free up more space? A defrag operation usually means locking up the volume for several minutes, which often means running your entire machine in *stand-alone* mode as well. This is pretty severe just to gain some free space.

Solution—Release Overallocated and Unused Space

The simplest solution in many cases is to release the overallocated and unused space in the other data sets on the volume. This quite likely will actually increase the degree of free space fragmentation, but it can also provide much larger areas of contiguous free space, which solves the problem. This function does not require exclusive use of the volume (let alone the entire machine), and is very fast. If a volume defrag is still needed, having released the idle space beforehand provides an even greater return from the defrag run.

Solution—Move and Recatalog

Another alternative is to move some of the data sets off to another volume and recatalog them. The Move/Copy function can do this with just one command. Data sets being used are skipped. The ones that are moved are also compressed, improving them as well as the target volume, and as they are scratched from the source volume, free space is made available. Again, exclusive control of only the data sets is needed, one at a time, not of the entire volume or machine.

Solution—Defragmentation with ARCHIVE and RECOVER

Although the previous options are highly recommended as first alternatives (and you can be able to think of several others), the point can still be reached when defragging the volume must be done. The following comments outline a method to do this with the ARCHIVE and RECOVER commands, as well as providing some added benefits and tips to keep in mind.

To CA Disk, a volume defrag means the following:

1. Consolidate used tracks to obtain large contiguous free space extents.
2. Combine multiple extent data sets into a single extent. Compress PDS data sets.
3. Relocate and reorganize ISAM data sets.
4. Reorganize VSAM clusters.
5. Compress the VTOC.

A recommended approach to doing this with CA Disk is as follows:

1. If using an FDS, allocate and format a new *mini* FDS.
2. The advantage of using a separate files data set for this special function is that it keeps these reorganization index entries separate from the Archive/Backup entries, and reduces the time needed to process the files data set, since it contains fewer entries.
3. Establish a JCL procedure that uses it in a DSCL run that archives and scratches all data sets on the pack. Follow this with a RECOVER command to restore all of the data sets on the volume.

The following JCL can be used as a guideline for *defragging* a disk volume.

- First, create the FDS to be used.

```
//jobname JOB (acct,info),etc.
//INIT EXEC PGM=ADSMI002,PARM=ADSDM100
//STEPLIB DD DISP=SHR,DSN=CAI.DISK.CCUWLOAD
//ABNLDUMP DD DUMMY
//CMDPRINT DD SYSOUT=A
//FILES DD DSN=CAI.DISK.DEFRAG.volser.FILES
// DISP=(,CATLG,DELETE),UNIT=SYSDA,
// DCB=(DSORG=DA),SPACE=(CYL,2,,CONTIG)
//MSGPRINT DD SYSOUT=A
//PARMLIB DD DISP=SHR,DSN=CAI.DISK.CCUWPARM
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
```

- Second, archive all non-VSAM data sets and VSAM clusters, from the desired volume.

```
//DEFRAGV EXEC DMS
//SYSIN DD *
//FILES DD DISP=SHR,DSN=CAI.DISK.DEFRAG.volser.FILES
SCAN REALVOLS
SELECT VOL=volser
ARCHIVE
```

- Third, recover all the data sets to the volume only if they were scratched successfully at archive time.

```
//RECOVER EXEC FMS
//FILES DD DISP=SHR,DSN=CAI.DISK.DEFRAG.volser.FILES
//SYSPARMS DD *
PREALLOCN no need to restore if not scratched after archive
VSPREDEFN no need to restore if not scratched after archive
//SYSIN DD *
SCAN DSNINDEX
SELECT DSN=dsnames
RECOVER TOVOL=volser
```

RECOVER Examples—FMS PROC

The RECOVER group consists of the following 16 examples.

Example 1—Recover Between Specified Dates

Recover between certain "date" criteria.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
      SELECT DSN=DISK820.BASE./,
             CRITERIA=(ARCDT,GT,TODAY-5,AND,ARCDT,LT,TODAY)
RECOVER POOL=MYPPOOL,DISP=RECATALOG
```

Example 2—Recover Between Specified Times

Recover during different "time" criteria.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
      SELECT DSN=DISK820.BASE./,
             CRITERIA=(ARCDT,EQ,95201,AND,ARCTIME,GT,1200,AND,ARCTIME,LT,2100)
RECOVER POOL=MYPPOOL,DISP=RECATALOG
```

Example 3—Recover Data Sets from Pseudo Volume

Recover of archive data sets cataloged to the CA Disk pseudo volume, allocating them onto volume DMS901, and altering each catalog entry from ARCIVE to DMS901.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
      SELECT DSN=DISK.XYZ./,CRITERIA=(ARCTYPE,EQ,RECAT)
RECOVER TOVOL=DMS901,DISP=RECATALOG
```

Example 4—Recover Data Sets Specifying Location and Portion of the Name

Recover data sets names contained in table FMSINCLD, as well as the data sets beginning with MYHLQ. Exclude data sets with an expiration date in the Format-1 DSCB if the date is less than 10 days from today.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
  SELECT TABLE=FMSINCLD,CRI=(ARCDT,NE,TODAY),VOL=(SSL/,SMS/)
  SELECT DSN=MYHLQ./
  EXCLUDE CRI=(EXPDT,LT,TODAY+10)
  RECOVER
```

Example 5—Recover Data Sets Specifying Volume Patterns and Locations

Recover data sets from the volume patterns SSL/ and SMS/ as long as they are located in table FMSINCLD, but not in table FMSEXCLD. Scratch any of the data sets if they already exist.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
  SELECT TABLE=FMSINCLD,VOL=(SSL/,SMS/)
  EXCLUDE TABLE=FMSEXCLD
  RECOVER SCRATCH
```

Example 6—Recover Current Backups by Name

Recover (and if necessary, scratch) all current backups of clusters located in an ARCHVOL named DISK820.DMSN.C1995306.T084345 and cataloging them into an alternate catalog.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
  SELECT TDSN=DISK820.DMSN.C1995306.T084345,
  CRI=(DSORG,EQ,AM,AND,ARCTYPE,EQ,BACKUP,AND,VERSION,EQ,0)
  RECOVER SCRATCH,TOCAT=ICFCAT.BACKUP
```

Example 7—Recover Data Sets Specifying Volume Patterns

Recover (and if necessary, scratch) all data sets from all the SSL8/ volume patterns, excluding volume SSL805. Also recover all catalogs except those backed up from the SMS/ volume pattern.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD *
SCAN DSNINDEX
  SELECT DSN=/,VOL=(SSL8/)
  EXCLUDE VOL=SSL805
  SELECT CAT=/,VOL=(/)
  EXCLUDE CAT=/,VOL=SMS/
  RECOVER SCRATCH
```

Example 8—Recover Selected Data Sets to a Specified Volume

Select all current versions of PDSE data sets that are on the VTOC image of SSL805, and recover them (and if necessary, scratch them) to volume SSL801.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD *
SCAN DSNINDEX
  SELECT VTOC,VOL=SSL805,CRI=(DSORG,EQ,P0,OR,DSORG,EQ,POE)
  EXCLUDE CRI=(ARCTYPE,NE,BACKUP,AND,VERSION,NE,0)
  RECOVER SCRATCH,TOVOL=SSL801
```

Example 9—Recover Selected Data Sets Archived Yesterday

Recover those data sets that were archived yesterday, excluding those data sets in FMSEXCLD. Any data set that happens to already exist will be left unaltered because sysparms PREALLOC and VSPREDEF both default to 'N'.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD *
SCAN DSNINDEX
  SELECT DSN=/,CRI=(ARCDT,EQ,TODAY-1,AND,ARCTYPE,EQ,ARCHIVE)
  EXCLUDE TABLE=FMSEXCLD
  RECOVER
```


Example 10—Recover Data Sets with Specified Criteria

Recover data sets based on the following criteria:

- Non-VSAM
- Begin with the pattern TSO
- Originally allocated in tracks
- The most current version
- Were archived and recataloged to the pseudo volume in the last 14 days
- Exclude data sets contained in table FMSEXCLD
- Anything that was already restored today will be excluded

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
SELECT DSN=TSO/,CRI=(ALLOTRKS,EQ,YES,AND,
        ARCDT,GT,TODAY-14,AND,VERSION,EQ,0,AND,
        ARCTYPE,EQ,RECAT,AND,DSORG,NE,AM)
EXCLUDE TABLE=FMSEXCLD
EXCLUDE CRI=(RESDT,EQ,TODAY)
RECOVER
```

Example 11—Recover Data Sets with Specified Criteria

Recover data sets based on the following criteria:

- Non-VSAM
- Begin with the HLQ of ISPGLH1
- Were archived since yesterday at 08:43
- Exclude those data sets that have a F1EXPDT that is set to expire in the next 30 days
- Recover them to the HLQ of SBGLH

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
SELECT DSN=ISPGLH1./,CRI=(ARCDT,EQ,TODAY-1,AND,
        ARCTIME,GT,0843,AND,ARCTYPE,EQ,ARCHIVE,AND,DSORG,NE,AM)
SELECT DSN=ISPGLH1./,
        CRI=(ARCDT,EQ,TODAY,AND,ARCTYPE,EQ,ARCHIVE,AND,DSORG,NE,AM)
EXCLUDE CRI=(EXPDT,LE,TODAY+30)
RECOVER NEWHLQ=SBGLH
```

Example 12—Recover Archived Data Sets and Catalogs

Recover (and if necessary, scratch) all data sets and catalogs archived to ARCHVOL @85347, excluding:

- Data sets listed in FMSEXCLD
- Catalog MASTER.CAT

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
SELECT DSN=/,CAT=/,CRI=(DSNKEY,EQ,<@85347>)
EXCLUDE TABLE=FMSEXCLD
EXCLUDE CAT=MASTER.CAT
RECOVER SCRATCH
```

Example 13—Recover with Tape I/O Error Abend

During the execution of the following command, you received a tape I/O error on tape 123456 which caused the Recover to abend:

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
      SELECT VOL=SSL805,CRI=(ARCTYPE,EQ,BACKUP,AND,VERSION,EQ,0)
      RECOVER SCRATCH
```

Two items must be completed before restarting the Recover:

Exclude the ARCHVOL that caused the abend.

Avoid recovering data sets that had previously been recovered in the first job.

To do this, submit the following JCL:

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
      SELECT VOL=SSL805,CRI=(ARCTYPE,EQ,BACKUP,AND,VERSION,EQ,0)
      EXCLUDE CRI=(DSNKEY,EQ,<123456>)
      EXCLUDE CRI=(RESDT,EQ,TODAY)
      RECOVER SCRATCH
```

Example 14—Restarting a Recovery

During the execution of the following command, an IPL occurred due to a system failure:

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
  SELECT VOL=PROD/,CRI=(DSORG,EQ,AM,AND,ARCTYPE,EQ,BACKUP,
    AND,ARCDT,GT,TODAY-7),DSN=PROD./,VTOCS
  SELECT VOL=TSO/,CRI=(DSORG,NE,AM,AND,ARCTYPE,EQ,BACKUP,
    AND,ARCDT,GT,TODAY-7),DSN=TSO./,VTOCS
  SELECT TABLE=PAYROLL1,CRI=(DSORG,EQ,AM,AND,
    ARCTYPE,EQ,BACKUP,AND,ARCDT,GT,TODAY-7),VTOCS
  SELECT CAT=USER/,VOL=(PROD/,TSO/)
  RECOVER SCRATCH
```

You need to restart the Recover as soon as the IPL is complete. Use the following JCL to restart this job and finish recovering the following criteria:

- The most current version of all VSAM clusters with the HLQ of PROD/, that were backed up from the PROD/ volume pattern in the last seven days, and were in the last VTOC backup.
- The most current version of all non-VSAM data sets backed up with the HLQ of TSO/ from the TSO/ volume pattern, that were backed up in the last seven days, and were in the last VTOC backup.
- The most current version of all VSAM clusters contained in the table PAYROLL1 that were backed up in the last seven days.
- All catalogs starting with USER/ from volume patterns PROD/ and TSO/.

```
//jobname JOB (acct,info),etc.
//          EXEC FMS
//SYSIN     DD  *
SCAN DSNINDEX
  SELECT VOL=PROD/,CRI=(DSORG,EQ,AM,AND,ARCTYPE,EQ,BACKUP,
    AND,VERSION,EQ,0,AND,ARCDT,GT,TODAY-7),DSN=PROD./,VTOCS
  SELECT VOL=TSO/,CRI=(DSORG,NE,AM,AND,ARCTYPE,EQ,BACKUP,
    AND,VERSION,EQ,0,AND,ARCDT,GT,TODAY-7),DSN=TSO./,VTOCS
  SELECT TABLE=PAYROLL1,CRI=(DSORG,EQ,AM,AND,VERSION,EQ,0,
    AND,ARCTYPE,EQ,BACKUP,AND,ARCDT,GT,TODAY-7)
  SELECT CAT=USER/,VOL=(PROD/,TSO/)
  EXCLUDE CRI=(RESDT,EQ,TODAY)
  RECOVER SCRATCH
```

Example 15—Plan Disaster Recovery

Plan in advance for disaster recovery purposes, the following command will assist you in knowing what resources will be required if you ever need to recover the XYZ application:

```
//jobname JOB (acct,info),etc.  
//          EXEC FMS  
//SYSIN    DD *  
           SET MODE=SIM  
           SCAN DSNINDEX  
           SELECT DSN=XYZ/  
           RECOVER REPORT
```

Three reports will be generated indicating every resource required to recover the selected data sets. See the description of the parameter REPORT in the section RECOVER Command Syntax in the chapter "RESTORE/RECOVER" for samples of these reports.

Example 16—Recovery with GDG and GDGLIMIT

Assume that the GDG base name is A.B, its limit is 3, and there are 5 GDSs in archive (A.B.G001V00, A.B.G002V00,..., A.B.G005V00) and A.B.G0003V00, A.B.G0004V00, and A.B.G0005V00 are cataloged to ARCIVE.

```
//      EXEC FMS
//SYSIN DD *
        SCAN DSNINDEX
        SELECT DSN=A.B/
        RECOVER
```

If you do not specify any GDGLIMIT= parameter, then CA Disk recovers the GDG to its latest status. CA Disk recovers as many generations as possible within the defined limit of the GDG base. In this case, A.B.G0002V00, A.B.G0003V00, and A.B.G0004V00 are recovered.

```
//      EXEC FMS
//SYSIN DD *
        SCAN DSNINDEX
        SELECT DSN=A.B/
        RECOVER GDGLIMIT=99
```

If the value of GDGLIMIT= is greater than the defined limit of the GDG base, CA Disk recovers as many generations as possible within the defined limit. In this case, A.B.G0002V00, A.B.G0003V00, and A.B.G0004V00 are recovered.

```
//      EXEC FMS
//SYSIN DD *
        SCAN DSNINDEX
        SELECT DSN=A.B/
        RECOVER GDGLIMIT=2
```

If the value of GDGLIMIT= is less than the defined limit of the GDG base, CA Disk recovers as many generations as possible within the value specified for GDGLIMIT=. In this case, only A.B.G0003V00 and A.B.G0004V00 are recovered.

```
//      EXEC FMS
//SYSIN DD *
        SCAN DSNINDEX
        SELECT DSN=A.B/
        RECOVER GDGLIMIT=0
```

CA Disk does not recover any GDS for GDG base A.B. In this case, no GDS is recovered.

Assume that the GDG base name is A.B, its limit is 3, and there are 5 GDS's in archive (A.B.G0001V00, A.B.G0002V00,..., A.B.G0005V00) and A.B.G0003V00, A.B.G0004V00, and A.B.G0005V00 are active GDGs cataloged to MIGRAT while A.B.G0001V00, and A.B.G0002V00 are rolled off cataloged to MIGRAT. The cataloged rolled-off generations are not counted as part of the limit since they are no longer part of the sphere and are always recovered.

```
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
SELECT DSN=A.B/
RECOVER
```

If no GDGLIMIT= parameter is specified, then CA Disk recovers the GDG to its latest status. CA Disk recovers as many generations as possible within the defined limit of the GDG base. In this case, all five generations are recovered.

```
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
SELECT DSN=A.B/
RECOVER GDGLIMIT=99
```

If the value of GDGLIMIT= is greater than the defined limit, CA Disk recovers as many generations as possible within the defined limit of the GDG base. In this case, all five generations are recovered.

```
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
SELECT DSN=A.B/
RECOVER GDGLIMIT=2
```

If the value of GDGLIMIT= is less than the defined limit, CA Disk recovers as many generations as possible within the value that you specify for GDGLIMIT=. In this case, only A.B.G0004V00 and A.B.G0005V00 are recovered with the rolled-off generations.

```
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
SELECT DSN=A.B/
RECOVER GDGLIMIT=0
```

CA Disk does not recover any GDS for GDG base A.B. In this case, no GDS is recovered except for the rolled-off generations.

BACKUP Examples

The BACKUP group consists of the following two examples.

Example 1—Typical Backup Examples

To illustrate the flexibility of DSCL BACKUP processing, some examples of typical use are provided.

The first example provides the command and entries necessary to accomplish the following:

- Archive, scratch and uncatalog all data sets on volume TEMPVS that are 14 days old.
- Scratch and uncatalog all data sets on all volume serials beginning with the letters DATA that are either uncataloged or have not been used in 30 days.
- Scratch and uncatalog all data sets on any volume being scanned that have not been used in 360 days.
- Test the actual run in simulation mode.

```
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD *
SET MODE=SIMULATE
SCAN REALVOLS
SELECT VOL=TEMPVS,CRI=(USED,LT,TODAY-14)
        ARCHIVE
SELECT VOL=DATA/,CRI=(CATSTAT,NE,YES)
SELECT VOL=DATA/,CRI=(USED,LT,TODAY-30)
SELECT CRI=(USED,LT,TODAY-360)
        DELETE
```

- The SET MODE=SIMULATE command is used to cause processing to be simulated. It can easily be removed after it has been verified that the desired results are being obtained.
- The VOL parameters are used to confine the processing to volume serials beginning with the letters specified.

Example 2—Backup Specifying Beginning Letters of Data Set Names

This example is provided to illustrate the requirements for the following types of control.

- Scratch and uncatalog on all volumes that have serial numbers beginning with the letters *TSO* those data sets that are uncataloged and have not been used in 45 days.
- Archive, scratch, and uncatalog on all volumes that have serial numbers beginning with the letters *USER* all data sets that are uncataloged, have no expiration date, and have not been used in 90 days.
- The DSCL commands required to accomplish the previous is:

```
//jobname JOB (acct,info),etc.  
//          EXEC DMS  
//SYSIN    DD *  
SET MODE=SIMULATE  
SCAN REALVOLS  
SELECT CRI=(CATSTAT,NE,YES,AND,USEDT,LT,TODAY-45),VOL=TSO/  
DELETE  
SELECT VOL=USER/,CRI=(CATSTAT,NE,YES,AND,EXPDT,EQ,0,AND,  
USEDT,LT,TODAY-60)  
ARCHIVE
```

Incremental BACKUP Examples

The Incremental BACKUP group consists of the following example.

Daily Incremental Backup Example

To do incremental backup, one can use a daily job to capture all of the changed data sets, and a second weekly job to get backup copies of all the data that has not changed during the week. Full pack dumps are never needed again.

Monday Through Saturday Runs

```
//jobname JOB (acct,info),etc.  
//          EXEC DMS  
//SYSIN    DD *  
SCAN REALVOLS  
SELECT CRITERIA=(MODIFIED,EQ,YES)  
BACKUP RETPD=15
```


Sunday Run

```
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD  *
SCAN REALVOLS
  SELECT CRITERIA=(MODIFIED,EQ,YES)
  SELECT CRITERIA=(ARCDT,LE,TODAY-7)
  BACKUP RETPD=15
```

LISTD/LISTV Examples

The LISTD/LISTV group consists of the following four examples.

Example 1—List All Archived Data Sets on a Tape

List all of the archived data sets contained on tape DMS10

```
//jobname JOB (acct,info),etc.
//          EXEC LISTD
//SYSIN     DD  *
LISTD VOL=DMS105
```

Example 2—List Seleted Archived Data Sets

List all of the archived data sets that contain either the name WOOD or BIRCH as the second-level qualifier.

```
//jobname JOB (acct,info),etc.
//          EXEC LISTD
//SYSIN     DD  *
LISTD DSN=(*.WOOD./,*.BIRCH./)
```

Example 3—List Archived Data Sets From from Archive-to-Disk and Tape

List all of the archive data sets, both from archive-to-disk and archive tapes.

```
//jobname JOB (acct,info),etc.
//          EXEC LISTD
//SYSIN     DD  *
LISTV
```

Example 4—List Selected Archive-to-Disk and Archive Tape Data Sets

List all of archive tapes and archive-to-disk data sets that begin with the name of DMS.BACKUPS.

```
//jobname JOB (acct,info),etc.  
//          EXEC LISTD  
//SYSIN    DD *  
          LISTV DSN=DMS.BACKUPS/
```

IXMAINT Examples

The IXMAINT group consists of the following two examples.

Example 1—Delete Expired Data Sets

Delete all of the expired data sets from the CA Disk archives. If all of the data sets on any given archive volume (an archive tape or an archive data set on disk) have expired, delete the tape or disk data set from the archives as well.

```
//jobname JOB (acct,info),etc.  
//          EXEC IXMAINT  
//SYSIN    DD *  
          DSNDELETE
```

Example 2—Delete Specified Archive Tape and Data Sets

Delete archive tape DMS002 and all of the archived data sets contained on it, regardless of their expiration dates.

```
//jobname JOB (acct,info),etc.  
//          EXEC IXMAINT  
//SYSIN    DD *  
          PURGE VOL=DMS002
```

MERGE Examples

The MERGE group consists of the following three examples.

Example 1—Merge Selected Archive Tapes

Merge the CA Disk archive tapes (but not the backup tapes) as identified by names that start with DMS.ARCH. Merge only those tapes that are less than 50 percent full of unexpired data. Also, generate a tape pull list for operations that can be used to find the needed tapes prior to actually starting the merge run. Create a duplicate copy of each new tape.

First, allocate the *pull-list* data set if not previously done.

```
//jobname JOB (acct,info),etc.
//ALLOC EXEC PGM=IEFBR14
//TAPEPULL DD DSN=SAMS.DISK.MERGE.TAPES,DISP=(NEW,CATLG),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//          SPACE=(TRK,(1,1)),UNIT=SYSDA
```

To generate the tape pull list:

```
//TAPEMERG EXEC MERGE
//MERGE.TAPEPULL DD DISP=OLD,DSN=SAMS.DISK.MERGE.TAPES
//MERGE.SYSIN DD *
MERGE SIM,TDSN=DMS.ARCH/,PERCENT=50,TAPEPULL=GEN,TYPE=TAPE
```

Note: SIMULATE is used since generation of the pull list is the only processing required at this time.

To use the generated list of tapes and begin the actual merge: (System defaults will dynamically allocate one input drive and two output drives, one for the primary and one for the copy.)

```
//jobname JOB (acct,info),etc.
//          EXEC MERGE
//MERGE.TAPEPULL DD DISP=OLD,DSN=SAMS.DISK.MERGE.TAPES
//MERGE.SYSIN DD *
MERGE TDSN=DMS.ARCH/,PERCENT=50,TAPEPULL=USE
```

Example 2—Merge Disk Archives Specifying Percent Full

Merge the disk archives that are less than 90 percent full of unexpired data. Keep the primary copy of the merged output on disk, a duplicate copy, and a tertiary copy on tape.

```
//jobname JOB (acct,info),etc.
//          EXEC MERGE
//MERGE.SYSIN DD *
MERGE SIM,TDSN=DMS.ARCH/,PERCENT=90,TYPE=DISK
//SYSPARMS DD *
MERP1TYPDISK overrides the default of tape
MERC1TYPTAPE the default, but shown for illustration
MERT1TYPTAPE overrides the default of NULL
```

Example 3—Merge Selected Disk Archives to Tape

Merge some of the archives residing on disk volumes off to tape, making more room on disk for "new" archives. Select only the "old" archive data sets; that is, those that have been there for 60 days or more. Create a duplicate backup tape of each new archive tape.

```
//jobname JOB (acct,info),etc.  
//          EXEC MERGE  
//MERGE.SYSIN DD *  
MERGE TYPE=DISK,DAYSOLD=60
```

REBUILD Examples

The REBUILD group consists of the following two examples.

Example 1—Rebuild an Archive Index for a Tape

Rebuild the archive index entries for tape DMS544; that is, the index entry for the tape volume and the entries for all of the data sets contained on the tape. The name in the tape label for this tape is SAMS.DISK.ARCHIVES.G0032V00.

```
//jobname JOB (acct,info),etc.  
//          EXEC REBUILD,DSN='SAMS.DISK.ARCHIVES.G0032V00'
```

Example 2—Rebuild an Archive Index for a Specified Archive Data Set

Rebuild the archive index entries for the archive data set SYS.ARC.SAMS.DISK.D15APR85.T142003 that resides on disk volumes PACK01 and PACK14.

```
//jobname JOB (acct,info),etc.  
//          EXEC REBUILD,DSN='SYS.ARC.SAMS.DISK.D15APR85.T142003',
```

IXUPDATE Examples

The IXUPDATE group consists of the following five examples.

Example 1—Delete Archive Index Entries

Delete all of the archive index entries for data set DEPT75.JOHN.TESTDAT1.

```
//jobname JOB (acct,info),etc.  
//          EXEC IXUPDATE  
//SYSIN     DD *  
           DELETE DSN=DEPT75.JOHN.TESTDAT1,ALL
```

Example 2—Update the Expiration Date

Update the expiration date for the current archive copy of DEPT75.JOHN from its present value to Jan. 1, 1984.

```
//jobname JOB (acct,info),etc.  
//          EXEC IXUPDATE  
//SYSIN     DD *  
           CHANGE DSN=DEPT75.JOHN,EXPDT=84001
```

Example 3—Disable an Archive tape Index

Since archive tape DMS879 was accidentally written over, disable its index entry such that all restore requests will automatically be directed to its duplicate copy tape.

```
//jobname JOB (acct,info),etc.  
//          EXEC IXUPDATE  
//SYSIN     DD *  
           RESET KEY=DMS879,DISABLE
```

Example 4—Create and Update an Entry for an Archive Tape

Create an ARCHVOLS entry (entry for an archive tape) for tape DMS222 that is named DMS.ARCHCOPY. Then update the entry for DMS221 to indicate DMS222 as its duplicate copy tape.

```
//jobname JOB (acct,info),etc.  
//          EXEC IXUPDATE  
//SYSIN     DD *  
BUILDARC VOL=DMS222,DSN=DMS.ARCHCOPY,TYPE=TAPE,ARCTYPE=COPY,EXPDT=87031  
RESET KEY=DMS221,COPY=DMS222
```

Example 5—Delete a Copy Volume from its Primary Volume

Delete a copy volume from its primary volume.

```
//jobname JOB (acct,info),etc.  
//          EXEC IXUPDATE  
//SYSIN     DD *  
          RESET KEY=prim,COPY=NULL  
          RESET KEY=copy,SCRATCH
```

Idle Space Release Examples

The Idle Space Release group consists of the following three examples.

Example 1—Scan for Overallocated PO Data Sets and Release Part of the Space

Scan all real online volumes and select all PO data sets that are 50 percent overallocated. Release a portion of the space such that only 20 percent of the data set is idle.

```
//jobname JOB (acct,info),etc.  
//          EXEC DMS  
//SYSIN     DD *  
SCAN REALVOLS  
      SELECT CRITERIA=(DSORG,EQ,PO,AND,PCTUSED,LT,50)  
      RELEASE PCTUSED=80
```

Example 2—Scan for Overallocated PO Data Sets and Release All Extra Space

Scan all real online volumes and select all PS data sets that are 5 percent overallocated. Release all of the extra space.

```
//jobname JOB (acct,info),etc.  
//          EXEC DMS  
//SYSIN     DD *  
SCAN REALVOLS  
      SELECT CRITERIA=(DSORG,EQ,PS,AND,PCTUSED,LT,95)  
      RELEASE
```

Example 3—Release Space with Specified Criteria

Release the idle space from those data sets that:

- Reside on real volumes starting with "TSO".
- Have names that start with "ADS" or "PAY".
- Are sequential, or are partitioned and have 20 percent or more idle space.

```
//jobname JOB (acct,info),etc.  
//          EXEC DMS  
//SYSIN     DD  *  
SCAN REALVOLS  
SELECT VOL=TSO/ ,DSN=(ADS/ ,PAY/ ) ,  
CRITERIA=(DSORG,EQ,PO,AND,PCTUSED,LT,80)  
SELECT VOL=TSO/ ,DSN=(ADS/ ,PAY/ ) ,CRITERIA=(DSORG,EQ,PS)  
RELEASE
```

Notice that all sequential data sets will have their idle space released, since no threshold is specified.

Sequential Migrate Examples

The Sequential Migrate group consists of the following example.

Example—Migrate Specifying Part of Name and Generations

Scan all real volumes that start with PLAN and migrate to tape all but the two most current generations of all GDG data sets.

```
//jobname JOB (acct,info),etc.  
//          EXEC MIGRATE  
//SYSIN     DD  *  
MODESET SIMULATE  
SCAN VOL=PLAN/ ,GDGSONLY,GEN=2
```

PDS Compression Examples

The PDS Compression group consists of the following three examples.

Example 1—Compress Specified Data Sets by Percent Full and Number of Extents

Compress the data set named DEVE.TESTLIB, and all PDS data sets on the TSO volumes that are 80 percent full or over, or are in more than 15 extents. Reallocate the TSO PDSs such that they are only 50 percent full. Adjust the directory size such that the directory of each PDS will be 80 percent used.

```
//jobname JOB (acct,info),etc.
//          EXEC COMPRES
//SYSIN     DD *
            COMPRESS DSN=DEVE.TESTLIB,USE
            SCAN VOL=TSO/,THR=(80,15,00),ALL=(50,0,80),SHRINK
```

Example 2—Compress ISPF Profile PDSs and Reallocate

Compress all ISPF profile PDSs that have allocated space (excluding the directory blocks) less than 50 percent used. Reallocate the data sets, such that the primary allocation (excluding the directory blocks) will be 85 percent used, and adjust the directory size such that the directory of each PDS will be 75 percent used.

Note: This example illustrates how to recover additional space from ISPF profile PDSs that are still overallocated due to compression performed using releases prior to 7.7. Some of these data sets occupying two tracks can be further compressed to use only one track.

```
//jobname JOB (acct,info),etc.
//          EXEC COMPRES
//SYSIN     DD *
            SCAN DSN=!ISP,VOL=/,THR=(0,0,0,50),ALL=(85,0,75),SHRINK
```

Example 3—Select Empty PDSs and Reallocate

Select from all WORK volumes only empty PDSs (that is, those having no members). Reallocate them using the minimum number of primary tracks or blocks that are necessary to store the original directory blocks.

```
//jobname JOB (acct,info),etc.
//          EXEC COMPRES
//SYSIN     DD *
            SCAN VOL=WRK/,THR=(0,16),EMPTY,CONVERT
```

With empty PDSs that have a directory larger than one track, you can recover even more space by additionally specifying the SHRINK parameter. This will cause those directories to be reduced to a single track, permitting the minimum possible primary allocation of one track.

Move/Copy Examples

The MOVE/COPY group consists of the following two examples.

Example 1 — Compress by Moving

Compress PDS PROD.TESTLIB by moving it to either PRIV01 or PRIV10.

```
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD *
FIND DSN=PROD.TESTLIB
      MOVE TOV=(PRIV001,PRIV10)
```

Example 2 — Convert a Disk Volume to a New Device Type

Convert a disk volume to a new device type. The recommended way to convert to new volumes and device types is to give the new volumes new names, and to then use the Move/Copy function to move and recatalog the data sets to the new volume.

```
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD *
SCAN REALVOLS
      SELECT VOL=DA3380
      MOVE TOV=DA3390
```

If there is an overriding reason why you must give the new volume the same name as the old volume, it is recommended that you archive, scratch, and uncatalog the data sets from the source volume, then run volume recovery to the new volume that has been given the same name. When CA Disk catalogs the data sets as they are being restored, the new device type will be put into the catalog entry.

```
//jobname JOB (acct,info),etc.
//          EXEC DMS
//SYSIN     DD *
SCAN REALVOLS
      SELECT VOL=MVS001
      ARCHIVE
```

Clip (relabel) the target volume (the 3390) to the same volume serial as the source volume (the 3380). Vary the source volume offline. Mount the target volume online. Then execute the RECOVER command.

```
//          EXEC RECOVER
//SYSIN     DD *
      RECOVER VOL=MVS001,RECATALOG
```

DASD BILLING Examples

Set up DASD Billing for all volumes beginning with the letters WORK, and for volume LABS81. The billing rate for data sets on the WORK packs is \$ 008500 per kilobyte-day. For LABS81 it is \$ 010000 per kilobyte-day. The length of the billing cycle will be monthly and will begin on the first day of every month. In addition, all transient data sets (those that are less than one day old on the volume) will not have charges accumulated for them.

First, establish the two parmlib members to define the volume categories and the billing rates for each.

PARMLIB(DSBCATEG) Contents

```
01WORK/X
02LABS81
```

PARMLIB(DSBRATEC) Contents

01008500

02010000

Second, accumulate the non-VSAM billing records for data sets on these volumes.

```
//jobname JOB (acct,info),etc.
//          EXEC BILLING
//SELECT.SYSIN DD *
SELECT VOLUME=(LABS81,WORK/)
//ACCUM.FILES DD DSN=SAMS.DISK.BILLING.FILES,DISP=SHR
//ACCUM.SYSIN DD *
ACCUMULATE BMONTH,MODE=IMMEDIATE,BYPASSNEW
```

Third, accumulate the VSAM billing records for clusters defined on these volumes.

```
//          EXEC VSAMBILL
//DMS.SYSIN DD *
SCAN CAT=/
EXCLUDE CRITERIA=(DSORG,NE,VSAM)
SELECT VOLUME=(LABS81,WORK/)
BILLING
//ACCUM.FILES DD DSN=SAMS.DISK.BILLING.FILES,DISP=SHR
//ACCUM.SYSIN DD *
ACCUMULATE BMONTH,MODE=IMMEDIATE,BYPASSNEW
```

After the billing period has passed (and several of the previous accumulation jobs have been run), the billing totals are extended, printed, and cleared with the following JCL. This JCL should be executed on the last day of the billing cycle (if the cycle is monthly, it would be run on the 30th or the 31st day of the month, depending what day the month ends).

```
//jobname JOB (acct,info),etc.
//          EXEC EXTEND
//EXTEND.FILES DD DSN=SAMS.DISK.BILLING.FILES,DISP=SHR
//SYSIN DD *
EXTEND VOLUME=(LABS81,WORK/)
```

XCOPY Example

The XCOPY command shown below selects all archive volumes that have not already been processed by XCOPY, and creates a second set of archive and backup data sets. The new backup data sets are stored in an ARCHVOL created in the secondary FILES allocated to the OFILES DD statement.

```
//jobname JOB (acct,info),etc.
//COPYTAPE EXEC XCOPY
//XCOPY.FILES DD DSN=DISK.PRIMARY.FILES,DISP=SHR
//XCOPY.OFILES DD DSN=DISK.OFFSITE.FILES,DISP=SHR
//XCOPY.SYSPARMS DD *
MERP1NAMDMS.MERGPRI1
MERP1TYP3480
MERC1NAMDMS.MERGCOP1
MERC1TYPDISK
//XCOPY.SYSIN DD *
SELECT DSN=(PROD1./,PROD3./)
SELECT DSN=SYSA./
SELECT DSN=PRODACCT.MASTER.LOADLIB
SELECT DSN=CHKPOINT.UPDATE.LIB
XCOPY FLAG,CHKFLAG,TDSN=(APP1.BKUP/,APP2.BKUP/)
```

Appendix C: Sample Reports

This section contains the following topics:

[Report Example](#) (see page 685)

Report Example

The following is a sample SMFRPT Report:

SDFS OUTPUT DISPLAY ISPDLH1A JOB27371 DSID 110 LINE 291 COLUMNS 02- 133										
COMMAND INPUT ==> SCROLL ==> CSR										
1997.310 NOV 06, 1997 UTILIZATION BY DATE AND TIME PAGE 6										
THURSDAY 2.31 PM CA:DISK 9.0.2										
DAY OF WEEK HOUR OF DAY										
1 1 1 1 1 1 1 1 1 2 2 2 2										
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3										
DATA SET NAME VOLUME S M T W T F S										

QSAFPROD.SMP530.SAMPNL0	QA1901			X				X		X
QSAFPROD.SMP530.SAMTGT.SMPMTS	QA1901			X				X		X
QSAFPROD.SMP530.SAMTGT.SMPSCDS	QA1901			X				X		X
QSAFPROD.SMP530.SAMTGT.SMPSTS	QA1901			X				X		X
QSAFPROD.UANTAGE.UAN3200P.PARMLIB	QA1901			X				X		X
QSAFPROD.UANTAGE.UAN3200P.PTFLIB	QA1901			X				X		X
QSAFPROD.UAN3100P.PARMLIB	QA1901			X				X		X
STAUANGR.UKG9999C.TRENDLOG.G0000U00	SSL802		X						X	X
SYS2.PRODUCT.PARMLIB	SY9CAT		X						X	X
SYS2.SIU9.OPS.PROD.SYSEXEC	SY9CAT		X						X	X
SYS4.JTRC.CHECKPT	SY9SY1		X						X	X
SYS4.SIU9.SMFEDMP	SY9PG1			X					X	X
SYS8.SMS.JCLLIB	SSL801		X						X	X
SYS97388.T235903.RA000.X59904T1.R0145105	SMS802			X						X
SYS97388.T235905.RA000.X59904T1.R0145111	IMS901			X						X
SYS97388.T235910.RA000.X59904T1.R0145120	IMS902			X						X
SYS97388.T235910.RA000.X59904T1.R0145121	IMS903			X						X
SYS97388.T235922.RA000.X59904T1.R0145122	ULS172			X						X
R 2 C 21 14:35 11/06/97										

ALLOCS

The following is a sample Volume Allocation Summary Report:

2008.283	OCT 09, 2008	VOLUME ALLOCATION SUMMARY														PAGE	1
THURSDAY	2.33 PM															CA DISK	r12
		DEVICE	UNIT	SMS	----DATA SET CONTROL BLOCKS (DSCB) COUNTS----						-----CYLINDERS-----				-----FREE SPACE-----		
VOLUME	TYPE	ADDR	STAT	FREE	F1	F2	F3	F4	F5	F6	DEVICE	ALLOC	AVAIL	IDLE	EXTENTS	THREE	LARGEST
EAV001	3390	2100	NONE	1429	13			1			102396	92	102304	.2	4	65438	73
TRACK MANAGED AREA											65520	6	65513		3	65438	.8
MN64P1	3390	4226	NONE	43	4		1	1			15	15					
MN64P2	3390	4225	NONE	44	4			1			15	14	.5	.2	2	.3	.2
MN64P3	3390	4220	NONE	45	3			1			15	14	.7		1	.7	
MN6427	3390	4227	NONE	37	10		1	1			15	15	.2	13	2	.2	.0
MN6428	3390	4228	NONE	32	15		1	1			15	15	.2	13	3	.1	.0
MN6429	3390	4229	NONE	38	10			1			15	3	12	2	3	7	.0
TOTAL VOLS															15		
TOTAL CYLINDERS PER DEVICE											102486						
TOTAL CYLINDERS ALLOCATED												168					
AVAILABLE													102318				
IDLE														28			

ATTRBS

The following is a sample Volume Attribute Summary Report:

2008.283	OCT 09, 2008	VOLUME ATTRIBUTE SUMMARY														PAGE	1		
THURSDAY	2.33 PM															CA DISK	r12		
	DEVICE	UNIT	SMS	MOUNT	ALLOCATION	-----DATA SET ORGANIZATION-----						--TOTAL--	----DSCBS----		---CYLINDERS---				
VOLUME	TYPE	ADDR	STAT	STATUS	ATTRIBUTE	PS	DA	PO	IS	AM	UN	DATA SETS	VTOC	ALLOC	DEVICE	ALLOC			
EAV001	3390	2100	NONE	RESIDENT	PRIVATE	3	1	2		7		13	1450	14	102396	92			
MN64P1	3390	4226	NONE	RESIDENT	PRIVATE	2				2		4	50	6	15	15			
MN64P2	3390	4225	NONE	RESIDENT	PRIVATE	1				2	1	4	50	5	15	14			
MN64P3	3390	4220	NONE	RESIDENT	PRIVATE	1				2		3	50	4	15	14			
MN6427	3390	4227	NONE	RESIDENT	PRIVATE	6				1	3	10	50	12	15	15			
MN6428	3390	4228	NONE	RESIDENT	PRIVATE	5		1		4	5	15	50	17	15	15			
MN6429	3390	4229	NONE	RESIDENT	PRIVATE	2		1		2	5	10	50	11	15	3			
TOTAL VOLS						7	TOTALS				20	1	4						
												TOTAL DSCBS VTOC		1750					
												ALLOC			69				
												TOTAL CYLINDERS PER DEVICE				102486			
												ALLOCATED					168		

CIARPT

The following is a sample CIARPT Report:

File Edit Transfer Options Connection Macro Window Help									
SDSF OUTPUT DISPLAY ISPDLM1A JOB22161 DSID 116 LINE 939 COLUMNS 02- 133									
COMMAND INPUT ==> SCROLL ==> CSR									
1997.303 OCT 30, 1997 CLUSTER INTERNAL ANALYSIS PAGE 19									
THURSDAY 9.43 AM CA:DISK 9.0.2									
CLUSTER NAME: LAB1UP4.DLR.P162810.USAM (KSDS)									
RBA VALUE DATA CONTROL AREA (CONTINUED)									
REC BYTE FREE DEAD OVER % % REL REL									
TOTAL TOTAL SPACE SPACE HEAD USED DEAD SEQ DEV									
868352									X
147456									
884736									*
163848									
981128									*
188224									
917584									
196688									
933888									
212992									
958272									
229376									
96656									X
245768									
983848									X
262144									*
999424									X
278528									
1015888									
Aa B TCP/IP R 2 C 21 9:44 10/30/97									

FREESP

The following is a sample Volume Free Space Detail Report:

2008.283	OCT 09, 2008	VOLUME FREE SPACE DETAIL	PAGE 1
THURSDAY	3.21 PM		CA DISK r12
VOLUME EAV001 DEVICE 3390 ADDR 2100 THIS VOLUME IS NOT SMS MANAGED			
TRACK	LEN TRACK	LEN TRACK	LEN TRACK
78	12		
165	1050		
1230	981570		
984060	551880		
TOTAL FREE TRACKS 1534512			

LAYOUT

The following is a sample Volume Layout Report:

File Edit Transfer Options Connection Macro Window Help									
SDSF OUTPUT DISPLAY ISPDLM1A JOB22052 DSID 116 LINE 3,553 COLUMNS 02- 133									
COMMAND INPUT ==> SCROLL ==> CSR									
1997.303 OCT 30, 1997 VOLUME LAYOUT PAGE 1									
THURSDAY 6.22 AM CA:DISK 9.0.2									
VOLUME SYCSY4 DEVICE 3390 ADDR 06A9 THIS VOLUME IS SMS MANAGED									
D - DATA EXTENT S - SHARED CYLINDER F - FREE SPACE U - UTOC									
TYP	LEN	TYP	LEN	TYP	LEN	TYP	LEN	TYP	LEN
F	5	D	31	D	144	D	11	F	2
D	1	D	1	F	6	D	15	D	18
D	9	D	40	D	45	D	1	D	14
D	14	D	17	D	5	D	11	F	57
D	97	D	1	D	9	F	3	D	90
D	135	D	15	D	16	D	15	F	30
D	5	D	60	D	34	D	15	F	7
D	22	D	15	D	11	D	15	F	42
F	3	D	8	D	60	D	15	D	11
D	15	F	7	F	7	D	15	D	120
D	10	D	41	D	14	D	15	D	131
D	19	F	4	F	69	D	15	F	4
D	25	D	90	D	185	D	14	D	20
F	3	D	185	D	18	D	8	D	54
D	6	F	18	D	2	F	5	D	1
D	8	D	18	D	3	D	11	D	15
F	1	D	1	D	48	D	11	F	60
Aa B TCP/IP R 2 C 21 6:28 10/30/97									

DSINDX

The following is a sample DSINDX Report:

File Edit Transfer Options Connection Macro Window Help					
SDSF OUTPUT DISPLAY ISPDLM1A JOB22179 DSID 116 LINE 1 COLUMNS 02- 133					
COMMAND INPUT ==> SCROLL ==> CSR					
1997.303 OCT 30, 1997 DATA SET INDEX PAGE 1					
THURSDAY 9.55 AM CA:DISK 9.0.2					
DATA SET NAME	VOLUME	DATA SET NAME	VOLUME	DATA SET NAME	VOLUME
LABIUP3.DAB1.CNTL	SMS901	LABIUP4.GDGER.TEST	SMS002	LABIUP4.SRB.P10443.AIX.INDEX	SMS002
LABIUP3.DLP.GDG	SMS901	LABIUP4.GDCHQVE.TEST	SMS002	LABIUP4.TEST02.MODEL.DSCB	SMS003
LABIUP3.DLR.TEST.DAT99	SMS903	LABIUP4.GLH.CDG.INCATLG.G0001U00	SMS001	LABIUP4.TEST02.MODEL.DSCB2	SMS003
LABIUP3.EMPTYPSE.TST	SMS901	LABIUP4.GLH2.ESDS.D	SMS002	LABIUP4.TEST02.MODEL.DSCB3	SMS003
LABIUP3.GLH.ABC	SMS901	LABIUP4.GLH55.CDG.TEST.G0001U00	SMS001	LABIUP4.TEST02.MODEL.DSCB4	SMS003
LABIUP3.GLH.CANDMUOL	SMS901	LABIUP4.ISPREB1.P10170.AIX.DATA	SMS002	LABIUP4.TEST02.MODEL.DSCB7	SMS003
LABIUP3.GLH.ITSATST3.DSN	SMS902	LABIUP4.ISPREB1.P10170.AIX.INDEX	SMS002	LABIUP4.TEST02.MODEL.DSCB9	SMS002
LABIUP3.GLH.JCL	SMS902	LABIUP4.ISPREB1.P10170.DATA	SMS001	LABIUP4.TETS02.MODEL.DSCB	SMS001
LABIUP3.GLH.TESTMCM	SMS902	LABIUP4.ISPREB1.P10170.INDEX	SMS001	LABIUP4.TSTLLM.MODEL.DSCB	SMS003
LABIUP3.NORESET.TST1	SMS901	LABIUP4.ISPREB1.REGRESS.DSETR065	SMS001	LABIUP4.TSTLLM.MODEL.CDC	SMS002
LABIUP3.PDSE.TEST	SMS901	LABIUP4.KDZ.DSN2	SMS001	LABIUP4.TSTLLM.MODEL.CDG4	SMS002
LABIUP3.RPN.P1163231.TEST2	SMS901	LABIUP4.KDZ.GDGTST.G0005U00	SMS001	LABIUP4.UDD.PSETEST.WRK.G0001U00	VL5173
LABIUP3.SMSCDG.TEST.G0001U00	SMS902	LABIUP4.KDZ.TEST.PS	SMS002	LABIUP4.UDD.PSETEST.WRK2	SMS001
LABIUP3.SMSCDG.TEST.G0002U00	SMS902	LABIUP4.KNI.GDGTST.G0002U00	SMS002	LABIUP4.UMB.PDSE.TEST	SMS001
LABIUP3.SMSCDG.TEST.G0003U00	SMS902	LABIUP4.KNI.GDGTST.G0002U00	SMS003	LABIUP4.MES.AVRLOCK	SMS901
LABIUP3.SMSCDG.TEST.G0004U00	SMS902	LABIUP4.KSDG.P6116300.DATA	SMS002	LABIUP4.MES.LIMITS.G0001U00	SMS003
LABIUP3.SMSCDG.TEST.G0005U00	SMS902	LABIUP4.LES.WK610463.G0001U00	SMS003	LABIUP4.MES.LIMITS.G0002U00	SMS003
LABIUP4.ABCDEF	SMS001	LABIUP4.LLM.TESTKSD9.DATA	DMS901	LABIUP4.MES.LIMITS.G0003U00	SMS003
LABIUP4.B03GLH.CDG.G0001U00	SMS002	LABIUP4.LLM.TESTKSD9.INDEX	DMS901	LABIUP4.MES.LIMITS.G0004U00	SMS003
LABIUP4.B03GLH.CDG.G0002U00	SMS003	LABIUP4.MBP.CDG.G0002U00	SMS003	LABIUP4.MES.LIMITS.G0005U00	SMS003
Aa B TCP/IP R Z C Z1 9:54 10/30/97					

DSUTIL

The following is a sample DSUTIL Report:

File Edit Transfer Options Connection Macro Window Help													
SDSF OUTPUT DISPLAY ISPDLM1A JOB22070 DSID 116 LINE 753 COLUMNS 02- 133													
COMMAND INPUT ==> SCROLL ==> CSR													
1997.303 OCT 30, 1997 DATA SET UTILIZATION PAGE 14													
THURSDAY 7.30 AM CA:DISK 9.0.2													
FLAG	-----DATES-----						C	P					
DAYS	USEDATE	MODDATE	CREDATE	EXPDATE	DATA SET NAME	VOLUME	T	O	OPEN	COUNT	JOBNAME	ALLOC	USED IDLE
\$	1997.239		1997.239		CSLUL2.MPM.R110.MPMTGT.CSI.	SMS901	-	R				2	2
					INDEX								
\$	1997.245		1997.245		CSLUL2.MPM.R110.MSGLOG	SSL007	N						
\$	1997.302	1997.241	1997.239		CSLUL2.MPM.R110.PARMLIB	SMS903	Y		21		ISPRKK17	30	20 10
\$	1997.142		1997.142		CSLUL2.P163062.KEYRANGE.DATA.	SSL001	N	R				15	15
					AB01								
\$	1997.142		1997.142		CSLUL2.P163062.KEYRANGE.DATA.	SSL003	N	R				15	15
					AB02								
\$	1997.142		1997.142		CSLUL2.P163062.KEYRANGE.INDX	SSL001	N	R				5	5
\$	1997.283		1997.283		CSLUL2.TESTSMS.EDU	SMS902	Y					7	7
	1997.213	1997.076	1997.031		CSLUL2.UANTAGE.B320.LOADLIB	SSL007	Y		45		ISPCNT19	390	320 62
	1997.300	1997.231	1996.067		CSLUL2.UANTAGE.R310.JCLLIB	SSL005	Y		656		ISPCNT1	45	37 8
	1997.302	1997.191	1996.067		CSLUL2.UANTAGE.R310.LOADLIB	SSL003	Y		560		ISPCNT1A	360	269 91
	1997.302	1997.234	1997.245		CSLUL2.UANTAGE.R310.PARMLIB	SSL003	Y		007		ISPCNT1	45	20 17
	1997.204		1996.067		CSLUL2.UANTAGE.R310.PROFILE.	SY9SY2	Y		2	- . 4.64		9	1 8
					ISPCNT2								
	1997.213	1996.267	1996.267		CSLUL2.UANTAGE.R310.T.LOADLIB	SSL001	Y		0		ISPCNT1C	450	365 85
\$	1996.354		1996.354		CSLUL2.UANTAGE.R310.T.MSGLOG	SSL002	Y						
\$	1996.354		1996.354		CSLUL2.UANTAGE.R310.T.	SSL004	Y						
Aa B TCP/IP R Z C Z1 9:10 10/30/97													

ISAMDS

The following is a sample ISAMDS Report:

```
File Edit Transfer Options Connection Macro Window Help
SDSF OUTPUT DISPLAY ISPDLM1A JOB22182 DSID 116 LINE 1 COLUMNS 02-133
COMMAND INPUT ==> SCROLL ==> CSR
1997.303 OCT 30, 1997 I S A M D A T A S E T S PAGE 1
THURSDAY 9.58 AM CA: DISK 9.0.2

ISPSMJ2.TESTIS FRG901 HAS P UOLSQ=1
DCB: LRECL=00000 BLKSZ=00000 RECFM=FB
DCB: OPTCD= KEYL=000 RKP=0000
INDEX: LOADED=NO RES=I=00000 NTM=00 LULS=00
PRIME: RECS=0000000 R/W=000000 IDLS=00000 FCOU=000 T/COU=00
OUPLOW: ***** ERROR ***** PRIME PTRS IN DSCBZ NOT WITHIN EXTENTS OF DATA SET
M RPTNIS AMDSIS
I SPACE: ALLOC TRKS=00000 USED=00000 IDLE=00000 PCT I IDLE= 00
O SPACE: ALLOC TRKS=00015 USED=00000 IDLE=00015 PCT O IDLE= 100
ALLOCATION: TOTAL TRKS=00075 EXTS= 1 TYPE=CYL PRIM=00005 UNIT=3390

QDMSTEST.P432325A.ISAM.DSN1 UL5172 HAS P UOLSQ=1
DCB: LRECL=00075 BLKSZ=01500 RECFM=FB
DCB: OPTCD=IYL KEYL=010 RKP=0001
INDEX: LOADED=YES RES=I=00000 NTM=00 LULS=1
PRIME: RECS=0000100 R/W=000000 IDLS=00000 FCOU=000 T/COU=02
OUPLOW: RECS=00000 IDLE=00014
I SPACE: ALLOC TRKS=00001 USED=00001 IDLE=00000 PCT I IDLE= 00
O SPACE: ALLOC TRKS=00014 USED=00000 IDLE=00014 PCT O IDLE= 100
P SPACE: ALLOC TRKS=00026 USED=00001 IDLE=00025 PCT P IDLE= 96
COU SPACE: ALLOC TRKS=00004 *USED=00000 *IDLE=00004 NCOU=002 T/COU=02 PCT * IDLE= 100
9 A B TCP1P R 2 C 21 9:58 10/30/97
```

MAPPER

The following is a sample MAPPER Report:

2008.283	OCT 09, 2008	V O L U M E				M A P		PAGE 1
THURSDAY	3.28 PM							CA DISK 112
VOLUME EA001 DEVICE 3390 ADDR 2100 THIS VOLUME IS NOT SMS MANAGED								
DATA SET NAME		CC	H	LEN	EXT	DATA SET NAME		CC H LEN EXT

VTOC		0000000	:	1	29	1		
STEWI09.XTEST.TRACE		0000004	:	0	4	1		
STEWI09.XTEST.TRACE2		0000004	:	4	4	1		
STEWI09.XTEST.XCLIST		0000004	:	8	4	1		
STEWI09.XTEST.CLIT		0000004	:	C	4	1		
STEWI09.XTEST.KSDS		0000005	:	1	1	3		
SCHPA17.PI3216.VSAM		0000005	:	2	1	2		
FREE SPACE		0000005	:	3	12			
SCHPA17.PI3216.VSAM		0000006	:	0	30	1		
SCHPA17.PI3216.TEST.EXTEND		0000008	:	0	15	1		
SCHPA17.PI3216.TEST.EXTEND		0000009	:	0	15	2		
SCHPA17.PI3216.TEST.EXTEND		000000A	:	0	15	3		
FREE SPACE		000000B	:	0	1050			
STEWI09.XE08.FILES		0000051	:	0	15	1		
FREE SPACE		0000052	:	0	981570			
STEWI09.XTEST.AA.KSDS		000FF0	:	0	315	1		
STEWI09.XTEST.KSDS		0010005	:	0	315	1		
STEWI09.XTEST.ESDS		001001A	:	0	315	1		

MEMBER

The following is a sample MEMBER Report:

SDFS OUTPUT DISPLAY ISPDLM1A JOB22218 DSID 116 LINE 1 COLUMNS 02- 133									
COMMAND INPUT ==> SCROLL ==> CSR									
1997.303 OCT 30, 1997 PARTITIONED DATA SET DIRECTORY PAGE 1									
THURSDAY 10.23 AM CA: DISK 9.0.2									
DATA SET NAME QDMTEST.SMP900.ADSLOAD VOLUME QA1901 DEVICE 3390 ADDR 0916									
MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER
\$\$\$SDMS	\$\$\$DM603	\$\$\$UT052	ADSDI202	ADSDM057	ADSDM116	ADSDM178	ADSDM239	ADSDM315	ADSDM375
\$\$\$AR002	\$\$\$DM613	\$\$\$UT105	ADSDI203	ADSDM058	ADSDM117	ADSDM179	ADSDM240	ADSDM318	ADSDM388
\$\$\$AR012	\$\$\$DM636	\$\$\$UT392	ADSDI221	ADSDM059	ADSDM118	ADSDM180	ADSDM243	ADSDM319	ADSDM391
\$\$\$DFLT	\$\$\$DM650	\$\$\$UC001	ADSDI223	ADSDM061	ADSDM120	ADSDM181	ADSDM244	ADSDM320	ADSDM392
\$\$\$DM002	\$\$\$DM908	AD\$AR001	ADSDM001	ADSDM063	ADSDM121	ADSDM182	ADSDM245	ADSDM321	ADSDM393
\$\$\$DM078	\$\$\$DS002	AD\$AR002	ADSDM002	ADSDM064	ADSDM122	ADSDM183	ADSDM246	ADSDM322	ADSDM395
\$\$\$DM100	\$\$\$DS028	AD\$AR003	ADSDM003	ADSDM065	ADSDM123	ADSDM184	ADSDM249	ADSDM323	ADSDM396
\$\$\$DM107	\$\$\$DS050	AD\$AR004	ADSDM004	ADSDM066	ADSDM124	ADSDM185	ADSDM250	ADSDM324	ADSDM397
\$\$\$DM114	\$\$\$DS060	AD\$AR005	ADSDM005	ADSDM068	ADSDM125	ADSDM186	ADSDM251	ADSDM325	ADSDM398
\$\$\$DM120	\$\$\$DS075	AD\$AR007	ADSDM006	ADSDM069	ADSDM126	ADSDM187	ADSDM252	ADSDM326	ADSDM401
\$\$\$DM124	\$\$\$DS076	AD\$AR008	ADSDM007	ADSDM070	ADSDM127	ADSDM188	ADSDM253	ADSDM327	ADSDM414
\$\$\$DM128	\$\$\$DS095	AD\$AR009	ADSDM008	ADSDM071	ADSDM128	ADSDM189	ADSDM254	ADSDM328	ADSDM415
\$\$\$DM151	\$\$\$SFH114	AD\$AR010	ADSDM009	ADSDM072	ADSDM129	ADSDM190	ADSDM255	ADSDM329	ADSDM417
\$\$\$DM177	\$\$\$SGEN	AD\$AR012	ADSDM010	ADSDM073	ADSDM130	ADSDM191	ADSDM256	ADSDM330	ADSDM418
\$\$\$DM179	\$\$\$SP001	AD\$AR013	ADSDM011	ADSDM074	ADSDM131	ADSDM192	ADSDM258	ADSDM331	ADSDM419
\$\$\$DM180	\$\$\$SP013	AD\$AR014	ADSDM012	ADSDM075	ADSDM132	ADSDM193	ADSDM259	ADSDM332	ADSDM420
\$\$\$DM191	\$\$\$SP014	AD\$CH000	ADSDM013	ADSDM077	ADSDM133	ADSDM194	ADSDM260	ADSDM333	ADSDM421
\$\$\$DM192	\$\$\$SP016	AD\$CH002	ADSDM014	ADSDM078	ADSDM134	ADSDM197	ADSDM261	ADSDM334	ADSDM422
Ra B TCP/IP R 2 C 21 10:22 10/30/97									

MVDICT

The following is a sample MVDICT Report:

SDFS OUTPUT DISPLAY ISPDLM1A JOB22070 DSID 116 LINE 24,316 COLUMNS 02-133																		
COMMAND INPUT ==> SCROLL ==> CSR																		
1997.303 OCT 30, 1997 MULTIPLE VOLUME DATA SET DICTIONARY PAGE 173																		
THURSDAY 7.30 AM CA:DISK 9.0.2																		
C P																		
DEVICE UNIT A R																		
---ALLOCATION--- ---TRACKS---																		
DATA SET NAME	VOLUME	TYPE	ADDR	T	D	DSO	RECFM	BLKSZ	LRECL	TYP	PRIM	SEC	ALLOC	IDLE	EXT	CREDATE	EXPDATE	USEDATE
STSMYV1.CNTL	SYCSY2	3390	090E	-	PO	FB		23440	80	TRK	13	10	13	8	1	1995.241		1997.008
STSMYV1.CNTL	SY9CAT	3380	0695	Y	PO	FB		3120	80	CYL	10	5	150	145	1	1995.172		1996.275
STSMYV1.ISPF.ISPPROF	SYCSY2	3390	090E	-	PO	FB		6160	80	TRK	3	1	3	1	1	1995.043		1997.008
STSMYV1.ISPF.ISPPROF	SY9SY1	3380	0696	Y	PO	FB		6160	80	TRK	2	1	5	4	1	1995.003		1997.275
STSMYV1.LOG.MISC	SY9SY2	3380	06AE	Y	PS	UB		3120	255	TRK	1	1	1	1	1	1996.030		1997.029
STSMYV1.NPM.CNTL	SY9CAT	3380	0695	Y	PO	FB		3120	80	CYL	3	1	45	37	1	1995.201		1996.275
STSMYV1.SA160.UTAMLIB	SYCSY2	3390	090E	N	PO	U		6144		CYL	1	1	15	14	1	1996.101		1996.117
STSMYV1.SPFTENP0.CNTL	SSL803	3380	06AF	Y	PS	FB		800	80	BLK	36	100	1	1	1	1996.067		1997.152
STSMYV1.TEXT	SYCSY4	3380	06A9	N	PO	FB		23440	80	TRK	2	10	2	1	1	1995.313		1996.255
STSMYV1.UTAMLIB	SYCSY4	3380	06A9	N	PO	U		23200		BLK	2	1	1	1	1	1996.126		1996.126
STSMYV1.UTAMLST	SYCSY4	3380	06A9	N	PO	FB		6160	80	BLK	7	2	1	1	1	1996.190		1996.326
STSPCY1.DATA1	SYXSH0	3380	060B	N	PO	FB		6160	80	BLK	21	20	3	2	1	1995.355		1997.078
STSPCY1.IEU90.LINKLIB	SYCSY2	3390	090E	-	PO	U		23200		BLK	12	3	6	1	1	1995.174		1996.289
STSPCY1.ISPF.ISPPROF.US2	SY9SY1	3380	0696	Y	PO	FB		6160	80	TRK	2	1	4	3	1	1995.068		1996.119
STSPCY1.JCL	SYCSY2	3390	090E	N	PO	FB		6160	80	TRK	2	27	2	1	1	1995.090		1996.365
STSPCY1.LCALRGAN	SYCSY2	3390	090E	N	PO	UBA		13300	133	BLK	12	3	3	1	1	1996.023		1996.023
STSPCY1.LOG.MISC	SY9SY2	3380	06AE	-	PS	UB		3120	255	BLK	15	24	15	14	1	1996.023		1996.023
STSPCY1.LOGON.CLIST	SYCSY2	3390	090E	-	PO	FB		6160	80	CYL	1	1	15	12	1	1996.005		1997.007
STSPCY1.LOGON.CLIST	SY9SY1	3380	0696	Y	PO	FB		6160	80	TRK	15	15	15	9	1	1995.095		1997.129
Ra B TCP/IP R 2 C 21 9:16 10/30/97																		

PODUMP

The following is a sample PODUMP Report:

File Edit Transfer Options Connection Macro Window Help														
SDSF OUTPUT DISPLAY ISPDLM1A JOB22241 DSID 116 LINE 606 COLUMNS 02- 133														
COMMAND INPUT ==> SCROLL ==> CSR														
1997,303 OCT 30, 1997 PARTITIONED DATA SET DIRECTORY (MEMBER NAME SEQUENCE) PAGE 12														
THURSDAY 10.37 AM VOLUME QA1901 DEVICE 3390 CA:DISK 9.0.2														
DATA SET NAME QDMSTEST.SMP900.LOADLIB														
MEMBER	LTH	C	C	H	H	R	T	T	R	C	USER	DATA		
ADSDM6 17	36	03FA000C1D	00A21D2C			00A22300	00000000	02E20000	30003000	00000000	01010000			
ADSDM6 18	36	03FA000C25	00A2262C			00A22C00	00000000	02E20009	60090000	00000000	01010000			
ADSDM6 19	36	03FA000C2F	00A22F2C			00A30100	00000000	02E20009	60096000	00000000	01010000			
ADSDM6 20	36	03FA000D04	00A3042C			00A30A00	00000000	02E20005	10051000	00000000	01010000			
ADSDM6 21	36	040A000E13	0194132C			01941900	00000000	02E2000B	F0B0F0B0	00000000	01010000			
ADSDM6 22	36	03FA000D16	00A3162C			00A31C00	00000000	02E2001D	10174E00	00000000	01010000			
ADSDM6 23	36	03FA000D22	00A3222C			00A32000	00000000	02E20005	D005D000	00000000	01010000			
ADSDM6 24	36	03FA000D2B	00A32B2C			00A33100	00000000	02E20005	C005C000	00000000	01010000			
ADSDM6 25	36	040500031C	013E1C2C			013E2200	00000000	02E20004	10041000	00000000	01010000			
ADSDM6 26	36	03FA000D34	00A3342C			00A40100	00000000	02E20005	C005C000	00000000	01010000			
ADSDM6 27	36	03FA000E04	00A4042C			00A40A00	00000000	02E20006	40064000	00000000	01010000			
ADSDM6 28	36	03FA000E0D	00A40D2C			00A41300	00000000	02E20006	E006E000	00000000	01010000			
ADSDM6 29	36	0409000306	017A062C			017A0C00	00000000	02E2000D	C00D2000	00000000	01010000			
ADSDM6 30	36	03FA000E1F	00A41F2C			00A42500	00000000	02E20007	F037F030	00000000	01010000			
ADSDM6 31	36	03FA000E28	00A4282C			00A42E00	00000000	02E20006	40064000	00000000	01010000			
ADSDM6 32	36	03FA000E31	00A4312C			00A43700	00000000	02E20005	80058000	00000000	01010000			
ADSDM6 33	36	03FA000E3A	00A43A2C			00A50200	00000000	02E20020	20100000	00000002	03010000			
ADSDM6 34	36	03FB00000A	00A50A2C			00A51000	00000000	02E20004	E004E000	00000000	01010000			
ADSDM6 35	36	03FB000013	00A5132C			00A51900	00000000	02E20004	30043000	00000000	01010000			
ADSDM6 36	36	040D000906	01BC062C			01BC0C00	00000000	02E20034	7014C000	00000000	03010100			
R 2 C 21 10:39 10/30/97														

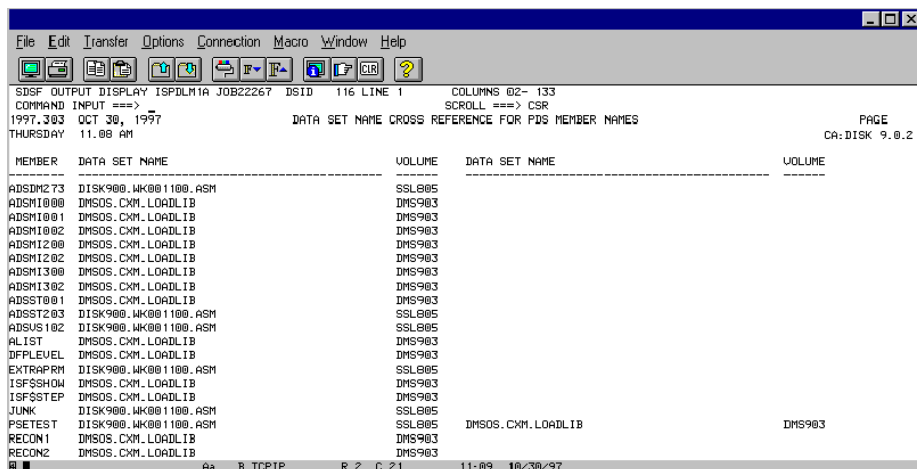
POTTRD

The following is a sample POTTRD Report:

File Edit Transfer Options Connection Macro Window Help									
SDSF OUTPUT DISPLAY ISPDLM1A JOB22241 DSID 116 LINE 1,066 COLUMNS 02- 133									
COMMAND INPUT ==> SCROLL ==> CSR									
1997,303 OCT 30, 1997 PARTITIONED DATA SET DIRECTORY (TTR SEQUENCE) PAGE 7									
THURSDAY 10.37 AM VOLUME QA1901 DEVICE 3390 CA:DISK 9.0.2									
DATA SET NAME	QDMSTEST.SMP900.LOADLIB								
T T R C	MEMBER	LTH C C H H R	USER	DATA					
00962B2C	ADSDM476	36 03FA000D2B	00963100	00000000	02E2000B	A00BA000	00000000	02010000	
0096352C	ADSDM477	36 03FA000D35	00970100	00000000	02E2000E	600E6000	00000000	01010000	
0097042C	ADSDM478	36 03FA000D104	00977000	00000000	02E2000E	600E6000	00000000	01010000	
00970D2C	ADSDM479	36 03FA000D100	00971300	00000000	02E2000D	400D4000	00000000	01010000	
0097162C	ADSDM480	36 03FA000D116	00971C00	00000000	02E2001D	40100000	00000000	01010000	
00972C2C	ADSDM483	36 03FA000D12C	00980400	00000000	02E20025	D0100000	00000002	02010100	
0098172C	ADSDM486	36 03FA000D217	00981D00	00000000	02E2001B	E0100000	00000000	05010000	
0098272C	ADSDM487	36 03FA000D227	00982D00	00000000	02E20006	A006A000	00000000	01010000	
009A012C	ADSDM490	36 03FA000D401	009A0700	00000000	02E2000D	200D2000	00000000	02010000	
009A0B2C	ADSDM491	36 03FA000D40B	009A1100	00000000	02E20014	60146000	00000000	04010000	
009A172C	ADSDM492	36 03FA000D417	009A1D00	00000000	02E20005	10051000	00000000	01010000	
009A202C	ADSDM493	36 03FA000D420	009A2600	00000000	02E20006	B006B000	00000000	01010000	
009A292C	ADSDM498	36 03FA000D429	009A2F00	00000000	02E20013	30133000	00000000	01010000	
009A322C	ADSDM502	36 03FA000D432	009B0100	00000000	02E2000C	200C2000	00000000	02010000	
009B052C	ADSDM503	36 03FA000D505	009B0800	00000000	02E20009	B009B000	00000000	01010000	
009B0E2C	ADSDM532	36 03FA000D50E	009B1400	00000000	02E2000B	90099000	00000000	01010000	
009B172C	ADSDM550	36 03FA000D517	009B1D00	00000000	02E2001A	C017E000	00000000	02010000	
009B222C	ADSDM551	36 03FA000D522	009B2B00	00000000	02E2000E	900E9000	00000000	01010000	
009B2B2C	ADSDM552	36 03FA000D52B	009B3100	00000000	02E20004	D004D000	00000000	01010000	
009C012C	ADSDM553	36 03FA000D601	009C0700	00000000	02E2002C	10100000	00000002	01010000	
R 2 C 21 10:43 10/30/97									

POXREF

The following is a sample POXREF Report:



The screenshot shows a window titled "SDFS OUTPUT DISPLAY" with a menu bar (File, Edit, Transfer, Options, Connection, Macro, Window, Help) and a toolbar. The main display area contains the following text:

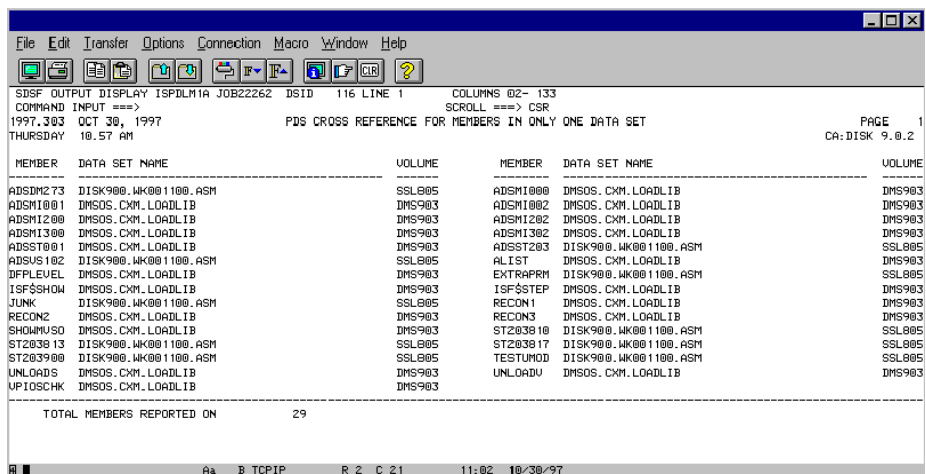
```
SDFS OUTPUT DISPLAY ISPDLMTA JOB22267 DSID 116 LINE 1 COLUMNS 02- 133
COMMAND INPUT ==>
1997.303 OCT 30, 1997 DATA SET NAME CROSS REFERENCE FOR PDS MEMBER NAMES PAGE 1
THURSDAY 11.00 AM CA:DISK 9.0.2
```

MEMBER	DATA SET NAME	VOLUME	DATA SET NAME	VOLUME
ADSDM273	DISK900.WK001100.ASM	SSL005		
ADSMI000	DMSOS.CXM.LOADLIB	DMS903		
ADSMI001	DMSOS.CXM.LOADLIB	DMS903		
ADSMI002	DMSOS.CXM.LOADLIB	DMS903		
ADSMI200	DMSOS.CXM.LOADLIB	DMS903		
ADSMI202	DMSOS.CXM.LOADLIB	DMS903		
ADSMI300	DMSOS.CXM.LOADLIB	DMS903		
ADSMI302	DMSOS.CXM.LOADLIB	DMS903		
ADST001	DMSOS.CXM.LOADLIB	DMS903		
ADST203	DISK900.WK001100.ASM	SSL005		
ADSVS102	DISK900.WK001100.ASM	SSL005		
ALIST	DMSOS.CXM.LOADLIB	DMS903		
DFFLEVEL	DMSOS.CXM.LOADLIB	DMS903		
EXTRAPRM	DISK900.WK001100.ASM	SSL005		
ISFSSHOW	DMSOS.CXM.LOADLIB	DMS903		
ISFSSSTEP	DMSOS.CXM.LOADLIB	DMS903		
JUNK	DISK900.WK001100.ASM	SSL005		
PSETEST	DISK900.WK001100.ASM	SSL005	DMSOS.CXM.LOADLIB	DMS903
RECON1	DMSOS.CXM.LOADLIB	DMS903		
RECON2	DMSOS.CXM.LOADLIB	DMS903		

At the bottom of the window, there is a status bar with the text: "Aa B TCP/IP R 2 C 21 11:09 10/30/97".

PO1REF

The following is a sample PO1REF Report:



The screenshot shows a window titled "SDFS OUTPUT DISPLAY" with a menu bar (File, Edit, Transfer, Options, Connection, Macro, Window, Help) and a toolbar. The main display area contains the following text:

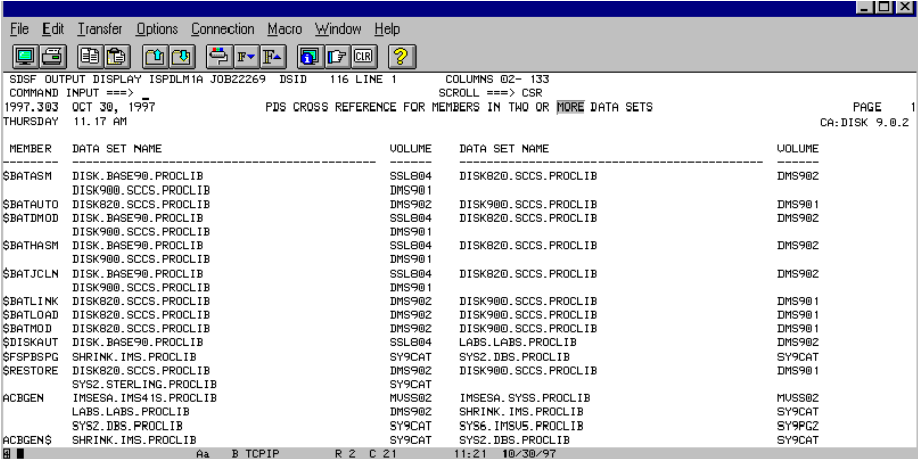
```
SDFS OUTPUT DISPLAY ISPDLMTA JOB22267 DSID 116 LINE 1 COLUMNS 02- 133
COMMAND INPUT ==>
1997.303 OCT 30, 1997 PDS CROSS REFERENCE FOR MEMBERS IN ONLY ONE DATA SET PAGE 1
THURSDAY 10.57 AM CA:DISK 9.0.2
```

MEMBER	DATA SET NAME	VOLUME	MEMBER	DATA SET NAME	VOLUME
ADSDM273	DISK900.WK001100.ASM	SSL005	ADSMI000	DMSOS.CXM.LOADLIB	DMS903
ADSMI001	DMSOS.CXM.LOADLIB	DMS903	ADSMI002	DMSOS.CXM.LOADLIB	DMS903
ADSMI200	DMSOS.CXM.LOADLIB	DMS903	ADSMI202	DMSOS.CXM.LOADLIB	DMS903
ADSMI300	DMSOS.CXM.LOADLIB	DMS903	ADSMI302	DMSOS.CXM.LOADLIB	DMS903
ADST001	DMSOS.CXM.LOADLIB	DMS903	ADST203	DISK900.WK001100.ASM	SSL005
ADSVS102	DISK900.WK001100.ASM	SSL005	ALIST	DMSOS.CXM.LOADLIB	DMS903
DFFLEVEL	DMSOS.CXM.LOADLIB	DMS903	EXTRAPRM	DISK900.WK001100.ASM	SSL005
ISFSSHOW	DMSOS.CXM.LOADLIB	DMS903	ISFSSSTEP	DMSOS.CXM.LOADLIB	DMS903
JUNK	DISK900.WK001100.ASM	SSL005	RECON1	DMSOS.CXM.LOADLIB	DMS903
RECON2	DMSOS.CXM.LOADLIB	DMS903	RECON3	DMSOS.CXM.LOADLIB	DMS903
SHOWVUSO	DMSOS.CXM.LOADLIB	DMS903	ST203910	DISK900.WK001100.ASM	SSL005
ST203913	DISK900.WK001100.ASM	SSL005	ST203917	DISK900.WK001100.ASM	SSL005
ST203900	DISK900.WK001100.ASM	SSL005	TESTMOD	DISK900.WK001100.ASM	SSL005
UNLOADS	DMSOS.CXM.LOADLIB	DMS903	UNLOADV	DMSOS.CXM.LOADLIB	DMS903
UPIOSCHK	DMSOS.CXM.LOADLIB	DMS903			

At the bottom of the window, there is a status bar with the text: "Aa B TCP/IP R 2 C 21 11:02 10/30/97".

PO2REF

The following is a sample PO2REF Report:



SDSF OUTPUT DISPLAY ISPDLM1A JOB22269 DSID 116 LINE 1 COLUMNS 02- 133
COMMAND INPUT ==> SCROLL ==> CSR
1997.303 OCT 30, 1997 PDS CROSS REFERENCE FOR MEMBERS IN TWO OR MORE DATA SETS PAGE 1
THURSDAY 11.17 AM CA:DISK 9.0.2

MEMBER	DATA SET NAME	VOLUME	DATA SET NAME	VOLUME
SBATASH	DISK.BASE90.PROCLIB	SSL004	DISK820.SCCS.PROCLIB	DMS902
	DISK900.SCCS.PROCLIB	DMS901		
SBATAUTO	DISK820.SCCS.PROCLIB	DMS902	DISK900.SCCS.PROCLIB	DMS901
SBATIMOD	DISK.BASE90.PROCLIB	SSL004	DISK820.SCCS.PROCLIB	DMS902
	DISK900.SCCS.PROCLIB	DMS901		
SBATHASH	DISK.BASE90.PROCLIB	SSL004	DISK820.SCCS.PROCLIB	DMS902
	DISK900.SCCS.PROCLIB	DMS901		
SBATJCLN	DISK.BASE90.PROCLIB	SSL004	DISK820.SCCS.PROCLIB	DMS902
	DISK900.SCCS.PROCLIB	DMS901		
SBATLINK	DISK820.SCCS.PROCLIB	DMS902	DISK900.SCCS.PROCLIB	DMS901
SBATLOAD	DISK820.SCCS.PROCLIB	DMS902	DISK900.SCCS.PROCLIB	DMS901
SBATMOD	DISK820.SCCS.PROCLIB	DMS902	DISK900.SCCS.PROCLIB	DMS901
SDISKAUT	DISK.BASE90.PROCLIB	SSL004	LABS.LABS.PROCLIB	DMS902
SFSPBSPC	SHRINK.IMS.PROCLIB	SY9CAT	SYS2.DBS.PROCLIB	SY9CAT
SRESTORE	DISK820.SCCS.PROCLIB	DMS902	DISK900.SCCS.PROCLIB	DMS901
	SYS2.STERLING.PROCLIB	SY9CAT		
ACBGEN	IMSESA.IMS415.PROCLIB	MUS902	IMSESA.SYSS.PROCLIB	MUS902
	LABS.LABS.PROCLIB	DMS902	SHRINK.IMS.PROCLIB	SY9CAT
	SYS2.DBS.PROCLIB	SY9CAT	SYS6.IMSUS.PROCLIB	SY9PC2
ACBGENS	SHRINK.IMS.PROCLIB	SY9CAT	SYS2.DBS.PROCLIB	SY9CAT

■ Ra B TCP/IP R 2 C 21 11:21 10/30/97

SVDICT

The following is a sample SVDICT Report:

</

DFSMS and Reports

Selection and reporting upon SMS constructs and their attributes is supported in the batch and online reporting features of CA Disk. Use the reports to audit the implementation to ensure compliance by all parties of the storage administration policies put into place.

An example of how to use the reports for auditing purposes is as follows:

```
FIND DSNAME=VSAM.KSDS. /  
    SELECT CRI=(DCNAME,NE,<VSKSDS>,AND,DCRECORG,EQ,KSDS)  
    REPORT MVD
```

The previous creates an MVD report on all VSAM KSDS clusters whose names begin with "VSAM.KSDS." and do not have a Data Class name equal to "VSKSDS". This field is supported for both SMS and non-SMS controlled data sets and thus both will be selected. Reviewing this report shows up any clusters violating this Data Class policy.

The SMS constructs and attributes can be included in online ISPF reports. See ISPF Custom Reports in the *Systems Guide* for more information about adding these report fields.

Appendix D: Glossary

archival

Provision for the storage of data at a lower cost, where it is not directly accessible to using programs. It includes the facility for restoring the data to the original medium on a demand basis.

archvol(s)

The tape volume(s) or disk data set(s) that contain the data sets that have been archived. Also the index to them that is kept in the files data set.

auto-restore

A CA Disk process by which data sets that have been archived are restored automatically if they are required by an application later. The data set can be restored by an online job or a batch job. This allows the DASD manager to keep free space available, yet gives the user access to data sets that are referenced infrequently. Data sets that have been archived to disk rather than to tape can be auto-restored with no operator intervention.

backup

A process by which a copy of a DASD data set is made, usually on tape, for the purpose of allowing the data set to be restored in the event of loss or damage to the original data set. Backup differs from the archive process, in that the data sets are copied, but not scratched or uncataloged.

- CA Disk backup processing provides for incremental backup, which limits copying to those data sets that have been changed since the last backup was taken. As each changed data set is identified, the backup copy is created and indexed, and then the change bit is turned off. To protect the data sets that are not changed, they are backed up regularly, such as once a week.

compress

The term *compress* has three different meanings in CA Disk.

- In PDS Compress, the term *compress*, describes the process of eliminating *dead space* in a PDS, or shrinking the size of the data set or its directory, by writing the PDS to a workfile, calculating reallocation requirements, then copying it back.
- The second usage applies to archive or backup processing. When data sets are copied to an archive medium, only active and used data blocks are read, thus resulting in PDS, ISAM and VSAM data sets being copied to the archive medium in compressed format. Subsequent restore or volume recovery results in the consolidation of several extents into one, a process sometimes referred to as *defragmentation*, and also the compression/reorganization of the data set.
- The third usage also applies to archive or backup processing, and is sometimes referred to as *compaction*. When data is being copied to an archive media, the data is analyzed and encoded (encrypted) into a new format that occupies less space. When the data set is restored, the data is automatically expanded (or de-compressed) into its original format.

data set index — DSNINDEX

A file containing a record for each data set that has been archived and has not yet expired. Separate records are created for each version of a data set that has been archived more than once. The existence of the data set index makes it possible to restore an archived data set by referencing only its name.

deferred processing

A means by which requests for certain actions are recorded and saved for later processing (queued processing). This is usually done to allow common requests to be grouped and processed in bulk in a more efficient manner, or to provide greater integrity by ensuring that certain actions are complete before other actions are begun.

dispose processing

The actual execution of those requests that at some prior time were deferred (queued); that is, deferred actions are disposed of later.

DSCB — data set control block

A record maintained by the operating system in a special DASD data set for each data set stored on a DASD volume. It contains attributes such as record format, block size and record length, as well as information about space occupied.

DSCL — data storage command language

A set of CA Disk commands providing a comprehensive, efficient common means to process VSAM and non-VSAM data sets. CA Disk to group all overlapping requests to allow all actions to be performed during a single pass of VTOCs or catalogs, preprocesses these commands.

explicit processing

Any CA Disk processing in which the data set(s) to be processed are not found by scanning either VTOCs or catalogs, but by processing a list of explicit names that either the user has provided, or have been extracted from the catalog. See "Implicit Processing".

file (first use)

A data set that has been archived and now exists as one of numerous internal files to the larger archive data set. This file represents a single archived data set and must be restored by a CA Disk utility function before an application program can reference it.

FILES (second use)

A set of logically related records maintained by CA Disk as a subfile or table. It is named after the DD statement FILES that points to it.

Files Data Set, FILES DS, or FDS

The direct access data set that has traditionally contained the FILES data. The entire BDAM data set is referred to as the files data set, while each subfile is usually referenced by its specific file name, such as the data set index file. Each file (subfile) is defined by an entry in the FILEDEFN member of parmlib.

Files Database, Files DB, or FDB

The Datacom database that contains the FILES data as tables using the original subfile names to reference them.

full pack backup

CA Disk uses the Volume-level backup method and is invoked by the VBACKUP command, making use of DSCL. Using the VRECOVER command restores data. For more information, see the chapter "VBACKUP/VRECOVER."

implicit processing

Any CA Disk processing in which the data set(s) selected for (or excluded from) processing are determined by scanning the input source, such as VTOCs or catalogs, from beginning to end, and applying user-specified criteria to determine what is or is not to be processed. The criteria can include factors such as last-use date, expiration date, create date, catalog status, blocksize, or being listed in a table. CA Disk permits multiple selection or exclusion criteria to be specified.

merge

The redistribution of archived data sets residing within several archive data sets (on tape or disk) to a smaller number of new archive data sets. Expired data sets are deleted, and the unexpired ones are sorted and grouped to use the archive media more efficiently.

migration

The movement of data from one type of storage medium to another without restricting accessibility. Reasons for migration include obtaining a lower-cost storage medium and changing the grouping of data.

MLA

Multi-Level Alias

PARMLIB — Parameter Library

A standard partitioned data set, or concatenation of data sets, containing both system-supplied members and user-supplied members that control various aspects of CA Disk processing.

prefix or group

A set of data sets or members that is identified with a common name prefix. A group name is indicated by placement of a slash (/) immediately following the last common character in a data set or member name. All data sets or members that have names that match on the number of characters preceded by the slash (/) are considered part of that group.

restore

The process by which an individual data set is restored (that is, copied from a backup or archive copy) to a DASD device. CA Disk permits restore processing without preallocation for ISAM, VSAM, PS, PO and DA data sets as well as for OS CVOL catalogs. Individual members of a PDS can also be restored. See also "Recover."

recover

The process by which a group of data sets, an entire volume, or a group of volumes is recovered (that is, copied from a backup or archive copy) to DASD. CA Disk recover processing provides for the same capabilities as restore processing, plus permits additional controls to be specified for the greater flexibility necessary for volume recovery.

retention control

An automated means by which CA Disk determines which data sets should be retained on disk and which removed.

SYSARM — System Parameter

A means by which CA Disk can be tailored and implemented to meet the unique processing needs of each installation. Sysarms control various aspects of each CA Disk function. The default values are in effect unless they are overridden either temporarily (through overriding DD statements) or permanently (through specification in the SYSARM member of the parmlib data set).

systems managed storage

The ability of an installation to specify the storage management rules and methodologies needed for their unique installation, and to then automate these within the system itself. Systems Managed Storage not only removes most if not all of the clerical effort and intervention that has historically been required, but must also provide the installation with control and flexibility in specifying its own rules and methodologies. If this second quality is overlooked, Systems Mismanaged Storage can easily be the result. The ability to manage and understand the automated system is of equal if not greater importance.

volume — level processing

CA Disk normally processes at the data set level. CA Disk can perform backup and recover processing at the volume level. For more information, see the chapter "VBACKUP/VRECOVER."

volume-level backup

This function (VBACKUP) is invoked through DSCL. It will produce a backup tape or tapes containing an entire DASD volume, starting at cylinder 0, track 0 and continuing through all used cylinders.

volume-level recover

This function is invoked under the RECOVER procedure, but through the VRECOVER command. When specified at recovery time, it overlays the entire volume in extent track order (i.e., from cylinder 0 to the end of the used cylinders).

VTOC — Volume Table of Contents

The data set on each DASD volume that contains a record called a data set Control Block (DSCB) for each data set stored on that volume. It also carries records that are used to account for free space.