

CA 1[®] Tape Management

Administration Guide

Release 12.6.00



Fifth 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 © 2013 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 Earl™ (CA Earl)
- CA Roscoe® Interface (CA Roscoe IE)
- CA Common Services for z/OS (CA Common Services)

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: Activating CA 1 and Daily Batch Processing 13

System Options	13
Activate CA 1 with TMSINIT.....	14
Dynamic Re-initialization	14
Real-time Expiration Processing.....	15
Activate Health Checks.....	15
Recommended Daily Processing	16
Sample Daily Batch Processing Flowchart.....	18
Sample Weekly Batch Flowchart.....	22
Sample Monthly Batch Flowchart	23
Reset Expiration Dates in the TMC with TMSEXPDT	24
Reports	25
Remove Tapes from Cycle Control with TMSCYCLE	25
Reports	25
Cycle Abuse	26
Remove Tapes from Catalog Control with TMSCTLG	26
Reports	27
Listing Mode.....	27
Two-Day Hold Criteria	28
Scratch and Clean Processing with TMSCLEAN	28
Data Set Retention Rules	29
Parameters	29
Required Conditions.....	30
Uncatalog Data Sets	31
TMSCLEAN and CA 1 Utilities	31
Reports	31
Track CA 1 Tapes Created on a Non-CA 1 System.....	32
Back Up and Restore the TMC and Audit Data Set with TMSCOPY	32
TMC and Audit Data Set Backup	33
Restore the Audit Data Set.....	34
Restore the TMC	34
TMC/Audit Restoration	34
TMC Backup and Restore Method for Disaster Recovery	35
Report Description	35
Execute the Vault Management System	36
Condition Code Checks	36
Test Vault Pattern Changes	37

Reports	37
Report on Tape Status in the TMC	37
TMSBINQ Utility—Print Volume, DSNB, and CA 1 Control Records	38
TMSAUDIT Utility—Analyze TMC Activity	38
TMSPTRS Utility—Maintain Multivolume/Multi-Data Set Chains.....	39
CA Earl—Produce Customized Reports	39

Chapter 2: Real-time Tape Processing 41

About Real-time Processing	41
How to Use Scratch Pool Management.....	42
Store and Identify Scratch Tapes	42
Multiple Tape Pools	43
Scratch Pool and Access Rules	43
Restrictions	44
Real-time Stacking.....	45
Mount Messages.....	45
Specific Mount Message Processing	48
Reject for Processing	49
Temporary Data Set Processing	49
Auxiliary Disposition Processing.....	49
NL and BLP Tape Processing	49
Nonresident Tape Processing.....	50
Alternate Internal Volume Substitution (AIVS)	50
Duplicate Internal Volume Serial Numbers.....	51
Foreign Tape Control.....	51
Automated Replies to IBM WTORs	52
Reset Expiration Dates in the TMC in Real Time	53

Chapter 3: Maintaining the Tape Library 55

Access and Update the TMC	55
Query and Update TMC Records.....	55
Activate and Deactivate TMC Volume Records.....	57
Add TMC Volume Ranges or DSNB Records.....	58
Remove Unused TMC Volume Ranges	59
Maintain Multivolume and Multi-Data Set Chains.....	59
Initialize and Report on Tape Volume Labels	59
Initialize Tape Volumes with TMSTPNIT.....	60
Process Header Label Information with TMSTPPRO	61
External Tape Label Production	61
TMSLBLPR.....	61
CA 1 System Option.....	62

Auxiliary Disposition	62
Online Interfaces	62
Online Label Interface	63
Problem Analysis	63
CA 1 NOT SCRATCH Demounts	64
Why Tapes Do Not Get Scratched	64
Tape Failures	65
TMSAUDIT	66

Chapter 4: Using Auxiliary Disposition 67

About Auxiliary Disposition	67
How to Use Auxiliary Disposition	67
Real-time Auxiliary Disposition	68
Job Control Statements	69
Message Control Statement	70
Real-time Auxiliary Disposition Example	71
Batch Auxiliary Disposition	72
Job Control Statements	73
Control Statement Specification	73
Data Set Name Control Statement	73
Message Control Statement	74
Batch Auxiliary Disposition Example	74

Chapter 5: Synchronizing the TMC and MVS Catalog 77

TMC and the Catalog	77
How to Use Catalog Control	77
Catalog Control Processing	77
TMSCLEAN Uncatalog Processing	78
TMSOSCAT	79

Chapter 6: Using the Vault Management System (VMS) 81

VMS Features	81
3480 Cartridge Sub-vaulting	81
On-Site VMS Protection	81
Extensive Reporting	82
VMS Overview	84
Execute VMS	85
Tape Volume Selection Criteria	86
Tape Volume Rejection Criteria	87
Exported Virtual Volume Considerations	87

Vault Reports.....	88
Picking List for Vault.....	89
Distribution List for Vault.....	90
Inventory List for Vault.....	91
Vaulted Tapes.....	91
Recommended Procedure.....	92
Use Vaulted Tapes Without Checking Them In.....	92
Prevent Tapes From Being Re-vaulted.....	92
Off-Site Vault Compression.....	93
Remove Data Sets from VMS Control.....	94
Alternate Procedure.....	94

Chapter 7: Using Online Inquiry and Update Facilities 95

CA1 TMC Record Facilities.....	95
CA 1 ISPF Facilities.....	95
CTS ISPF Facilities.....	96
Security.....	96
Fields.....	97
CA 1 ISPF Facility.....	100
Panel Description.....	100
Request CA 1 ISPF Help and Tutorial Facilities.....	101
Query and Update CA 1 ISPF Session Options.....	102
Query and Update TMC Records.....	103
Update the TMC Using the CA 1 ISPF Utility Functions.....	120
Maintain System Options and Display the Security Table.....	122
CTS ISPF Facility.....	124
Variable Record Maintenance.....	125
Volume Pool Record Maintenance.....	128
Create or Update a Volume Pool.....	129
How to Monitor Volume Pool Activity.....	131
Volume Pool Alert Maintenance.....	132
Tape Inquiry (TI) Display.....	136
TIQ Online Inquiry/Update.....	137
TMC Record Access.....	137
TIQ Online Inquiry and Update.....	138
TMC Inquiry by Volume Serial Number.....	139
TMC Inquiry by DSNB Number.....	140
TMC Inquiry on Control Records.....	140
TMC Inquiry by Cataloged Data Set Name.....	141
TMC Inquiry on Specific TMC Record Fields.....	142
TMC External Tape Label Request Generation.....	143

Update the TMC with TIQ	144
Special Utility Commands	147
Console-Unique Procedures.....	155

Chapter 8: Using the CA Dynam/T Interface 159

About the CA Dynam/T Interface	159
Communications	160
Database Synchronization.....	161
CA Dynam/T Interface Utilities.....	162

Chapter 9: Automated Tape Libraries and Virtual Tape Support 163

Tape Robotics	163
TMC Fields Used to Track Tapes in Automated Tape Libraries	164
The IBM Tape Library	165
IBM CBRUXENT Volume Entry Exit	167
IBM CBRUXEJC Volume Eject Exit.....	167
IBM CBRUXVNL Volume Not in Library	168
CA 1 ROBSCR	168
CA 1 CTSSYNC Utility Support for the IBM ATL	169
IBM ATL Recommendations and Procedures.....	169
IBM Virtual Tape Server (VTS)	170
IBM VTS - CBRUXENT Volume Entry Exit.....	171
IBM VTS - CBRUXEJC Volume Eject Exit.....	171
IBM VTS - CBRUXVNL Volume Not in Library	172
CA 1 ROBSCR Support for the IBM VTS	173
CA 1 CTSSYNC Utility Support for the IBM VTS	173
Recommendations and Procedures for IBM VTS Without IMPORT/EXPORT.....	174
Recommendations and Procedures for IBM VTS with IMPORT/EXPORT	175
IBM Basic Tape Library Support (BTLS)	177
IBM BTLS Insert Tapes	177
IBM BTLS - Synchronize the IBM ATL and TMC	178
IBM BTLS - Eject Cartridges	178
IBM BTLS - Scratch Cartridges	179
Share an IBM ATL or VTS	179
Common CA 1 TMC and Common OAM TCDB	180
Separate CA 1 TMCs and Separate OAM TCDBs	181
Common CA 1 TMC and Separate OAM TCDBs.....	182
Separate CA 1 TMCs and Common OAM TCDB.....	182
CA Vtape Virtual Tape System	183
The CA Vtape Virtual Tape System CA 1 Interface	184
Sun/STK Tape Libraries.....	185

Oracle/STK Virtual Storage Manager (VSM).....	185
---	-----

Chapter 10: Common Tape System 187

About the Common Tape System	187
CTS Initialization	188
CTS Started Task	189
CTS Commands.....	190
CA 1 Automatic Pointers Error Correction (APEC).....	191
APEC in a Shared TMC Environment	192
Startup Procedure	192
APEC Commands	197
CA 1 Data Base Services (DBS).....	198
Startup Procedure	199
Automatic Startup	199
Manual Startup	199
DBS Commands	200
CA 1 Health Checker Service (HCK)	200
Startup Procedure	201
Operator Commands.....	201
Online Label Interface (LAB).....	202
Processing Steps.....	203
Modify Labels	215
User Data Access.....	216
Input, Output and Label Examples	217
Output Assignments.....	218
Fields: PRT1_REQ, PRT2_REQ and PRT3_REQ.....	218
Fields: WTO1_REQ, WTO2_REQ and WTO3_REQ.....	218
Field: CCI_DEST	219
Sample Labels.....	220
Label formats for REELS and CARTS	220
Sample REEL Setup Label Output	221
Sample REEL Label Output	221
Sample CART Setup Label Output	222
Sample CART Label Output	222
Initialization Procedures	223
LAB Description	225
Startup Procedure	225
LAB Commands	227
Command	228
Distributed Tape Support (DTS)	233
Volume Serial Number Ranges in the TMC	234

CTS proc changes	234
Security Requirements	235
Startup Procedure	235
DTS Control Statements Contained in CTODTS00	236
Operator Commands	239
Sample SRVDEF Definitions	240
CTS Scheduler (SCHD)	241
CTS Proc Changes	241
Automatic Startup	241
Manual Startup	242
SCHD Control Statements Contained in CTOSCH00	242
Event Definition Statements	242
Sample Schedules	244
CA 1 SMF Queue (SMFQ)	249
Startup Procedure	250
SMFQ Commands	251
Tape Map (TMAP)	252
Startup Procedure	252
TMAP Commands	253
Command Sample	253
Volume Pool Monitor (VPM)	254
VPM Prerequisites	255
VPM Subtask Variables	256
VPM Email Variables	257
Start the VPM Subtask Automatically	259
Start the VPM Subtask Manually	259
Operator Commands	260

Chapter 11: Using Volume Pool Monitor Alert and Email Facilities **263**

Volume Pool Monitor Usage and Components	264
Volume Pools	265
Volume Pool Alerts	265
Variables	266
Variable Substitution	267
User and Option Variables	267
Product Variables	268
System Variables	274
Create a WTO	276
WTO Example	277
Create an Email	278
Define Email Headers	279

Email Header Fields	280
Email Body.....	281
Email Example 1	283
Email Example 2	284

Appendix A: Integration with CA OPS/MVS EMA 287

Overview	287
Ensure that CA OPS/MVS Is Enabled for Capturing These Events	288
CA 1 Active State Events.....	289
CA 1 Heartbeat Events	290

Appendix B: CA 1 Health Checker 293

About the Health Checker	293
Prerequisites	294
Parameter Overrides for CA 1 Checks	294
CA1_AUDIT_VRFY_WITHIN_LOW_THRSH.....	295
CA1_AUDIT_VRFY_WITHIN_MED_THRSH.....	296
CA1_FREE_DSNB_LOW_THRSH.....	297
CA1_FREE_DSNB_MEDIUM_THRSH	298
CA1_FREE_DSNB_QUICK_SCAN	299
CA1_TMC_AUDIT_PLACEMENT.....	300
CA1_USED_DSNB_FREE_CHAIN	301
CA1_VRFY_OPTION_DCHG.....	302
CA1_VRFY_OPTION_LCHG	303
CA1_VRFY_OPTION_TCHG	304
CA1_VRFY_MIXED_EXPDT_OPTION	305
CA1_VRFY_SECURITY_EXIT_FUNC.....	306
CA1_VRFY_SECURITY_EXIT_PSWD	307
CA1_VRFY_SECURITY_EXIT_YSVC.....	308
CA1_VRFY_SECURITY_PROFILE_CMD	309
CA1_VRFY_SECURITY_PROFILE_TAPE	310

Index 311

Chapter 1: Activating CA 1 and Daily Batch Processing

This section contains the following topics:

[System Options](#) (see page 13)
[Activate CA 1 with TMSINIT](#) (see page 14)
[Recommended Daily Processing](#) (see page 16)
[Reset Expiration Dates in the TMC with TMSEXPDT](#) (see page 24)
[Remove Tapes from Cycle Control with TMSCYCLE](#) (see page 25)
[Remove Tapes from Catalog Control with TMSCTLG](#) (see page 26)
[Scratch and Clean Processing with TMSCLEAN](#) (see page 28)
[Back Up and Restore the TMC and Audit Data Set with TMSCOPY](#) (see page 32)
[Execute the Vault Management System](#) (see page 36)
[Report on Tape Status in the TMC](#) (see page 37)

System Options

Several of the CA 1 system options have a significant impact on daily CA 1 maintenance processing. These options are defined to CA 1 during initialization in the *hlq.CTAPOPTN* library.

Note: For more information about CA 1 system options, see the *Programming Guide*. For more information about the JCL, control statements, and sample output of each utility, see the *Utilities and Reports Reference Guide*.

Activate CA 1 with TMSINIT

CA 1 can only be activated by executing the TMSINIT batch utility. Run the CAS9/CAIRIM initialization procedure before TMSINIT. CAIRIM initialization for CA 1 must complete successfully for TMSINIT to execute properly. During initialization, you receive the IEFTMSO message through a WTOR. The normal response to this message is U to activate CA 1. Other acceptable responses are listed on the console when HELP is replied to the IEFTMSO WTOR.

Depending on when TMSINIT was last executed, and the selected options, you can receive other date verification messages.

Note: For more information about the console messages, see the *Message Reference Guide*.

Responses to TMSINIT are eliminated by using the JCL parameter NODATE. Using this parameter automatically places CA 1 in an active state and uses the system date as the current date.

After the first TMSINIT is executed at system IPL, a security call is made. The call validates that the user submitting TMSINIT, or the console operator if TMSINIT is executed as a started task, is authorized to reinitialize, batch activate, or deactivate CA 1.

Initialize CA 1 by entering START TMSINIT through the console at IPL time, or through the CAUTOCMD member of *hlq*.CTAPOPTN for automated startup following CAIRIM initialization. Once CA 1 is initialized, TMSINIT terminates. TMSINIT does not stay active as a task.

Do not execute TMSINIT while TMSCLEAN, TMSCYCLE, TMSCTLG, TMSEXPDT, or TMSXTEND is running.

Dynamic Re-initialization

You can dynamically reinitialize CA 1® Tape Management to change one of the *hlq*.CTAPOPTN processing options. If you execute TMSINIT with no parameter, TMSINIT reloads the options into the CA 1® Tape Management resident module and the new options take effect immediately. You do not have to stop tape processing or deactivate CA 1® Tape Management for the purpose of changing the options.

Real-time Expiration Processing

You can optionally activate Real-time Expiration processing by adding a TMSRDS DD statement for the Retention Data Set (RDS) to your TMSINIT procedure. When Real-time Expiration processing is activated this way, TMSINIT processes the RDS rules that you have defined and creates an in-memory table that is referenced as each tape file is being created. For each tape file matching a retention rule, the Tape Management Catalog (TMC) is updated with the corresponding expiration date specified in the RDS.

Activate Health Checks

For the first execution of TMSINIT after an IPL to activate or batch activate CA 1, TMSINIT initiates the CA 1 health checks in the CA Common Services CAHCHECK address space. The CAHCHECK address space comes up early in the IPL and remains up for the entire IPL. The CA 1 health checks that are hosted by the CAHCHECK address space monitor CA 1 for the life of the IPL.

TMSINIT will initiate the CA 1 health checks only if the CAHCHECK address space is available. To initiate and host the CA 1 health checks, start the CTS address space. If CTS is started in an environment where the CAHCHECK address space is not available, CTS automatically starts the CA 1 health checks.

TMSINIT can be used to change system options that CA 1 health checks examine. Any subsequent executions of TMSINIT trigger the CA 1 health checks to run again regardless of whether TMSINIT or CTS originally initiated them.

Recommended Daily Processing

Many of the following utilities and reports are involved in the CA 1® Tape Management daily maintenance cycle.

Note: For more information about the utilities, see the *Utilities and Reports Reference Guide*.

TMSEXPDT

TMSEXPDT enables you to enforce tape data set retention standards. This utility updates the expiration date in the Tape Management Catalog (TMC) based on control statements in the Retention Data Set (RDS). TMSEXPDT is used to establish expiration dates in the TMC. Execute the TMSEXPDT before TMSCTLG, TMSCYCLE and TMSCLEAN. The TMSCTLG and TMSCYCLE utilities use the expiration dates in the TMC to determine which data sets will be removed from Catalog Control and Cycle Control and expired. Expired data sets are eligible for scratch processing with the next execution of TMSCLEAN.

IDCAMS/IEHLIST

If TMSCTLG is needed, then as an option, the system utilities IDCAMS or IEHLIST can be executed to create a temporary data set for use by CTSSTEX before TMSCTLG. TMSCTLG issues LOCATEs to the system catalog. The use of IDCAMS or IEHLIST eliminates the LOCATEs required, which reduces the runtime of the TMSCTLG program. An IDCAMS/IEHLIST input listing is required if non-shared system catalogs exist in an environment where multiple CPUs share the TMC.

TMSCTLG

If you are using CA 1® Tape Management Catalog Control (CATLG or CATALOG in control statements and online interfaces, or LABEL=EXPDT=99000 or ACCODE=xCACATLG in JCL) for your tape data sets, and the TMOOPTxx parameter OCTLG is set or defaulted to YES, then TMSCTLG should be executed after TMSEXPDT and before TMSCLEAN to identify those data sets which no longer have an entry in the system catalog. TMSCTLG determines which data sets will be removed from Catalog Control. The expiration date in the TMC Volume record is changed to the current date plus an extension that is defined by parameter R9 in the TMOOPTxx member of *hlq.CTAPOPTN*, or the highest date from any secondary files. If the expiration date was changed to the current date, the next execution of TMSCLEAN places these tapes in scratch status.

TMSCYCLE

If you are using CA 1® Tape Management Cycle Control (CYCLE/ccc in control statements and online interfaces or LABEL=EXPDT=99ccc in JCL), execute TMSCYCLE after TMSEXPDT and before TMSCLEAN to identify data sets which will be removed from Cycle Control. To determine if the tape should be expired, TMSCYCLE looks at the first file on the tape only. The expiration date in the TMC volume record is changed to the current date plus an extension that is defined by parameter RC in the TMOOPTxx member of *hlq*.CTAPOPTN. If the expiration date was changed to the current date, the next execution of TMSCLEAN places these tapes in scratch status.

TMSCLEAN

TMSCLEAN is the *primary* method that is provided by CA 1. In this method, tapes under CA 1 control can be scratched and made available for output use. A tape cannot be reused until it is put into scratch status. The program determines which data sets will be removed from CA 1 Days Since Last Used Control (LDATE/ddd in control statements and online Interfaces or LABEL=EXPDT=98ddd in JCL). The program resets data sets using CA 1 Catalog Days Control (CATLG/ddd in control statements and on-line interfaces, or LABEL=EXPDT=90ddd in JCL) to CA 1 Catalog Control after the minimum ddd days requirement has been satisfied. The program can optionally expire tapes whose expiration date exceeds the assigned SMS Management Class maximum retention value. The expiration date in the Volume record is changed to the current date.

If the OCTLG option is set to NO (indicating all systems sharing the TMC are at r11.5 or higher and have had the TMSOSCAT utility run with the SYNC option), TMSCLEAN changes the expiration date of all files that are set to Catalog Control and are no longer cataloged. You can then stop running TMSCTLG.

TMSCOPY

TMSCOPY should be executed daily, or as frequently as site needs require (such as once per shift), to back up the TMC and Audit data sets to tape. It should *always* be executed after daily maintenance, and is often executed before daily maintenance. The execution of TMSCOPY before daily maintenance gives you the opportunity to restore the TMC to this backup level if there is a problem with the maintenance run. Invalid update activity resulting from the maintenance run can be discarded when the TMC is restored to its pre-maintenance state. Run TMSCOPY before the execution of the Vault Management System (VMS) so that an alternate copy of the TMC can be moved with your other data sets to an off-site vault. VMS must be equipped with the data set name of this alternate backup.

Vault Management

The Vault Management System (VMS) consists of three batch programs: TMSVMEDT, TMSVMVLT, and TMSVMUPD. CA Earl is used to print the reports. If you plan to incorporate these programs into your daily maintenance job stream, they should be executed after TMSCLEAN has performed scratch processing. If VMS is executed before TMSCLEAN, it is possible that tapes returned from off-site are expired and eligible to be scratched. The tapes appear on the Scratch and Clean Listing while still in transit. Running TMSCLEAN before VMS prevents you from looking for scratch tapes that are not yet back in the library. If the vault programs are not included in your daily maintenance cycle, execute them when you are ready to move tapes off-site.

Execute TMSVMEDT after all tapes to be vaulted have been created. Otherwise, movement may not be complete, especially when a multi-volume data set is still in creation. When TMSVMEDT selects the TMC records, only those volumes that have already been created and the volume still on the drive are considered. Volumes that are created subsequently are not moved until the next VMS execution.

Reporting Programs

Inventory, pre-staging, Audit, and customized reports are typically included in daily maintenance. Some of the more commonly used reporting programs and CA Earl source members are:

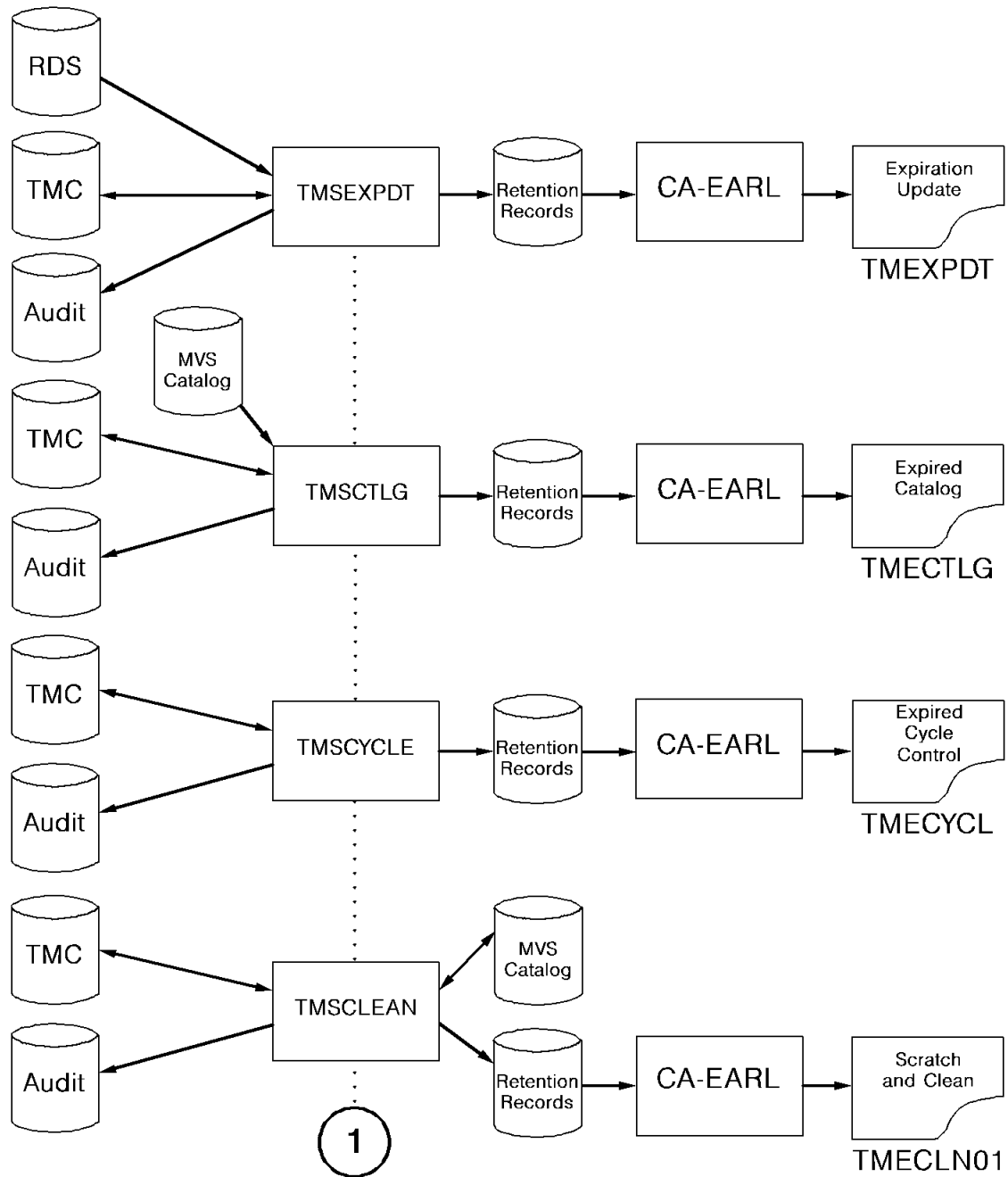
- TMEVSNM - volume serial master report that is produced by CA Earl
- TMEJOB01 - DSN list by creating job that is produced by CA Earl
- TMEDSN02 - scratch forecast that is produced by CA Earl
- TMSAUDIT - reports on Audit data set activity

Sample Daily Batch Processing Flowchart

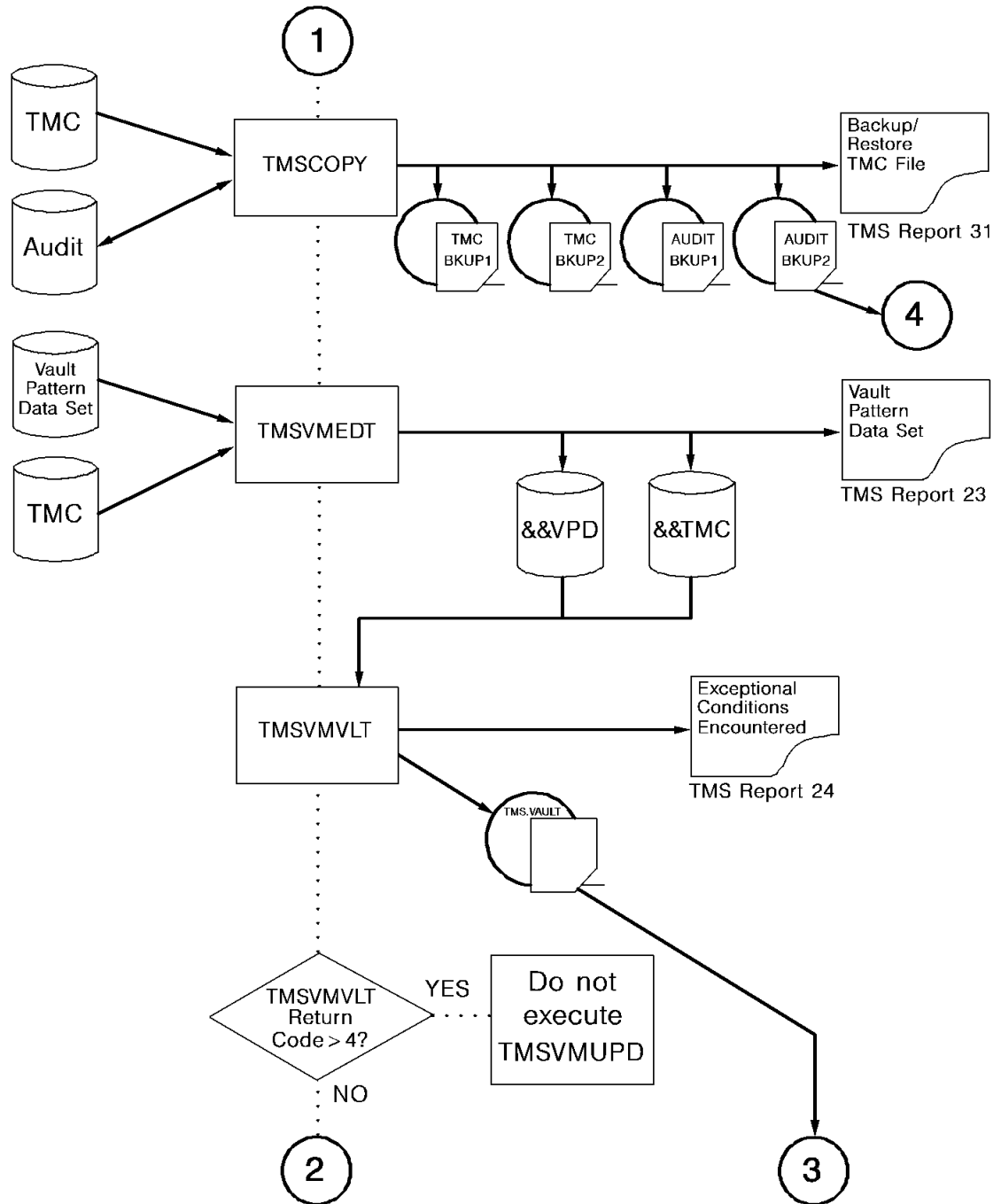
The Sample Daily Batch Processing Flowchart illustrates the *sequence* in which many of the utilities should be executed.

Note: For a report cross-reference see the *Utilities and Reports Reference Guide*. Source member EARLINDX lists and describes each CA Earl member.

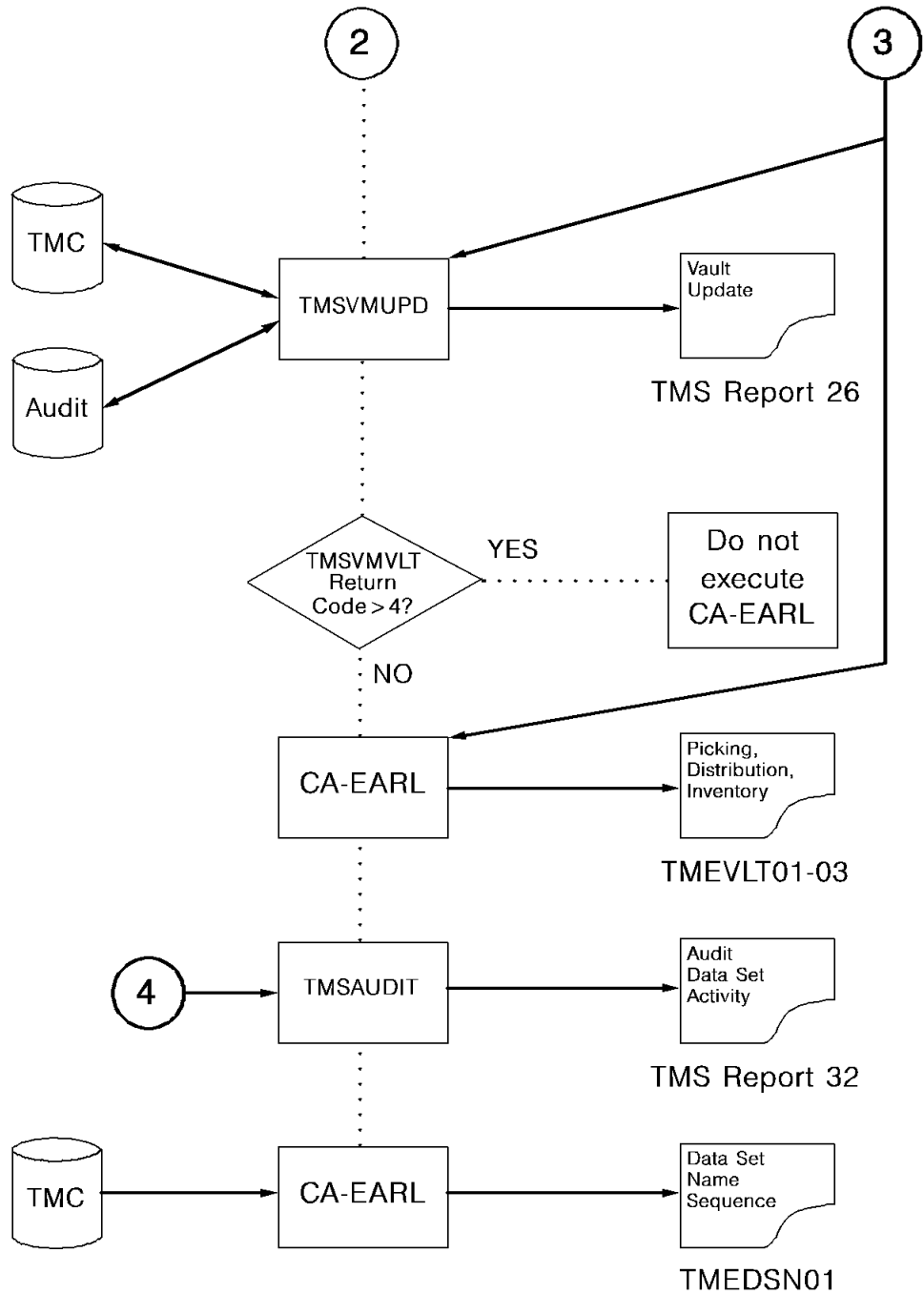
The following is a Sample Daily Batch Processing Flowchart (1 of 3):



The following is a continuation of the Sample Daily Batch Processing Flowchart (2 of 3):



The following is a continuation of the Sample Daily Batch Processing Flowchart (3 of 3):



Sample Weekly Batch Flowchart

The normal daily maintenance job stream should include the following additional steps for weekly maintenance:

IEBGENER

Concatenate the Audit data set backups that are created by TMSCOPY in chronological sequence as input to IEBGENER or a similar utility. Copy them to a single file for use in continued reporting. Do *not* edit the Audit data set backups being concatenated, copy the Audit control record with the Audit records.

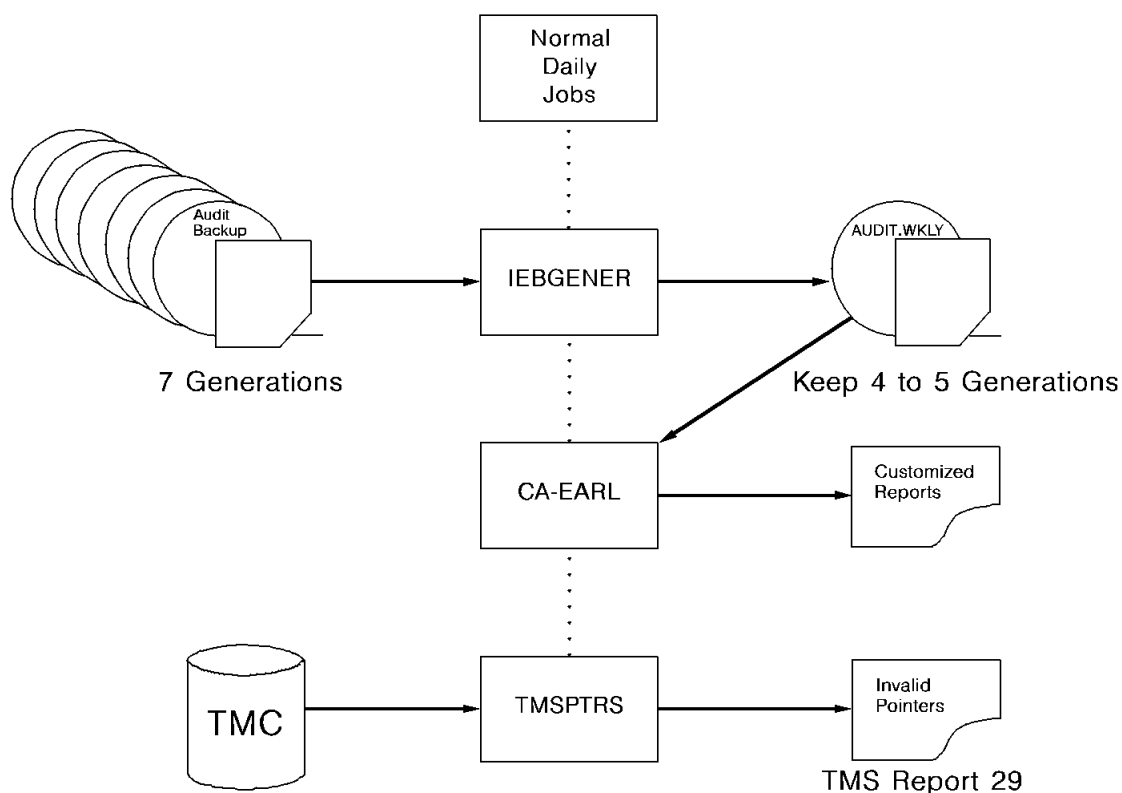
CA Earl

Research problems and trends in your tape processing environment by running CA Earl to process the cumulative Audit data.

TMSPTRS

Run the TMSPTRS utility to analyze and report on any errors in CA 1 multi-volume chains or in chains with more data sets.

The following is a sample Weekly Batch Flowchart:



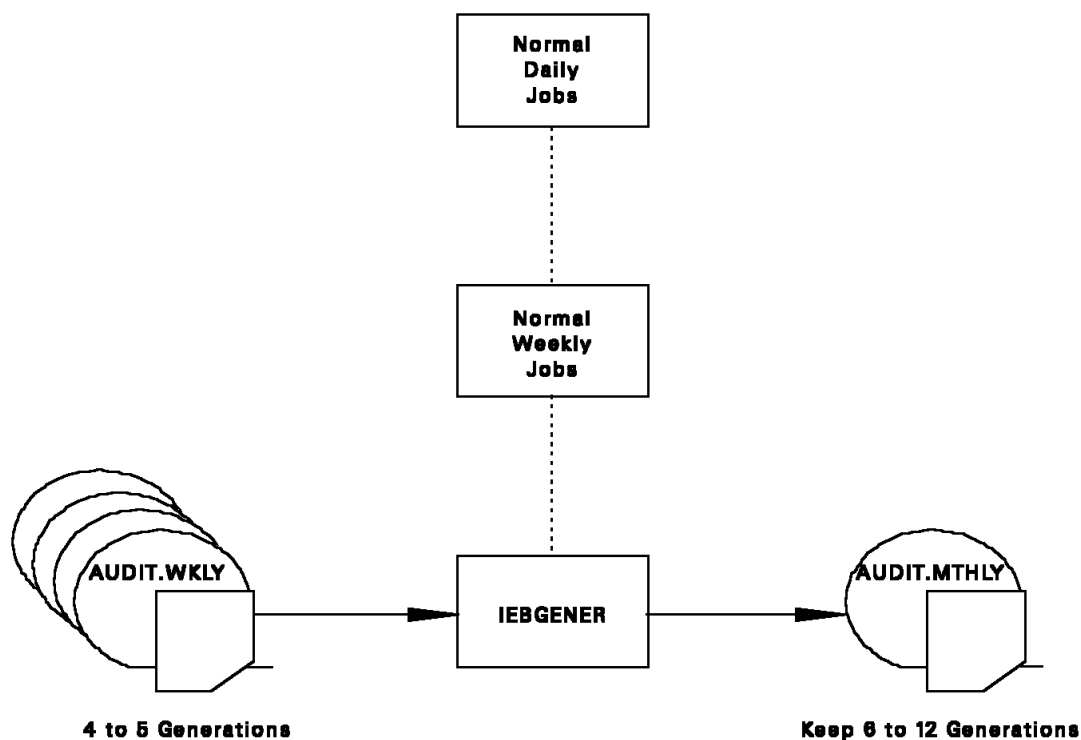
Sample Monthly Batch Flowchart

Create a monthly tape of cumulative Audit data. Add an additional step to your normal weekly maintenance job stream.

IEBGENER

Concatenate the weekly Audit data set backups that are created by TMSCOPY in chronological sequence as input to IEBGENER or a similar utility. Copy the backups to a single file to create a monthly maintenance tape. Always have from 6 to 12 months of Audit data available.

The following is a sample Monthly Batch Flowchart:



Reset Expiration Dates in the TMC with TMSEXPDT

The TMSEXPDT utility is used to assign expiration dates in batch. You can also assign expiration dates in real time as tape files are being created. For more information about assigning expiration dates using the Real-time Expiration feature, see [Reset Expiration Dates in the TMC in Real Time](#) (see page 53).

TMSEXPDT enables you to enforce data set retention standards without modifying the JCL used to create tape data sets. If the JCL does not specify an EXPDT, RETPD, or an ACCODE date specification, CA 1 assigns the *default retention* which is defined using the RP parameter in the TMOOPTxx member of *hlq*.CTAPOPTN. When the CA 1 default retention is used, TMSEXPDT has the capability to reset the date if the Retention Data Set (RDS) has an entry for the data set name.

Optionally, you can override EXPDT or RETPD values supplied by the JCL by setting the Retention Override (RO) parameter in TMOOPTxx to YES. The eligible for RDS override bit in FLAG1 is turned on for all tape data sets, allowing you to enforce your data set retention standards. If the Retention Override option is set to NO, the eligible for RDS override bit in FLAG1 is not turned on (when the JCL specifies an EXPDT or RETPD), preventing RDS override action.

The RDS is a sequential file containing 80-byte records describing the SMS Management Class or the data set name, and optional qualified job name and the desired EXPDT or RETPD. A separate retention can be assigned to data sets that are closed by Abend processing by using the ABEND= keyword. The data set name can be fully qualified or can be coded using the CA 1 pattern masking specifications. Parameters are available that enable you to test changes to your RDS and enforce retention for secondary files (TMSEXPDT defaults to override only file 1).

When TMSEXPDT queries the TMC to determine if an RDS override should be applied, it first verifies that the record is eligible for RDS override processing. The X'01' bit in the FLAG1 field must be on for the record to be eligible for RDS override processing. Once the Volume record is processed, the X'01' bit in FLAG1 is turned off. The X'04' bit in the FLAG3 (default expiration date that is used at open) in the TMC indicates whether the default retention field was assigned when the data set was created.

A keyword of SELECT=ALL allows all data sets (even data sets that were previously processed by TMSEXPDT) to be reprocessed. This keyword is specified at the data set level in the RDS. Thus, TMSEXPDT updates all data sets, with this keyword specified, with the new expiration date.

If you are using the Retention Data Set, execute TMSEXPDT before TMSCTLG, TMSCYCLE, and TMSCLEAN in your daily maintenance cycle.

Note: For more information about TMSEXPDT, see the *Utilities and Reports Reference Guide*.

Reports

For reporting, a flat file in the CA 1 Retention Record format is created for use with CA Earl in a subsequent step.

Remove Tapes from Cycle Control with TMSCYCLE

TMSCYCLE selects all TMC records with an EXPDT subject to Cycle Control (CYCLE/ccc in control statements and online interfaces, or LABEL=EXPDT=99ccc in JCL), to determine if any volumes associated with the data set should be removed from Cycle Control. TMSCYCLE retention is determined by the first file in a volume set only.

To remove volumes from Cycle Control, TMSCYCLE updates the expiration date in the TMC to the current date plus a number of days into the future. This additional number of days is defined in the TMOOPTxx member of *hlq.CTAPOPTN* (the RC parameter specifies this cycle extension). TMSCYCLE does not consider the generation/version number for GDG data sets. GDGs under Cycle Control can expire before meeting the GDG limit. TMSCYCLE updates the X'04' bit in FLAG2 to indicate the volume has been expired from Cycle Control. Once expired, the volumes are eligible for scratch processing with the next execution of TMSCLEAN.

Two other parameters in the TMOOPTxx member define the characteristics of a cycle to TMSCYCLE:

- The CDAY parameter can be set to TAPE or DAY. When set to TAPE, each individual creation of a data set under Cycle Control is considered a single cycle. When set to DAY, all creations of a data set with the same creation date are collectively considered a single cycle.
- The CJOB parameter can be set to JOB or DSN. When set to JOB, multiple job names creating the same data set have separate cycles. When set to DSN, the creating job name is not considered by TMSCYCLE.

Execute TMSCYCLE before TMSCLEAN to determine the scratch eligibility of tapes under Cycle Control.

Note: For more information about TMSCYCLE, see the *Utilities and Reports Reference Guide*.

Reports

A flat file in the CA 1 Retention Record format is created for use with CA Earl in a subsequent step.

Cycle Abuse

Cycle abuse is the difference between any two Cycle-Controlled tapes with the same data set name. Using a JCL parameter to TMSCYCLE, you can identify abuse of Cycle Control.

Assume, for example, that a data set uses LABEL=EXPDT=99005 in JCL for a five-cycle rotation. A specification of LABEL=EXPDT=99050 was accidentally used during the most recent creation of the data set. When the data set created with CYCLE/050 becomes the oldest, TMSCYCLE keeps 50 creations of the data set until the 51st is created. A cycle abuse setting of 1 effectively allows you to know if the Cycle Control value has changed in any way; however, you can use a higher setting.

Remove Tapes from Catalog Control with TMSCTLG

Note: If all systems sharing the TMC are at r11.5 or higher *and* have had the TMSOSCAT utility run with a PARM=SYNC, this utility is *no* longer needed. Change the system option OCTLG to NO (default is YES) and reinitialize the CA 1 options. TMSCLEAN processes all Catalog Control files once OCTLG has been changed to NO.

Catalog Control is one of the most effective ways to retain tapes under CA 1. TMSCTLG reads the TMC for volumes under Catalog Control (CATLG or CATALOG in control statements and online interfaces, or LABEL=EXPDT=99000 or ACCCODE=xCACATLG in JCL). TMSCTLG queries the system catalog to see if the data set name still has a corresponding entry.

If no entry for the first file is found, TMSCTLG queries each DSNB associated with secondary files on the volume. If a DSNB using Catalog Control still has an entry for the DSN in the system catalog, Catalog Control is still used for the volume set. Otherwise, TMSCTLG resets the expiration date to *one* of the following:

- The current date plus the number of days that is defined to the R9 parameter in the TMOOPTxx member of *hlq*.CTAPOPTN
- The highest expiration date that is associated with a DSNB record, if it is greater than the current date, plus R9

TMSCTLG updates the X'08' bit in FLAG2 of the volume record or the X'20' bit in FLAG1 of the DSNB record to indicate the data set has been expired from Catalog Control. Once expired, volumes are eligible for scratch processing with the next execution of TMSCLEAN. Execute TMSCTLG before TMSCLEAN to determine the scratch eligibility of tapes under Catalog Control. A RESETCAT parm is provided with TMSCLEAN to update all TMC records with the expired from Catalog Control bit on by turning this bit off and changing the expiration date back to CATALOG before scratching the tapes. This is used in cases, where system failures have resulted in lost catalogs, missing catalog lists, or some other problem that caused more than the correct number of volumes to be expired.

Note: For more information about TMSCTLG, see the *Utilities and Reports Reference Guide*.

Reports

A flat file in the CA 1 Retention Record format is created for use with CA Earl in a subsequent step.

Listing Mode

This mode of operation for TMSCTLG is available for multi-CPU installations in which system catalogs are not accessible to the CPU on which TMSCTLG is executed. IDCAMS listings are provided from all system catalogs for tape data sets, including the system on which TMSCTLG is executed. Data set information such as the data set name, file sequence, volume sequence, and media type is extracted, sorted, and written to a VSAM data set in a separate procedure before executing TMSCTLG. This VSAM file is used by TMSCTLG to determine if a data set is still cataloged on any system catalog. This includes file 1 data sets and secondary data sets. LOCATEs are not performed in this mode.

All IDCAMS listings must be provided to the listings extract program CTSLSTEX. Catalog Listing Verification uses supplied system catalog data set names and verifies that the extract data set contains entries from each system catalog data set name specified. If any system catalog data set specified is not found, TMSCTLG terminates.

Note: If you have non-shared catalogs, we recommend setting OCTLG to NO.

Two-Day Hold Criteria

The expiration date in the TMC is posted during OPEN processing but the system catalog entry is posted after step termination. TMSCTLG provides a two-day hold limit to protect long-running tape jobs from expiring prematurely.

If TMSCTLG cannot find an entry in the system catalog but the data set in question is less than two days old and has not been read for input using the system catalog to retrieve the VOLSER (the X'80' bit in FLAG2 is off), the data set is not removed from Catalog Control.

Note: The two day hold also applies when OCTLG is set to NO. TMSCLEAN honors the same two-day hold.

Scratch and Clean Processing with TMSCLEAN

TMSCLEAN is a multipurpose program that performs different functions depending on parameters specified in the JCL.

TMSCLEAN also processes the TMC for volumes under Days Since Last Used Control (LDATE/ddd in control statements and online interfaces, or LABEL=EXPDT=98ddd in JCL). Tapes that have not been used for ddd number of days, based on the current system date and the last used date, are processed. TMSCLEAN resets the expiration date of the Volume record in the TMC to the current date. TMSCLEAN updates the X'02' bit in FLAG2 to indicate that the volume has been expired by Days Since Last Used control. USER/uuu dates can be processed by TMSCLEAN with the optional user exit TMSXCLN2.

TMSCLEAN also processes the TMC for volumes under Catalog Days Control (CATLG/ddd in control statements and online interfaces, or LABEL=EXPDT=90ddd in JCL). Catalog Days Control retains a data set for the number of days specified. Once the retention days requirement is met, TMSCLEAN resets the expiration date to CATALOG, and then the tape volume is under Catalog Control.

If the OCTLG option has been set to NO and reinitialization of CA 1 options has taken place, the TMSCLEAN processes all files under Catalog Control. The new TMISCAT or DSNBISCA flag bits are checked. If the file is not currently cataloged and either A) it was cataloged (TMCATLOG or DSNBISCA bits) or B) the file was created at least 2 days before, then the expiration date is changed to the current date plus R9.

Criteria for assigning tapes to scratch tape pools and the scratch tape pool names can be defined in the TMOSCRxx and TMONSMxx members of *hlq.CTAPOPTN*. If you have assigned scratch pools to the TMC, the Scratch and Clean Listing can be segregated by pool name and label type within the pool.

TMSCLEAN counts the tapes that are owned by CA Dynam/T and includes the tapes in the report totals. Tapes controlled by EDMs are included in report totals. CA 1 does not scratch tapes owned by CA Dynam/T or EDMs.

Data Set Retention Rules

SMS maximum retention rules can be applied to data sets. If parameters SMS and SR are both specified to YES in the TMOOPTxx member of *hlq*.CTAPOPTN, and a valid SMS Management Class is assigned to the data set, the expiration date is checked with the SMS Management Class definition.

If the TMC expiration date exceeds the SMS Management Class rule, the expiration date is set to the current date plus one day. The X'80' bit in FLAG4 is set to indicate that the volume was expired by SMS. A warning message appears on TMS Report-85.

This method expires *any* volume that exceeds the SMS Management Class maximum retention value. The volumes include data sets with CA 1 keyword expiration dates.

Parameters

The following parameters are used with TMSCLEAN:

SCRATCHLIST

This parameter marks expired tapes available for use, and produces a file containing a list of all scratched data sets with a starting volser and file sequence number. Although a tape volume has expired, the volume cannot be used for output until it has gone through scratch processing. This listing identifies the volume serial numbers of tapes that have moved into scratch status since TMSCLEAN was last executed. Tapes are bypassed from SCRATCHLIST for the following reasons:

- They are marked out-of-area
- They are 3592 WORM cartridges
- They are managed by CA Dynam/T or an External Data Manager

Virtual WORM tapes are processed and get their WORM information cleared when they are scratched.

SCRATCHRELIST

This parameter produces a report file of all active tapes in scratch status, regardless of how long the volumes have been in scratch status. No additional scratching of volumes is performed with SCRATCHRELIST.

EXTEND

This parameter extends the expiration date for tapes that are due to expire on a specific date or within a range of days, such as during company holidays or vacation shutdowns.

RESET

This parameter scans the TMC and turns off the scratch bit for any volumes that have an expiration date in the future. If an invalid date was used to IPL the system, and TMSCLEAN was executed with the SCRATCHLIST parameter, the invalid date could cause the scratch bit to be turned on prematurely. The data sets are not re-cataloged to the system catalogs.

RESETCAT

This parameter updates TMC volume and DSNB records that are marked as expired by Catalog Control. This parameter is used if TMSCTLG expired data sets due to missing system catalog information.

TEST

This parameter suppresses updating of the TMC but the appropriate reports are still generated. The TEST parameter can be used with any of the options that are listed previously. When specified, TEST must be the first entry in the parameter field.

Required Conditions

For TMSCLEAN to scratch a tape, meet all of the following conditions:

1. Execute TMSCLEAN with the SCRATCHLIST parameter and without the TEST parameter.
2. The expiration date in the TMC must be less than or equal to the system date on which TMSCLEAN is executed.
3. The X'01' bit indicating expire by an External Data Manager in FLAG2 is turned on if the controlled by EDM bit (X'20') or the controlled by CA Dynam/T bit (X'10') is on in FLAG3.
4. The OUTCODE is set to hex zeros, &blank.VMS (blankVMS) or blanks.
5. The X'40' bit in FLAG3, indicating that the volume was released by an external vault manager, is on for tapes containing a valid OUTCODE.

Uncatalog Data Sets

As tapes are scratched, data sets written on the volume can be uncataloged from the MVS catalog, depending on the UNCATA parameter in member TMOOPTxx in hlq.CTAPOPTN. If the UNCATA parameter is set to YES, all data sets processed for scratch are uncataloged from the MVS catalog. If the UNCATA parameter is set to G, GDG data sets and data sets that are expired from Catalog Control or Cycle Control are excluded. If the UNCATA parameter is set to NO, no uncatalog processing takes place at all.

For this feature to work properly, TMSCLEAN must have access to all MVS catalogs.

The result of the uncatalog request is included in the report file. This report file can be used to produce an additional report of all files successfully uncataloged. In addition, the report file can be used as input to the TMSUNCAT utility that uncatalogs files on other systems. TMSCLEAN can execute in an environment without shared catalogs and helps maintain the integrity of the system catalogs.

TMSCLEAN and CA 1 Utilities

The execution of various CA 1 utilities can be required before TMSCLEAN, depending on the CA 1 keywords that are used to set retentions.

- TMSEXPDT can be executed to reset the hold criteria for the files that are eligible for RDS override. These retentions are established by dates defined in the Retention Data Set. Based on an installation option, TMSEXPDT can reset the expiration for all data sets or only those data sets that were created with the CA 1 default retention.
- TMSCTLG **may** be executed to determine which tape data sets under Catalog Control no longer have an entry on the MVS catalog if all systems are not yet at r11 or higher. The expiration date for these volumes is reduced to the current date plus any catalog extension default.
- TMSCYCLE **must** be executed to determine which tapes should be released from Cycle Control. The expiration date for these volumes is reduced to the current date plus any cycle extension default.

Note: For information about TMSCLEAN, see the *Utilities and Reports Reference Guide*.

Reports

A flat file in the CA 1 Retention Record format is created for use with CA Earl in a subsequent step.

Track CA 1 Tapes Created on a Non-CA 1 System

For tapes created on a non-CA 1 system and used as input under CA 1, the *Key Tape Procedure* allows control with minimal manual effort. To implement this procedure, determine the number of tapes which are needed during a certain period. When the TMSCLEAN program is executed, indicate the number of tapes that should be logged out-of-area to the individual locations. (Tapes can be selected based on density or label type.) As the tapes go scratch, they are automatically logged out-of-area.

The scratch tapes can then be placed in an area close to the non-CA 1 system. After the tapes are written on by the non-CA 1 system, they are filed in the tape library. Protection is maintained because CA 1 does not allow these scratch tapes to be used for output as long as they are marked out-of-area. When the tape is later used as input under CA 1, the TMC is automatically updated to contain the new data set name and expiration date. If EXPDT or RETPD is specified in the JCL, it is used. If it is not used, the system default KEYTAP parameter is used.

The default value of the KEYTAP parameter protects the tape for seven days from the date the tape was last used. All other pertinent TMC fields are updated, and the out-of-area code is removed. CA 1 then has complete control of the tape for tracking, reporting and protection purposes.

Back Up and Restore the TMC and Audit Data Set with TMSCOPY

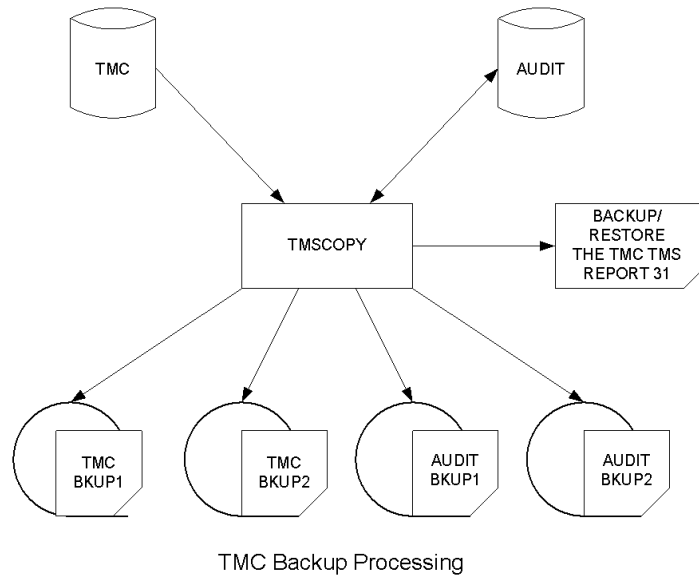
TMSCOPY is used to do the following:

- Back up the TMC and selected Audit data to tape. The Audit data set is a direct-access, wraparound file. Back up the file regularly, otherwise it fills up.
- Restore the TMC if it is compromised or destroyed.

TMC and Audit Data Set Backup

Back up the TMC and Audit data set at least once a day. Tape activity can continue while the backup is being performed. The TMC is backed up first, then the Audit records created since the previous backup are written to tape. Retain this data for at least 6 to 12 months.

The following is a flowchart illustrating the Backing Up the TMC and the Audit Data Set:



The THAUDT parameter in the TMOOPTxx member of *hlq*.CTAPOPTN establishes the warning threshold limit on the number of Audit records written before the first IEFTMS9 TMC BACKUP REQUIRED, AUDIT FILE UTILIZATION IS CURRENTLY *nn*% warning message is issued. The limit is a percentage of the total number of formatted Audit records. The system default is 80 percent. CA 1 utilities (excluding TMSCOPY and TMSINIT,) abends with a U0008 code as soon as the Audit threshold is reached.

The warning message is issued to the operating system console at each percentile point as the percentage utilization increases until the Audit data set is backed up or full capacity is reached. If the TMC is not backed up before the Audit data set reaches capacity, CA 1 causes all jobs except TMSCOPY to abend (S7yy) when tape mounts are requested.

You may want to have the JCL for the execution of TMSCOPY available in SYS1.PROCLIB (or another procedure library available to the system START command). Your operators can immediately start a backup in the event the IEFTMS9 TMC BACKUP REQUIRED, AUDIT FILE UTILIZATION IS CURRENTLY *nn*% warning message appears on the system console.

Restore the Audit Data Set

The Audit data set *cannot* be restored. In the event the Audit data set is damaged, format a new Audit data set with TMSFORMAT.

Note: For more information about the TMSFORMAT utility, see the *Utilities and Reports Reference Guide*.

Restore the TMC

The TMC is restored by reallocating the TMC (if necessary) and using the latest TMC backup tape. The Audit data set is then used as a transaction file to post all entries that have occurred since the last backup.

Important! A DASD RESTORE of the device holding the TMC may back level the TMC unless followed immediately by a TMSCOPY RESTORE of the TMC with the production Audit file. (The TMC should be allocated as unmovable, DSORG=PSU.) If disk volumes containing the TMC or Audit are reorganized, compacted, or compressed, exclude the TMC or Audit files. Do not run other tape jobs during the TMSCOPY RESTORE.

TMC/Audit Restoration

The Tape Management Catalog (TMC) needs to be restored if:

1. A full-volume restore has overlaid the TMC.
2. The TMC was incorrectly specified as eligible to be archived, was archived but was then RESTOREd from the archive backup.
3. Hardware problems (such as HDA failures and I/O errors) occur on the unit storing the TMC.
4. The TMC was not allocated as unmovable (DSORG=PSU) and failed in a DASD defragmentation or compaction operation.

Note: For more information about restoring the Tape Management Catalog (TMC), see TMSCOPY in the *Utilities and Reports Reference Guide*.

If similar problems occur on the Audit data set, reformat it with the TMSFORMAT utility.

Note: For more information about the process involved in reformatting the Audit data set, refer to the *Utilities and Reports Reference Guide*

TMC Backup and Restore Method for Disaster Recovery

At a disaster recovery site, CA 1 usually cannot initialize due to the lack of a valid TMC or Audit data set. In most instances, deactivate CA 1 to allow you to perform your data set restorations without interference.

During your restorations of the DASD, you can restore the volumes with the TMC and Audit files on them. Consider which is the more current copy of the TMC, the full volume backup or the TMSCOPY backup. A full volume backup of the TMC or a TMC restored from a TMSCOPY backup with the PARM of NOAUDIT can contain pointer errors. Resolve the errors before starting tape processing. If either of these methods are used, run TMSPTRS or the CTS task named APEC as soon as you have CA 1 active and correct the errors. Otherwise you can have tape jobsabend with a message IEFTMS70 3YY-116 indicating that the DSNB free chain is invalid.

The preferred method for the restoration of the TMC and Audit files requires you to run TMSCOPY to backup the TMC to tape. Then run an IEBGENER of the full Audit file to tape. The steps to accomplish this backup and restore are:

At production site:

1. TMSCOPY, PARM=BACKUP. This step will copy the TMC to tape.
2. IEBGENER the full Audit file to the same tape as used in Step 1.

At DR site:

1. IEBGENER the full Audit file from the correct backup tape to DASD.
2. TMSCOPY,PARM=RESTORE from the same tape as you used in Step 1.

This method eliminates the need to run TMSPTRS and correct pointer errors in the TMC before starting tape processing.

Report Description

To avoid the problems that can occur if the Audit data set or the DSNB allocation reaches capacity, carefully monitor TMS Report 31 - Backup/Restore the TMC File. Save the report for at least six weeks. In the event it becomes necessary to restore the TMC, the report assists in recovery.

Execute the Vault Management System

Many installations require some type of facility for the storage of critical tape data sets at a location other than the tape library. The storage unit can be a fireproof vault at the installation, or a location other than the computer site. The off-site storage facility enables you to protect valuable data in case of fire or disaster.

CA 1® Tape Management includes a Vault Management System (VMS) to automatically control, rotate, assign vault slot numbers, and report on vaulted (off-site storage) tape data sets. Execute the utilities that make up the CA 1® Tape Management Vault Management System as often as necessary to meet your site's guidelines for transferring tapes to off-site storage. Execute VMS only after all data sets to be moved have been created. For a discussion of VMS see the section [Using the Vault Management System \(VMS\)](#) (see page 81).

If you incorporate the VMS utilities (TMSVMEDT, TMSVMVLT, and TMSVMUPD) in your normal CA 1® Tape Management daily maintenance cycle, execute the utilities *after* TMSCLEAN has finished scratch processing. Expired tapes in the vault do not appear prematurely on the Scratch and Clean Listing before they have been physically returned to the tape library.

Run TMSCOPY before the VMS utilities, then move the alternate backup of the TMC (created by TMSCOPY) to the off-site location with your other data sets. (Provide the Vault Management System with the data set name of the alternate backup.)

Condition Code Checks

TMSVMVLT, the key utility in VMS, determines which data sets require movement and the location to which they are moved. TMSVMVLT creates a data set which the TMSVMUPD program uses and subsequent CA Earl steps. If TMSVMVLT terminates with a condition code higher than 4, execute the TMSVMUPD program and the CA Earl steps after the exception encountered by TMSVMVLT can be resolved.

Execute the TMSVMEDT utility which processes your Vault Pattern Description data set with no parameter options. If there is any error in the Vault Pattern Description data set (VPD), TMSVMEDT abends with a U4000-0006 code.

Test Vault Pattern Changes

Always test changes to your VPD before implementing. To perform VPD syntax validation without processing the TMC, execute TMSVMEDT with PARM=TEST. Testing the vaulting movements can be accomplished easily because TMSVMUPD is the utility that performs updates to the TMC to reflect any changes in the vaulting environment. TMSVMUPD is not executed or TMSVMUPD is executed with PARM=TEST. Updates to the TMC are suppressed but the Picking, Distribution, and Inventory reports can be generated to show the impact of your changes.

Reports

Reporting is done with CA Earl. Three sample CA Earl reports are included:

- Picking List
- Distribution List
- Inventory List

Report on Tape Status in the TMC

Various inventory, audit, specific inquiry, and customized reports can be created with CA 1 batch utilities and CA Earl. Typically, several of these programs are included in the normal CA 1 daily batch maintenance cycle.

For example, inventories of all active volumes by data set name and volume serial number, an audit report listing all CA 1 NOT SCRATCH dismounts, and perhaps an inventory of all tapes marked out-of-area can be created with the normal scratch list and vault management reports.

To generate these reports to a DASD file, the TMSRPT DD (used for some CA 1 output) must use a disposition of MOD. Some CA 1 utilities open and close the report file several times during the creation of the output.

TMSBINQ Utility—Print Volume, DSNB, and CA 1 Control Records

TMSBINQ performs batch inquiry on selected TMC Volume or DSNB records. You can also print control records. Control records are the first records in any TMC or Audit data set.

Note: For more information about control records, see the *Programming Guide*.

TMSBINQ can also report on the TMC record that is associated with any tape data set that has an entry on the MVS catalog.

Based on SYSIN control statements, the Batch Inquiry report can be produced in SHORT, LONG, or DUMP format for the TMC records. The default format is LONG. The SHORT format prints only the most significant fields, and the DUMP format provides a hexadecimal dump of the entire record image. Control records are printed in both hexadecimal and character format. These records are printed when you specify the HEADERS control statement.

TMSAUDIT Utility—Analyze TMC Activity

TMSAUDIT is an excellent tool for analyzing specific problems and tape processing trends in your environment. Any time a TMC record is updated, the updated image is placed in the Audit data set with a date, time, and code stamp. TMSAUDIT can report on the live Audit data set that reflects your most recent activity on the TMC, or on the backups of the Audit that are created during TMSCOPY execution for historical information.

You are advised to keep from 6 to 12 months of Audit backup data available for historical reporting and suspected error report analysis.

The Audit record types (or codes) generated are:

- Type 0 - reflects updates by CA 1 batch or on-line facilities
- Type 1 - reflects real-time input activity
- Type 2 - reflects real-time output activity
- Type 3 - reflects nonresident tape processing and other exceptions
- Type 4 - reflects NOT SCRATCH demount activity
- Type 5 - reflects CA 1 functional modification and initialization activity

Control statements allow you to select the Audit records for a specific VOLSER, a group of specific VOLSERS, a contiguous range of VOLSERS, or a given type of Audit record. You can specify the sort sequence by VOLSER, DSN, TIME (the default) or TYPE.

Depending on the control statements that are used, TMSAUDIT produces any of the following reports:

- Batch Update Audit Records (Type 0)
- Input Activity Audit Records (Type 1)
- Output Activity Audit Records (Type 2)
- Audit Exception Records (Type 3)
- Audit Not-Scratch Rejects (Type 4)
- Audit of Functional Modifications (Type 5)

TMSPTRS Utility—Maintain Multivolume/Multi-Data Set Chains

Execute TMSPTRS regularly to report on errors that may exist in the TMC multi-volume and multi-data set chains and in the unused DSNB chain. Certain types of errors in these chains can cause tape jobs to abend.

TMSPTRS generates TMS Report 29 - Invalid Pointers to describe any chaining errors that exist in the TMC.

Note: For more information about errors and the corrective action, see the *Message Reference Guide*.

You can also use the Automatic Pointers Error Correction (TMSAPEC) subtask of the CTS address space to inspect and repair problems in the multi-volume and multi-data set chains. TMSAPEC is documented in [CA 1 Automatic Pointers Error Correction](#) (see page 191).

CA Earl—Produce Customized Reports

Use CA Earl, the CA Easy Access Report Language, to design and produce customized reports. CA Earl provides methods for easy access to TMC Volume and DSNB record information.

Note: For more information about CA Earl, see the *Utilities and Reports Reference Guide* and the *CA Earl Reference Guide*.

Chapter 2: Real-time Tape Processing

This section contains the following topics:

[About Real-time Processing](#) (see page 41)

[How to Use Scratch Pool Management](#) (see page 42)

[Specific Mount Message Processing](#) (see page 48)

[Reject for Processing](#) (see page 49)

[Temporary Data Set Processing](#) (see page 49)

[Auxiliary Disposition Processing](#) (see page 49)

[NL and BLP Tape Processing](#) (see page 49)

[Nonresident Tape Processing](#) (see page 50)

[Alternate Internal Volume Substitution \(AIVS\)](#) (see page 50)

[Automated Replies to IBM WTORs](#) (see page 52)

[Reset Expiration Dates in the TMC in Real Time](#) (see page 53)

About Real-time Processing

CA 1 Real-time Processing includes the following features which enhance the tape processing facilities of the operating system:

- The Scratch Pool Management feature allows you to restrict nonspecific output access to tape pools of CA 1 controlled volumes and also allows the option of enabling real-time stacking.
- The Specific Mount Message option provides for message prefixing of specific output requests to indicate tape volume status in the TMC.
- CA 1 may not allow the use of a tape volume for input or output for various reasons.
- NL and BLP processing requires additional processing including the request for the volume serial number.
- Nonresident tape volume processing may require additional information from the operator.
- Catalog Interface keeps the CA 1 database updated with the current OS catalog status for each CA 1 controlled tape file.

How to Use Scratch Pool Management

The CA 1® Tape Management Scratch Pool Management feature allows you to protect defined ranges of tapes in the TMC by restricting nonspecific output access. Access can be restricted based on data set name, SMS Management Class name, creating job name, creating unit name, expiration date, temporary file status, or by specifying the pool name in the JCL.

Many applications require tapes with specific physical characteristics, such as tape length, quality of tape, reel size, type of container, and so on. Such requirements can be imposed by government agencies or internal business procedures or simply for operating convenience. The multiple scratch pool management capabilities of CA 1® Tape Management provide the necessary support to satisfy these requirements in a fully automated manner.

Each scratch pool can optionally be defined as a sub-pool that allows real-time stacking. While file stacking is common, real-time stacking is different. With real-time stacking, there is little control over what file sequence number or which volume within the scratch sub-pool the file is added to.

The READONLYnnnnn subpool definition is also allowed. If the first eight characters of the subpool name are READONLY (followed by any five characters), the tapes in that subpool can open for READ processing only. No output is allowed. A TMONSMxx rule should *never* point to a READONLYnnnnn subpool. Use this function when testing a peer-to-peer (grid) configuration of a virtual tape system and you do *not* want to allow any production tapes to be updated while at DR.

Note: Apart from READONLYnnnnn subpooling, CA 1® Tape Management does not support scratch subpooling in an OAM managed robotic environment. If READONLYnnnnn is used with an OAM managed OAM robot, all tapes in the same category must be part of the same subpool.

Store and Identify Scratch Tapes

Contiguously store scratch tapes with similar characteristics, such as those belonging to the same scratch pool, both in the tape library and in the tape pool area. All members of the same pool should be identified in the same manner, such as by color, strap type, or tape length. You can use different label colors for identification but labels are not easily visible from the side of the tape. You can distinguish cartridge media from reel media.

Multiple Tape Pools

Pulling scratch tapes from multiple tape pools requires more organization than the single pool approach. The Scratch and Clean (TMSCLEAN) program provides a file containing all scratch tapes. This file can be read as input to a CA Earl step to produce a scratch and clean list sorted by tape pool and label type within the tape pool.

Access to specific tape pools should be as convenient and efficient as possible. CA recommends that scratch tape pools be pulled and stored separately. If each tape pool is defined as a single range or within a single range, it is a simple process.

Scratch Pool and Access Rules

Scratch pools are assigned by adding control statements to two *hlq.CTAPOPTN* members:

- *TMOSCRxx* defines the tape pool name and the ranges of volumes that are assigned to the tape pool
- *TMONSMxx* defines the rules for access to a tape pool

TMSINITD reads these two members during CA 1 initialization. TMSINITD can be executed in batch mode to verify that any changes to these members are correct before CA 1 initialization.

Note: For more more information about the TMSINIT and TMSINITD programs, see the *Utilities and Reports Reference Guide*.

A scratch pool can be directly requested by specifying the scratch pool name in the JCL *VOL=SER=* parameter, for example:

```
//ddname DD DSN=tape.dsname,DISP=(NEW,CATLG),UNIT=cart,  
// VOL=SER=poolnm,LABEL=(1,SL)
```

Due to operating system edits on the VOL=SER= parameter, the specified scratch pool name must be six characters or less. If you use scratch pooling, at least one scratch pool assignment control statement must be specified in the TMONSMxx member. Assignment control statements are not necessary for scratch pools that are intended to be directly specified in the JCL if the TMONSMxx member references at least one scratch pool.

Notes:

- The specified unit must be able to satisfy at least one of the tapes from the specified pool.
- You may only specify *one* pool name in the VOL=SER= field, and no other volsers can be added in addition to the scratch sub-pool name.
- You cannot have two or more DD statements in the same step pointing to the same scratch subpool via the VOL=SER= specification. You may have two or more DD statements using VOL=SER=poolid, but they must be different pool names.

Once CA 1 is installed, you can maintain these members using the CA 1 ISPF facility that is described in the section [Using Online Inquiry and Update Facilities](#) (see page 95).

Restrictions

CA 1 assigns tape pools based only on the *first file* (primary data set) of the volume. Secondary data sets receive the same pool assignment as the primary data set for the same volume set.

In the case of multi-volume and multi-file data sets, the assignment for a new scratch volume is identical to the previous volume. The assignment can be traced back to the primary data set assignment on the first volume.

For multi-volume files, when a new volume is opened during output, the pool assignment is the same as for the preceding volume. If the assignment rules have been changed since the creation of the original volume, the assignment of the new volume is based on the new assignment rules.

The following hierarchy of scratch pool assignment is observed:

1. Specification of the scratch pool through the VOL=SER=pool JCL parameter
2. Assignment through an SMS Management Class
3. Assignment through External Data Manager rules
4. Assignment through the DSN, JOB, UNIT, RETPD and EXPDT parameters

Data sets which are not assigned to a tape pool by the assignment rules are automatically assigned to a generic tape pool. The generic tape pool contains all tapes that are not currently defined to a tape pool.

Real-time Stacking

You can allow each scratch sub-pool that is defined in the TMOSCRxx member of *hlq.CTAPOPTN* as allowing real-time (dynamic) file stacking. If allowed, there are a few controls that you can place at the individual sub-pool level. These options can be broken down into the following groups.

- The first group controls how many files or how much data can be put onto one volume.
- The second group controls which files should be stacked and what to do if stacking is no longer possible.

Note: Only files with a label type of SL are eligible to be real-time stacked. Temporary SL data sets are excluded.

Mount Messages

Once activated by TMSINIT, the CA 1 WTO subsystem passes control to CA 1 for all tape mount messages. If the mount request is nonspecific and requires a tape from a specific pool, the mount message is modified to include the tape pool identification (up to 13 characters). If the mount request is specific, the mount message reflects the location of the volume and whether it is a CA 1 controlled volume.

In all cases, a reissued operating system mount message is prefixed with a CA 1 message identification to notify the operator that the intercept is present and functioning. The original and reissued WTO appears on the SYSLOG and JESLOG; only the new WTO is displayed on the console.

On a system console, a non-pool mount message is prefixed with TMS001, while a pool-controlled mount is prefixed with TMS002.

Assume, for example, that a MINIREEL scratch pool has been assigned the range, MI0001-MI9999, and data sets using a high-level alias of MINI are authorized to write to the MINIREEL pool. When the mount request is issued, CA 1 modifies the mount message to read:

```
*TMS002 IEF233A M
05A0,MINIREEL,SL,6250,JOBNAME,STEP1,MINI.DATASET.NAME
```

For tape mounts not reflecting a data set name that is prefixed with MINI, the mount message reads:

```
*TMS001 IEF233A M
05A0,SCRATCH,SL,6250,JOBNAME,STEP1,NON.MINI.DATASET.NAME
```

JES3 messages use the prefix TMS004 for nonpool mount messages, and TMS005 for pool-controlled mounts.

Note: For CA 1 to properly intercept mount messages, it is required that the system console data set name monitor is activated. This is done by issuing the MN DSNAMES system console command. During TMSINIT processing a check is made to ensure the monitor is active. A WTOR prompts the operator if it is not.

If using silos, you must have an entry coded above the silo entry in the sub-system name table. The entry is TMS.

Scratch Pool Enforcement

When Scratch Pool Management is enabled, CA 1 provides enforcement for both pool and nonpool nonspecific output requests. If a data set to be created requires the use of a specific pool, but the volume mounted to satisfy the request is not part of the assigned ranges, CA 1 demounts the volume. CA 1 issues an IECTMS3 NOT SCRATCH message (code 80). The request is reissued.

If the request is not for a specific pool, but a volume is mounted that is assigned to a specific pool, CA 1 dismounts the volume and issues the IECTMS3 NOT SCRATCH message (code 84). The request is reissued.

If you have changed the scratch sub-pool rules and you are trying to MOD onto a tape, CA 1 continues to enforce using the same sub-pool as the first tape when requesting the mount of another tape.

3480/3490 Cartridge Tape Drive Support

A special exit point available on MVS/ESA 3.1.1 (JES2 only) systems and above enables you to better use the features of the 3480/3490 cartridge tape drive. CA 1 provides a sample exit that does the following:

- Displays the first eight characters of the tape pool name on 3480/3490 cartridge tape drives.
- Disables the automatic cartridge loader (ACL), if present, to prevent the ejection of all tapes in the ACL when a NOT SCRATCH condition occurs from Scratch Pool Management.

Note: For more information about this exit, see the *Installation Guide*.

Specific Mount Message Processing

If the mount message is for a volume that is CA 1 controlled, and the volume status in the TMC indicates that it has been checked out-of-area, the following mount message is issued:

```
*TMS007 IEC501A M 0E82,X00007(VLT1;000121),,JOBNAME,STEPNAME,DSNAME
```

The out of area code (VLT1) and slot number (000121) are displayed within the preceding message.

If the mount message is for a volume that is nonresident (not defined in the TMC), the following mount message is issued:

```
*TMS008 IEC501A M 0E82,X00001,,JOBNAME,STEPNAME,DSNAME
```

If the requested volume is CA 1 controlled and the volume status in the TMC indicates that it is in-house, the following message is issued:

```
*TMS009 IEC501A M 0E82,X00011,,JOBNAME,STEPNAME,DSNAME
```

If the mount message is for a volume that is CA 1/AIVS controlled and the volume status in the TMC indicates that it has been checked out-of-area, the following mount message is issued:

```
*TMS010 IEF233A M UUUU,extvol  
(AIVS=intvol),SL,,JOBNAME,STEPNAME,DSN
```

If the mount message is for a volume that is nonresident (not defined in the TMC), the following mount message is issued:

```
*TMS011 IEF233A x UUUU,extvol  
(AIVS=intvol)(outc;slotno)SL,JOBNAME,STEPNAME,DSN
```

If the mount message is a keep message for AIVS showing the internal and external volser, the following message is issued:

```
*TMS015 IEF233A x UUUU,extvol (AIVS=intvol),SL,JOBNAME,STEPNAME,DSN
```


Reject for Processing

For input and output processing, if CA 1 rejects the tape volume for use, CA 1 issues a console message IECTMS3. The message indicates the reason for the reject before the tape is demounted. If another tape volume can be used to satisfy the tape volume mount request, the mount message is reissued. If another tape cannot be used to satisfy the tape volume mount request, CA 1 issues a system abend S7xx-04. The xx is the CA 1 SVC number in hexadecimal after the reject message.

CA 1 can use the LCD display on a 3480 cartridge unit to display the NOT SCRATCH reason code when CA 1 rejects a cartridge for use. The CA 1 parameter MG3480 in the member TMOOPTxx in *hlq.CTAPOPTN* specifies whether this feature is used. Apply usermod CTSJUMSG to display the NO SCRATCH.

Temporary Data Set Processing

For output processing, the CA 1 message IECTMS9 is issued at OPEN before the tape volume being written to. If the tape or file is temporary, another IECTMS9 message is issued at CLOSE with WORK TAPE or WORK FILE within the message. Once work tapes are demounted, the operator can place them back in the scratch pool. CA 1 also issues this message for input processing when reading an expired volume or processing a file not recorded in the TMC. The option WRKFLS in TMOOPTxx allows temporary files.

Auxiliary Disposition Processing

The CA 1 Real-time Auxiliary Disposition performs several functions at CLOSE for input and output processing. The feature uses CA 1 console messages IECTMS8 to send messages to the operator, checks tape volumes out of area, and generates external tape label requests. A batch Auxiliary Disposition utility is also available.

Note: For more information, see [Using Auxiliary Disposition](#) (see page 67).

NL and BLP Tape Processing

When a tape volume is opened for input or output using NL or BLP processing, the CA 1® Tape Management message IECTMS1 requests the operator to supply the volume serial number. The data set can be cataloged to the operating system or accessed from the TMC. For a specific input or output volume request, the volume serial number must be the same as specified in the JCL or cataloged to the operating system. If not, CA 1® Tape Management assumes the wrong tape volume is mounted and has the tape demounted. For a nonspecific output request, enter the same volume serial number twice consecutively to ensure that you do not hit a wrong key.

Nonresident Tape Processing

If a tape volume was requested and the JCL specified LABEL=EXPDT=98000 or ACCODE=xCANORES, the volume is considered to be nonresident. For output, if the volume serial number matches one in the TMC, the CA 1 console message IECTMS2 requests the operator to verify that the volume is nonresident.

Nonresident tapes are protected with standard operating system type protection. For SL tape output, if the HDR1 expiration date is greater than the current date, the CA 1 console message IECTMS4 requests the operator to override the expiration date and allow the tape to be overwritten.

Alternate Internal Volume Substitution (AIVS)

With AIVS processing, CA 1® Tape Management can track foreign and duplicate volumes. Foreign volumes are created outside of CA 1® Tape Management control. Duplicate volumes have volume serial numbers which conflict with numbers already defined to CA 1® Tape Management. Duplicate volumes have been created on another CA 1® Tape Management system merging its TMC with an existing TMC.

AIVS support is provided in CA 1® Tape Management by recording the internal or actual volume serial number (ACTVOL) from the tape header. When an AIVS tape is to be opened for input processing, the JCL VOL=SER= parameter or system catalog entry for the file must refer to the external volume serial number. The mount message that is issued contains both the external and actual volume serial number that is obtained from the TMC volume record.

AIVS tape support is *read-only*. The tape allows foreign and duplicate volumes to be used and tracked by CA 1® Tape Management without copying to a CA 1® Tape Management controlled volume. If an AIVS tape is to be reused, it must be initialized with an actual volume serial number equal to the external volume serial number.

AIVS tapes are defined to CA 1® Tape Management either by directly entering the foreign tape information through a new ISPF facility or by executing the TMSMERGE utility. The utility merges another TMC with duplicate volume serial numbers.

Duplicate Internal Volume Serial Numbers

CA 1 can track tapes or cartridges that have duplicate internal volser. This situation may arise when the workload is moved from one system to another, multiple systems are consolidated, or a separate system needs to be absorbed into your production system. The merge phase (TMSMERGE) of a split/merge operation can assign a unique external volser to the tape/cartridge. This external volser is used in all mount messages, OS catalog, and reports. However, the internal (VOL1 record) is used during OPEN processing to mount the correct tape.

Once an AIVS volume goes scratch, reinitialize it before it can be used as a scratch tape. Once the tape is placed in scratch status, it is also placed in delete status. The scratch list produced by TMECLN01 can be modified to list all scratched AIVS volumes separately (simply uncomment out one line).

Foreign Tape Control

CA 1 can track and merge foreign tapes. Use the new ISPF panels to build (or assign) a CA 1 volser to a foreign tape. The CA 1 volser will become the external volser. When the tape is used for input processing, CA 1 resolves the difference in labels. The expiration date should be the date when you want the tape returned (or reinitialized to become an in-house tape).

Once a CA 1 controlled foreign tape is scratched, the CA 1 controlled volser (the external volser) is put into delete status. CA 1 controlled volser can be reused to control another foreign tape. To use the foreign tape as an in-house tape, add the volser. Then use TMSUPDTE or TSOTIQ to turn off the TMNRS indicator in FLAG4 and reinitialize the tape.

Automated Replies to IBM WTORs

CA 1 automatically replies to the following IBM messages:

IEC507D

This MVS message indicates that the EXPDT on the tape label is not past. CA 1 replies to this message if the tape is known to CA 1 and it is in scratch status. If it is a foreign tape CA 1 does not reply.

IEC534D

This MVS message indicates that the label on the tape is not compatible with the requested tape mount. Depending on your CTAPOPTN parms of LCHG, TCHG, and DCHG CA 1 either allows or disallows the change.

IEC704A

This MVS message is first issued as:

IEC704A L DDD, , SL, DEN/TRTCH, JOBNAME, STEPNAME.

A tape is mounted and does not contain a standard label for a SL request. Processing goes through OMODVOL1/EMODVOL1 first, where the VOLSER can be picked up from the tape label or the UCB. The operator is prompted for the VOLSER. If the VOLSER was picked up from the label or the UCB, and it is controlled by CA 1, it replies with the VOLSER.

This message can also be issued as:

nn IEC704A REPLY 'U, OWNER INFORMATION' OR 'M'.

This message is issued when changing a 36-TRACK cart back to an 18-TRACK tape. The message can also be issued for other track changes such as 256->128 or 384->256 or 384->128. If the VOLSER is controlled by CA 1, it responds with U or U, TMVENDOR.

Reset Expiration Dates in the TMC in Real Time

CA 1 provides an optional feature that allows you to assign expiration dates in real time. The Real-time Expiration feature uses the same matching logic as the batch utility TMSEXPDT, with the exception of SELECT=ALL processing. You can apply the expiration dates that are defined in the Retention Data Set (RDS) to tape files as they are being created. Each tape file being created goes through data set CLOSE processing. CA 1 accesses a memory resident RDS table to perform a lookup on any expiration rules matching the data set name being created. This feature is activated by adding a TMSRDS DD statement to the TMSINIT procedure that points to your installation RDS. When Real-time Expiration is activated on all systems sharing the TMC, you no longer need to run the batch TMSEXPDT utility daily.

Note: For more information about activating TMSINIT, see the *Utilities and Reports Reference Guide*.

Note: SELECT=ALL requires that the TMC be processed for all data sets that match the updated rule. You can perform this function using TMSEXPDT only. TMSEXPDT is supported for users who do not implement Real-time Expiration Dates in the TMC.

Chapter 3: Maintaining the Tape Library

This section contains the following topics:

[Access and Update the TMC](#) (see page 55)

[Initialize and Report on Tape Volume Labels](#) (see page 59)

[External Tape Label Production](#) (see page 61)

[Problem Analysis](#) (see page 63)

Access and Update the TMC

CA 1 provides batch utilities and online interfaces to query the status of any record or update any record in the TMC. Security is provided to prevent unauthorized inquiry or update requests.

Query and Update TMC Records

This section describes the CA 1 features that are used for specific Volume or DSNB record access.

The TMSBINQ utility allows you to report on the following:

- A specific volume serial number.
- A contiguous range of volume serial numbers.
- A specific DSNB record.
- A contiguous range of DSNBs.
- Any tape data set name that has an entry in your system catalog.
- TMC and Audit Control record information.

The TMSBINQ report can produce Volume or DSNB record information in one of the following formats:

SHORT

Only those fields which CA considers most significant are printed.

LONG

All fields in the record are listed. This format is the default.

DUMP

The record is printed in character and hexadecimal format. (Control record reports are automatically generated in both character and hexadecimal formats.)

Each field in the TMC Volume, DSNB and Control records is associated with a keyword name. The keyword name is used to access that field. You can access TMC Volume, DSNB and Control record fields using the keywords listed in the following table. TMC Control Record 1 and Control Record 3 are intentionally excluded.

Note: For more information about TMC record definitions and keywords, see the *Programming Guide*.

The TMSUPDTE utility allows you to update Volume records in the TMC. This program is used to update or verify fields in the TMC Volume record.

The TMSUDSNB utility allows you to update DSNB records in the TMC, and Control records in the TMC and Audit data sets.

Activate and Deactivate TMC Volume Records

When a TMC is created for the first time using TMSFORMT or TMSBLDVR, it is often created with an additional range or ranges of Volume records in anticipation of future library growth. The ranges are usually placed in *inactive* (DELETE) status to prevent the records from appearing on various CA 1 reports. The status also prevents nonresident tapes with the same VOLSER from appearing to be under CA 1 control.

Use the TMSDELET batch utility to do the following:

- Generate a list of Volume records in the TMC that are currently in inactive (DELETE) status.
- Activate (ADD) a Volume record or ranges of records that are currently in inactive (DELETE) status.
- Deactivate (DELETE) a Volume record or ranges of records that are currently in scratch status.

For TMSDELET to deactivate a record, the record must be in scratch status. Suppose, for example, that a tape was destroyed and is not replaced immediately. By executing TMSDELET, you can place the TMC Volume record in inactive status. You indicate to CA 1 that an entry exists in the TMC for this volume serial number but there is no tape volume in the tape library.

CA 1 online facilities have special functions that can ADD (activate) or DELETE (deactivate) TMC Volume records. If you are planning to activate or deactivate a range of volumes, use TMSDELET. The online function can only operate on a single volume record at a time.

Volume records can be activated automatically when you initialize the tape volume with the TMSTPNIT program. Once the status of a TMC Volume record has been changed from inactive to active, the tape volume is immediately available as a scratch tape.

Add TMC Volume Ranges or DSNB Records

The TMSXTEND utility is the preferred method of adding TMC volume ranges and DSNB records, unless selected nonstandard, alphanumeric volume serial numbers are in use in your installation.

Note: For more information about the TMSXTEND utility and the restrictions on nonstandard VOLSERS, see the *Utilities and Reports Reference Guide*.

The process of adding new records is called *extending* the TMC. The TMSXTEND utility extends the TMC with minimal disruption to tape processing. If you cannot run TMSXTEND because your installation uses non-standard tape volume serial numbers, the process requires additional planning. Before the extend operation, use the TMSFORMT utility and stop all tape processing and all CA 1 batch and online activity. Place CA 1 into a *batch active* mode.

The THDSNB parameter in the TMOOPTxx member of *hlq.CTAPOPTN* establishes the warning threshold limit on the number of DSNB records that are used before the first IEFTMS9 USED DSNBS HAVE PASSED THE THRESHOLD, CURRENT UTILIZATION IS *nn%* warning message is issued. The limit is a percentage of the total number of formatted DSNB records. The system default is 90 percent. Regularly review the statistics that are printed by the TMSCOPY program in TMS Report 31 to ensure that your DSNB allocation is not nearing capacity.

The warning message (IEFTMS9) is highlighted and non-deletable. This message is issued for each percentile change in use and each time TMSINIT is executed. As new IEFTMS9 messages are issued, the previous ones are deleted. When all DSNBs, except for the last one, have been used, all jobs allocating DSNB recordsabend with S2yy.

Important! Extending the TMC is a critical process which structurally modifies the TMC. The TMSXTEND and TMSFORMT procedures should be performed by the personnel responsible for the installation and maintenance of CA 1.

Remove Unused TMC Volume Ranges

If a contiguous range or ranges of TMC records are no longer in use, they can be removed using the TMSXTEND or TMSREMOV utilities. TMSXTEND is the preferred method of removing volume ranges because it can perform this function without stopping tape processing. You can be required to use TMSREMOV to delete unused volume ranges if you use nonstandard alphanumeric VOLSERS.

Note: For more information about the TMSXTEND utility and the restrictions on nonstandard VOLSERS, see the *Utilities and Reports Reference Guide*.

Before the execution of TMSREMOV, stop all tape processing and all CA 1 batch and online activity. Place CA 1 into a *batch active* mode.

Important! This critical process structurally modifies the TMC. The procedures for TMSXTEND and TMSREMOV are documented in the *Utilities and Reports Reference Guide*. The procedures should be only performed by the personnel responsible for the installation and maintenance of CA 1.

Maintain Multivolume and Multi-Data Set Chains

Analyze multi-volume, multi-data set, and DSNB free chains regularly to prevent possible CA 1 abends of your tape jobs.

The batch program, TMSPTRS, or the Automatic Pointer Error Correction (APEC) feature is used to detect and report any chaining errors in the TMC.

Note: For more information about TMSPTRS, see the *Utilities and Reports Reference Guide*. For more information about TMSAPEC (Automatic Error Correction), see [CA 1 Automatic Pointers Error Correction](#) (see page 191).

Incorporate TMSPTRS or the Automatic Pointer Error Correction (APEC) feature into your regular maintenance cycle on a weekly or monthly basis.

Initialize and Report on Tape Volume Labels

The CA 1® Tape Management utility TMSTPNIT coordinates the proper integration of new tape volumes into the library and the replacement of old volumes.

CTSDEU is useful for erasing sensitive data from tapes that have gone scratch. CTSTMAP is useful to report on the content of your tape header labels.

Note: For more information about CTSDEU and CTSTMAP, see the *Utilities and Reports Reference Guide*.

Initialize Tape Volumes with TMSTPNIT

TMSTPNIT initializes SL, NL, and AL tapes. TMSTPNIT checks the status of the TMC record for the volume serial number to ensure that it is in scratch or inactive status. Optional information for the OWNER and ACCESS fields of the header label can also be specified.

Existing labels on a cartridge are verified to help that ensure the volume is in the proper status before the cartridge is initialized.

The TMSTPNIT program is driven by control statements. The four categories of control statements are:

NEWTAPE

The volser is under CA 1 control and is being replaced with a new cartridge or reel. The Volume record in the TMC must be in scratch status. TMSTPNIT updates the BTHDATE (birth date) field to the date of initialization and set all eight read/write error count fields to zero.

OLDTAPE

The volser is under CA 1 control and needs to be reinitialized. The Volume record in the TMC must be in scratch status. The four fields containing read/write error counts since cleaning are set to zero.

OUTTAPE

The volser is not defined in the TMC (CA 1 does not control the volser).

INTAPE

The volser is under CA 1® Tape Management control and is to be initialized. The Volume record in the TMC reflects that the record must be in inactive (DELETE) and scratch status. Once the record is initialized, the inactive status is turned off and the BTHDATE (birth date) field is set to the date of initialization. All eight read/write error count fields are set to zero.

Process Header Label Information with TMSTPPRO

TMSTPPRO performs the following functions:

- Verifies the content of tape volume header labels
- Erases data from scratched volumes
- Updates the TMC with the content of the header labels (see Note)
- Creates TMC record images with the content of the header labels (see Note)

Note: The use of these functions is not recommended because tape header labels contain only the last 17 characters of the data set name.

TMSTPPRO operates as a started task, and continues to request tape mounts until the system STOP command is issued. The action terminates with the final mount of a volume once the STOP command is issued.

Note: Do not use this utility for tapes in a robotic device.

External Tape Label Production

External gummed labels are unnecessary in a CA 1 controlled environment, unless you mark tapes out-of-area for off-site processing. CA 1 provides many methods for requesting external labels: TMSLBLPR, a CA 1 system option, Auxiliary Disposition, online interfaces, and requests made directly to the Online Label Interface.

All external label requests are processed by the Online Label Interface. The interface provides one central point of control and ease in customizing labels.

More information:

[Common Tape System](#) (see page 187)

TMSLBLPR

TMSLBLPR is a stand-alone program that generates external tape label requests in either a single or multiple CPU environment. TMSLBLPR can be executed as a started system task or as a batch program. Scanning the Audit data set for records corresponding to a new output tape, TMSLBLPR extracts information for the label and sends the label request to the Online Label Interface for processing.

In a multiple CPU environment where the TMC and Audit data sets are shared, TMSLBLPR enables you to generate external labels for all of your CPUs from a single source.

CA 1 System Option

If the LAB parameter in the CA 1 system options member TMOOPTxx in *hlq.CTAPOPTN* is set to YES, CA 1 sends an external label request to the Online Label Interface every time a volume is opened for output.

Auxiliary Disposition

The CA 1® Tape Management Auxiliary Disposition feature enables tape users to selectively generate external label requests by adding the appropriate DD and control statements to the JCL which reads or writes the data sets during real-time processing.

Use the TMSDISP batch program to generate label requests after processing in a similar manner for cataloged data sets. The Auxiliary Disposition feature can also issue WTO messages to communicate post-processing requests.

Note: For more information, see Using Auxiliary Disposition.

Online Interfaces

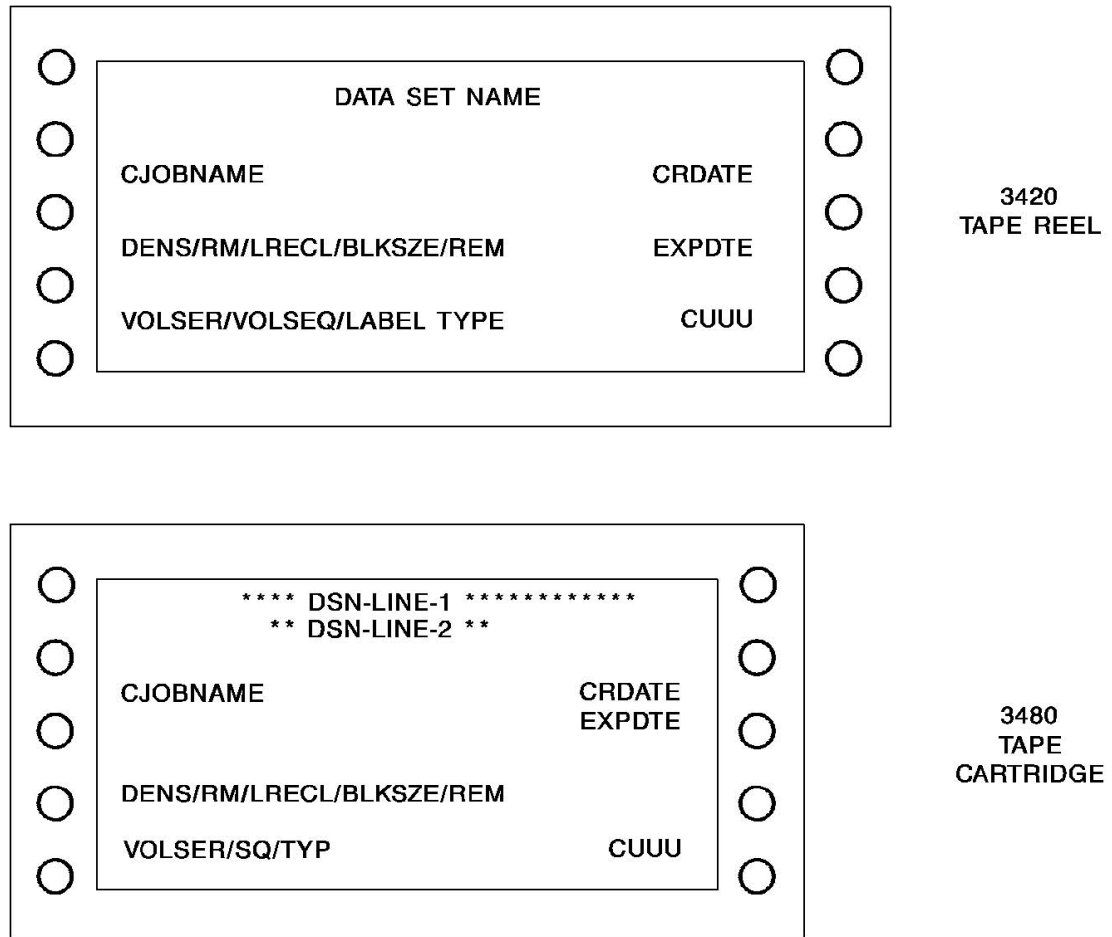
Any of the CA 1 online interfaces can be used to generate an external label request for any given volser.

Note: For more information, see Using Online Inquiry and Update Facilities.

Online Label Interface

An external label request can also be made directly to the Online Label Interface through CTS address space commands. The commands are discussed in the [Common Tape System](#) (see page 187).

The following diagram provides Label Format Examples:



Problem Analysis

This section provides information about analyzing problems with the tape library.

CA 1 NOT SCRATCH Demounts

If CA 1 determines that a tape mounted to satisfy an output request should not be written on, an IECTMS3 WTO is issued designating a two-digit, NOT SCRATCH reason code.

Note: For more information about reason codes, see the *Message Reference Guide*.

When NOT SCRATCH condition occurs, a Type 4 Audit record is written to the Audit data set. As part of your normal daily batch maintenance for CA 1, consider creating a report of the Type 4 Audit records. To the report, use the backup of the Audit data set that is created by TMSCOPY as input to the TMSAUDIT utility. Then pull the indicated volumes from the library for subsequent analysis.

In some cases, it is necessary to dump the header label of the tape to determine the nature of the problem. CA 1® Tape Management provides CTSTMAP for this purpose.

Why Tapes Do Not Get Scratched

If you want to scratch a tape using TMSCLEAN, meet all of the following conditions:

1. Execute TMSCLEAN with the SCRATCHLIST parameter and without the TEST parameter.
2. The expiration date in the TMC must be less than or equal to the system date on which TMSCLEAN is executed.
3. The X'01' bit indicating expired by an External Data Manager in FLAG2 is turned on if the controlled by EDM bit (X'20') or the controlled by CA Dynam/T bit (X'10') is on in FLAG3.
4. The OUTCODE is set to hex zeros, (blank)VMS (VMS) or blanks.

When the above conditions are met but a volume fails to be scratched, execute the TMSBINQ program with the DUMP option. You get a hexadecimal image of the volume record in question.

Often the problem is the OUTCODE containing decimal, and not hexadecimal, zeros (X'F0F0F0F0' as opposed to X'00000000'). An online CHECKIN command or a batch update with TMSUPDTE (specify OUTCODE=HEXZEROS in the control statement) resets the OUTCODE and allow TMSCLEAN to scratch the tape with its next execution.

Tape Failures

The following procedures are recommended if a tape has hard fails or excessive soft fails.

To be applicable to all users, these procedures do not distinguish between types of failures. Excessive soft fails tend to become hard fails quickly, and burst errors are sometimes accompanied by other serious problems on the tape. Use the following only as guidelines in establishing your own procedures.

Data Recovery

If the tape contains good data that must be recovered, typically determined after a system abend such as S001, S613, and on, perform the following steps:

1. Clean the tape (be sure to use a *nondestructive* cleaning process).
2. Clean the tape drive that is involved, regardless of how recently it was cleaned.
3. Attempt to read the data on the tape. If the read was successful, the suspect tape has a high probability of continued readability in the next few uses.
4. When the tape becomes a scratch, subject it to your full certification or replacement process.

Scratch Tapes

If the tape does not contain data which must be recovered and is a scratch tape, typically tapes cited as BAD on TMS Report 5 - Scratch and Clean Listing, perform the following steps:

1. Ensure that the tape is still in scratch status using the CA 1 online inquiry and update facilities (CA 1 ISPF or TIQ) or TMSBINQ.
2. Pull the volume from the library.
3. Subject the volume to your full certification process.
4. Depending on your guidelines, you can replace the old volume.
5. If the tape fails certification or you want to replace the physical volume, initialize a new volume and place it in the place of the old volume. Use NEWTAPE in TMSTPNIT.
6. If the tape passes certification and you want to retain the same physical volume, initialize the tape to its original volser using the OLDTAPE verb in TMSTPNIT.
7. Replace the tape in its slot in the library or scratch pool.

TMSAUDIT

Your Audit backup data, created by TMSCOPY, can help you analyze specific problems and general trends in your tape processing environment. It is a good idea to keep from 6 to 12 months of Audit data available at your site. The TMSAUDIT or CA Earl reporting utilities can be used to examine each change to a TMC record and show which CA 1 facility performed the change. These programs can also report on the *live* Audit data set currently active in your environment.

The following Audit Record types are useful:

Type 0

Shows all batch or online updates that are performed on a TMC record.

Type 1

Shows any activity when a volume has been read for input.

Type 2

Shows any activity when a volume has been OPENed and CLOSEd for output.

Type 3

Shows nonresident tape (LABEL=EXPDT=98000 or ACCODE=xCANORES in JCL) processing and security exceptions or before images if AUIDB4=ALL or if Batch is specified.

Type 4

Shows NOT SCRATCH demount activity.

Type 5

Shows CA 1 functional modification and initialization activity.

Each Audit record contains a date and time stamp. It is possible to identify every event in the TMC in chronological order. For example, if one of your tape users indicates that a tape was scratched prematurely, you can use the Audit data to show the original creation of the data set (Type 2), any input activity (Type 1), and the CA 1 facility used to expire the record before it was scratched by TMSCLEAN (Type 0).

Chapter 4: Using Auxiliary Disposition

This section contains the following topics:

[About Auxiliary Disposition](#) (see page 67)

[How to Use Auxiliary Disposition](#) (see page 67)

[Real-time Auxiliary Disposition](#) (see page 68)

[Batch Auxiliary Disposition](#) (see page 72)

About Auxiliary Disposition

The Auxiliary Disposition feature performs several functions which streamline special tape handling requests and eliminate most of the associated paperwork. This feature does the following:

- Communicates post-processing requests directly to the console operator or tape librarian.
- Generates external label requests for cartridges or reels.
- Automatically mark tapes out-of-area.
- Uncatalogs data set name entries from the system catalog.

How to Use Auxiliary Disposition

Auxiliary Disposition can be performed in either *real-time* or *batch* mode. Both modes require the use of a DD statement to reference either in-stream control statements or a data set containing the control statements. The default ddname that is supplied by CA 1 is TMSDISP. The ddname can be changed if your installation standards require the use of a different ddname.

Note: For more information about changing the DSN parameter, see the *Programming Guide*.

The write-to-operator (WTO) that is issued by Auxiliary Disposition contains the volume serial number that is associated with the data set being processed. The default route code of the messages is 13 but can be changed through control statement specification.

This message has the following format:

`IENTMS8 dddd,vvvvvv,sss,(area),message.text`

IENTMS8

Specifies the message ID.

dddd

Specifies the address of the tape device.

vvvvvv

Specifies the volume serial number.

sss

Specifies the volume sequence number.

(area)

Specifies the optional out-of-area code and, when used, is enclosed in parentheses for verification purposes.

message.text

Specifies the text that is copied from the control statement.

Real-time Auxiliary Disposition

Real-time Auxiliary Disposition processing is performed at CLOSE or EOVS for volumes, and requires the addition of the TMSDISP DD statement to the JCL step that processes the data set.

Job Control Statements

In the following example, in-stream TMSDISP control statements are used:

```
//STEP1    EXEC  PGM=CUTATAPE
//INPUT    DD   DSN=INPUT.DATA.SET.NAME,
//          DISP=SHR
//OUTPUT   DD   DSN=OUTPUT.DATA.SET.NAME,
//          DISP=(NEW,KEEP),UNIT=TAPE
//REPORT   DD   SYSOUT=A
//TMSDISP  DD   *
Auxiliary Disposition control statements here
/*
//
```

The following JCL references a fixed block, 80-byte, sequential data set:

```
//STEP1    EXEC  PGM=CUTATAPE
//INPUT    DD   DSN=INPUT.DATA.SET.NAME,
//          DISP=SHR
//OUTPUT   DD   DSN=OUTPUT.DATA.SET.NAME,
//          DISP=(NEW,KEEP),UNIT=TAPE
//REPORT   DD   SYSOUT=A
//TMSDISP  DD   DSN=CAI.CA1.AUXDISP,
//          DISP=SHR
//
```

The JCL can also reference Auxiliary Disposition control statements in a PDS member:

```
//STEP1    EXEC  PGM=CUTATAPE
//INPUT    DD   DSN=INPUT.DATA.SET.NAME,
//          DISP=SHR
//OUTPUT   DD   DSN=OUTPUT.DATA.SET.NAME,
//          DISP=(NEW,KEEP),UNIT=TAPE
//REPORT   DD   SYSOUT=A
//TMSDISP  DD   DSN=CAI.CA1.AUXDISP(PAYMASTR),
//          DISP=SHR
//
```

Message Control Statement

The Message control statement defines the format and content of messages, determines whether labels are printed, and defines the route code. This statement is required for Real-time Auxiliary Disposition processing and multiple statements are allowed.

The format for the Message control statement is:

ddname xrLOUT=xxxx,message.text

or

ddname xrLmessage.text

ddname

The ddname of the input or output tape. The ddname can be a maximum of eight characters and it must begin in position one.

x

The placement of any non-blank character in position 9 requires the operator to delete the WTO from the console.

r

Identifies the route codes to which the WTO is issued (the default is 13 and it must be in position 10). Possible values for r are:

- A = master console (route code 2)
- B = tape pool (route code 3)
- C = tape library (route code 5)
- * = SYSMSG only (route code 11)
- F = SYSMSG and console (route codes 2 and 11)
- G = SYSMSG and tape pool (route codes 3 and 11)
- H = SYSMSG and library (route codes 5 and 11)

L

An L placed in position 11 indicates that you want to generate an external label request for the volume being processed.

External label requests are processed by the Online Label Interface. For more information see Common Tape System.

M

An M placed in position 11 indicates that user data is present and should be passed to the Online Label Interface when the label request is generated. For more information about accessing user data when formatting a label, see [Common Tape System](#) (see page 187). You can specify up to 51 characters.

OUT=xxxx

xxxx is the one- to four-character out-of-area code that is placed in the TMC volume record. When used, it must be separated from the message text by a comma. OUT=xxxx must begin in position 12.

Note: For the out-of-area code to be maintained in the TMC, the volume must not be read for input once it has been marked out-of-area, or CA 1 automatically checks the volume back into the library. Always add the TMSDISP DD statement after the *last* DD statement to use the volume if the volume is to be marked out-of-area, or use Batch Auxiliary Disposition after you are certain that all processing has been completed.

message.text

Defines up to 60 characters of text to be placed in the IECTMS8 WTO. When an out-of-area code is specified, the text should begin in position 21; otherwise, the text should begin in position 12. If additional text is needed, specify the same ddname in another statement, use the same route code, and begin the text in position 12. A maximum of one hundred ddname statements is allowed for message text.

When the user data option is specified (an M in position 11), message.text defines up to 51 characters of text to be passed to the Online Label Interface with the label request. Additional ddname statements are ignored when user data is specified.

Real-time Auxiliary Disposition Example

A PDS named CAI.CA1.AUXDISP contains a member for job PAYMASTR. PAYMASTR uses the following JCL:

```
//PAYMASTR JOB (accounting),CLASS=A,MSGCLASS=A
//STEP1 EXEC PGM=PAYROLLM
//INPUT01 DD DSN=PAYROLL.WEEKLY,
//          DISP=SHR
//OUTPUT01 DD DSN=PAYROLL.MONTHLY.MASTER(+1),
//           DISP=(NEW,CATLG),UNIT=TAPE
//OUTPUT02 DD DSN=PAYROLL.MONTHLY.MASTER.BACKUP(+1),
//           DISP=(NEW,CATLG),UNIT=TAPE
//OUTPUT03 DD DSN=PAYROLL.IRS.MASTER(0),
//           DISP=MOD,UNIT=TAPE
//OUTPUT04 DD DSN=PAYROLL.SSA.MASTER(0),
//           DISP=MOD,UNIT=TAPE
//OUTPUT05 DD DSN=PAYROLL.FICA.MASTER(0),
//           DISP=MOD,UNIT=TAPE
//REPORT DD SYSOUT=A
//TMSDISP DD DSN=CAI.CA1.AUXDISP(PAYMASTR),
//          DISP=SHR
//
```

The PAYMASTR member in CA1.CA1.AUXDISP contains the following control statements:

```
INPUT01  A *** RETAIN VOLUME FOR THE NEXT JOB ***
OUTPUT01XALOUT=SUPV,*** DELIVER VOLUME TO SHIFT SUPV ***
OUTPUT02XA OUT=SAFE,*** DELIVER VOLUME TO SAFE STORAGE ***
OUTPUT03 A *** RETAIN VOLUME FOR THE NEXT JOB ***
OUTPUT04 A *** RETAIN VOLUME FOR THE NEXT JOB ***
OUTPUT05  M *** DELIVER VOLUME TO OFFSITE LOCATION ***
```

These results in the following Auxiliary Disposition processing actions:

1. A WTO is issued to route code 2 for the INPUT01 DD for volume 990082 on device 05A0:

IECTMS8 05A0.990082.0001. *** RETAIN VOLUME FOR THE NEXT JOB ***
2. A highlighted, nondeleteable WTO is issued to route code 2 for the OUTPUT01 DD:

IECTMS8 05A2.990133.0001 (SUPV).*** DELIVER VOLUME TO SHIFT SUPV ***

Volume 990133 on device 05A2 is marked out-of-area to an OUTCODE of SUPV, and an external label request is generated.
3. A highlighted, nondeleteable WTO is issued to route code 2 for the OUTPUT02 DD:

IECTMS8 05A3.990312.0001 (SAFE).*** DELIVER VOLUME TO SAFE STORAGE ***

In addition, the volume record for 990312 from drive 05A3 is marked out-of-area to an OUTCODE location of SAFE.
4. A WTO is issued to route code 2 for the OUTPUT03 DD and the OUTPUT04 DD:

IECTMS8 05A8.990819.0001 *** RETAIN VOLUME FOR THE NEXT JOB ***

IECTMS8 05A9.990015.0001 *** RETAIN VOLUME FOR THE NEXT JOB ***
5. The text "*** DELIVER VOLUME TO OFFSITE LOCATION ***" is passed to the Online Label Interface as user data. The user data can be used to generate a unique label for the volume associated with the OUTPUT05 DD.

If the Online Label Interface is not available at the time an external label request is made, the following console message is generated:

```
IEBTMS97-01 LABEL SEND REQUEST DID NOT COMPLETE SUCCESSFULLY
```

Batch Auxiliary Disposition

The CA 1 batch utility TMSDISP is used to perform Auxiliary Disposition processing. The batch mode of Auxiliary Disposition differs from the real-time mode in that the data set being processed must have an entry on the system catalog. The corresponding volume entries on the TMC must not be in scratch or inactive status. The system catalog entry can be uncataloged as part of Auxiliary Disposition processing. The TMSDISP program can be executed as the last step of a job, or as a job by itself. TMSDISP completes with a return code of 16 if any errors are encountered.

Job Control Statements

```
//AUXDISP EXEC PGM=TMSDISP
//STEPLIB DD DSN=CAI.CTAPLINK,DISP=SHR
//TMSDISP DD *
Auxiliary Disposition control statements here
/*
```

JCL Considerations

TMSDISP DD

This DD statement can reference an 80-byte sequential file or a member of a PDS containing the control statements to be processed.

Control Statement Specification

There are two control statements for TMSDISP:

- The DSN control statement identifies the tape volume by data set name.
- The Message control statement defines the format and content of messages.

Multiple control statements are allowed for each program execution.

Data Set Name Control Statement

The Data Set Name (DSN) control statement identifies the cataloged data set name that is processed. This control statement is required for BATCH Auxiliary Disposition Processing and precedes the optional Message control statement.

The DSN statement has the following format:

DSN=*data.set.name* [,UNCATLG] [, LABEL]

DSN=*data.set.name*

Describes the cataloged data set name to be processed. If it is a GDG, a relative generation specification is permitted. This must begin in position one.

UNCATLG

Allows you to uncatalog the data set entry from the system catalog. This parameter is optional.

LABEL

Generates an external label request. External label requests are processed by the Online Label Interface.

Note: For more information see Common Tape System.

Message Control Statement

The Message control statement used in Real-time Auxiliary Disposition on Message Control Statement is also used in Batch Auxiliary Disposition. The Message control statement is optional. However, three exceptions should be noted when using this control statement in batch mode:

1. The ddname does *not* have to match the ddname used in the JCL when the tape was read or written. The ddname can be any value.
2. The Message control statement is *not* used for external label generation. In batch mode, external label generation is requested through a parameter on the DSN control statement.
3. The user data option is not valid with Batch Auxiliary Disposition.

Batch Auxiliary Disposition Example

To perform the identical functions illustrated in the Real-time Auxiliary Disposition Example, the TMSDISP program could be executed as an additional job step or as a job by itself:

```
//PAYMASTR JOB (accounting),CLASS=A,MSGCLASS=A
//STEP1 EXEC PGM=PAYROLLM
//INPUT01 DD DSN=PAYROLL.WEEKLY,
//          DISP=SHR
//OUTPUT01 DD DSN=PAYROLL.MONTHLY.MASTER(+1),
//            DISP=(NEW,CATLG),UNIT=TAPE
//OUTPUT02 DD DSN=PAYROLL.MONTHLY.MASTER.BACKUP(+1),
//            DISP=(NEW,CATLG),UNIT=TAPE
//OUTPUT03 DD DSN=PAYROLL.IRS.MASTER(0),
//            DISP=MOD,UNIT=TAPE
//OUTPUT04 DD DSN=PAYROLL.SSA.MASTER(0),
//            DISP=MOD,UNIT=TAPE
//REPORT DD SYSOUT=A
//STEP2 EXEC PGM=TMSDISP
//STEPLIB DD DSN=CAI.CTAPLINK,DISP=SHR
//TMSDISP DD DSN=CAI.CA1.AUXDISP(PAYMASTR),
//          DISP=SHR
//
```

The control statements in the PAYMASTR PDS member are:

```
DSN=PAYROLL.WEEKLY
INPUT01  A *** RETAIN VOLUME FOR THE NEXT JOB ***
DSN=PAYROLL.MONTHLY.MASTER(0),LABEL
OUTPUT01XA OUT=SUPV,*** DELIVER VOLUME TO SHIFT SUPV ***
DSN=PAYROLL.MONTHLY.MASTER.BACKUP(0)
OUTPUT02XA OUT=SAFE,*** DELIVER VOLUME TO SAFE STORAGE ***
DSN=PAYROLL.IRS.MASTER(0)
OUTPUT03 A *** RETAIN VOLUME FOR THE NEXT JOB ***
DSN=PAYROLL.SSA.MASTER(0)
OUTPUT04 A *** RETAIN VOLUME FOR THE NEXT JOB ***
```

The ddname field of the Message control statement can be any value for batch Auxiliary Disposition; the ddname does not have to be the actual ddname used in the PAYMASTR JCL.

If the Online Label Interface is not available at the time an external label request is made, the following console message is generated:

```
IEBTMS97-05 LABEL REQUEST FOR xxxxxx FAILED - ONLINE LABELS INTERFACE
INACTIVE
```


Chapter 5: Synchronizing the TMC and MVS Catalog

This section contains the following topics:

[TMC and the Catalog](#) (see page 77)

[How to Use Catalog Control](#) (see page 77)

[TMSCLEAN Uncatalog Processing](#) (see page 78)

[TMSOSCAT](#) (see page 79)

TMC and the Catalog

Although CA 1 is the final authority on the status of your tape data sets, many operating system users rely on the system catalog to determine which volume should be mounted to process a file.

By carefully monitoring the TMC and system catalog databases, you should be able to reduce the number of abends that may occur when the system catalog and the TMC differ on the content of a volume. In addition, changes to JCL can be kept to a minimum.

How to Use Catalog Control

The most effective way to maintain your tape data sets and keep the entries in the system catalog synchronized with the TMC is to use the CA 1 Catalog Control feature.

Catalog Control Processing

CA 1 does not allow a volume under Catalog Control (CATLG or CATALOG in control statements and online interfaces, or LABEL=EXPDT=99000 or ACCODE=xCACATLG in JCL) to be scratched as long as an entry on the system catalog matches the data set(s) on the volume. If the system catalog does not have a corresponding entry, the expiration date for the tape volume is reset to the current date plus any catalog extension.

Proper use of Catalog Control can eliminate job abends that may occur when the system catalog and the TMC differ on the content of a volume. If the system catalog indicates that data set ABC is on volume 123456, but the volume actually contains data set XYZ because the volume was scratched earlier and reused under CA 1 control, an abend is likely to occur.

Catalog Control processing is performed by either the batch utility TMSCTLG or by TMSCLEAN.

TMSCLEAN Uncatalog Processing

When a volume is scratched, TMSCLEAN has the ability to remove the associated data set(s) entry from the system catalog using operating system catalog macros.

The UNCATA parameter in the system options member, TMOOPTxx in *hlq.CTAPOPTN*, specifies which data sets TMSCLEAN can uncatalog. A value of YES for this option allows TMSCLEAN to uncatalog all data sets. If the value is set to GDG, GDG data sets and data sets expired from Cycle Control or Catalog Control are excluded. A value of NO prevents any uncatalog operations when a tape is scratched by CA 1.

Note: For more information about CA 1 system options and instructions for changing the options, see the *Programming Guide*.

Another aspect in keeping the TMC and system catalog synchronized involves TMSCLEAN scratch processing. The report file produced by TMSCLEAN can be used as input to the TMSUNCAT utility on other systems in an environment that does not share catalogs. The report file helps synchronize the system catalogs on all CPUs while executing TMSCLEAN on one CPU.

TMSOSCAT

TMSOSCAT is a reporting utility that identifies possible discrepancies between the TMC and the system catalog. This program allows the sort sequence to be specified in a control statement. The report is sorted *either* by data set name, volume sequence number, file sequence number, volume serial number, and type of record, *or* by volume serial number, volume sequence number, file sequence number, data set name, and type of record.

TMC fields reported include data set name, volume serial number, volume sequence number, file sequence number, expiration date and an error code. The operating system catalog fields reported include data set name, volume serial number, volume sequence number and file sequence number. An error code is the last field and indicates a discrepancy. The last page of the report is an error summary that provides totals and a description of each error code.

TMSOSCAT can be used to set the new TMISCAT and DNSBISCA on all files currently cataloged. This usage of TMSOSCAT should be done *once* on each system sometime after upgrading to a r11 version. Until this has been done on all systems sharing the TMC, the new TMOOPT option, OCATLG, should be left as YES (the default). Once TMSOSCAT has been run to synchronize the operating system catalogs and the TMC on all systems sharing the TMC and Audit files, the OCTLG option can be changed to NO.

To run TMSOSCAT so that the new flag bits are set correctly, no IDCAMS or IEHLIST input is required or accepted and the SYSIN control statements are ignored. Run TMSOSCAT with a PARM of RESYNC,DETAIL on one system. When this is complete, run TMSOSCAT with a PARM of SYNC,DETAIL on all other systems. When all other of the TMSOSCAT utilities are complete you can change the OCTLG option to NO and remove the TMSCTLG (and EARL) step(s) from your daily CA 1 job stream.

You may want to consider executing this program on a regular basis to help keep the two databases synchronized.

Note: For more information about TMSOSCAT, see the *Utilities and Reports Reference Guide*.

Chapter 6: Using the Vault Management System (VMS)

This section contains the following topics:

[VMS Features](#) (see page 81)

[VMS Overview](#) (see page 84)

[Execute VMS](#) (see page 85)

[Vault Reports](#) (see page 88)

[Vaulted Tapes](#) (see page 91)

VMS Features

Storage of tape media in off-site locations is vital to a data center. The CA 1 Vault Management System (VMS) provides the means to automatically generate reports describing the necessary movement of volumes to and from off-site locations once your critical tape data sets have been created. A Vault Pattern Description (VPD) data set describes to VMS those files that require movement based on your requirements.

Note: For more information about the JCL and control statements and sample output and the exception messages generated by each VMS utility, see the *Utilities and Reports Reference Guide*.

3480 Cartridge Sub-vaulting

Sites that have data sets on 3420 reel media and cartridge media can use VMS facilities to automatically sub-vault the different formats into the same off-site location. Define any number of locations, each of which supports an unlimited number of reels and cartridges.

On-Site VMS Protection

VMS provides pseudo-vaults, which are the locations LIBR and (blank)LIB. These OUTCODEs use the normal slot for the volume in the tape library and have special meaning to VMS; tapes in these locations are not eligible for scratch until VMS releases control. Additionally, tape data sets vaulted to the LIBR OUTCODE cannot be recreated. The LIBR and (blank)LIB locations are frequently used for prestaging, delaying movement of a tape from the library while keeping the tape under VMS protection.

Note: The character (blank) denotes a blank.

Extensive Reporting

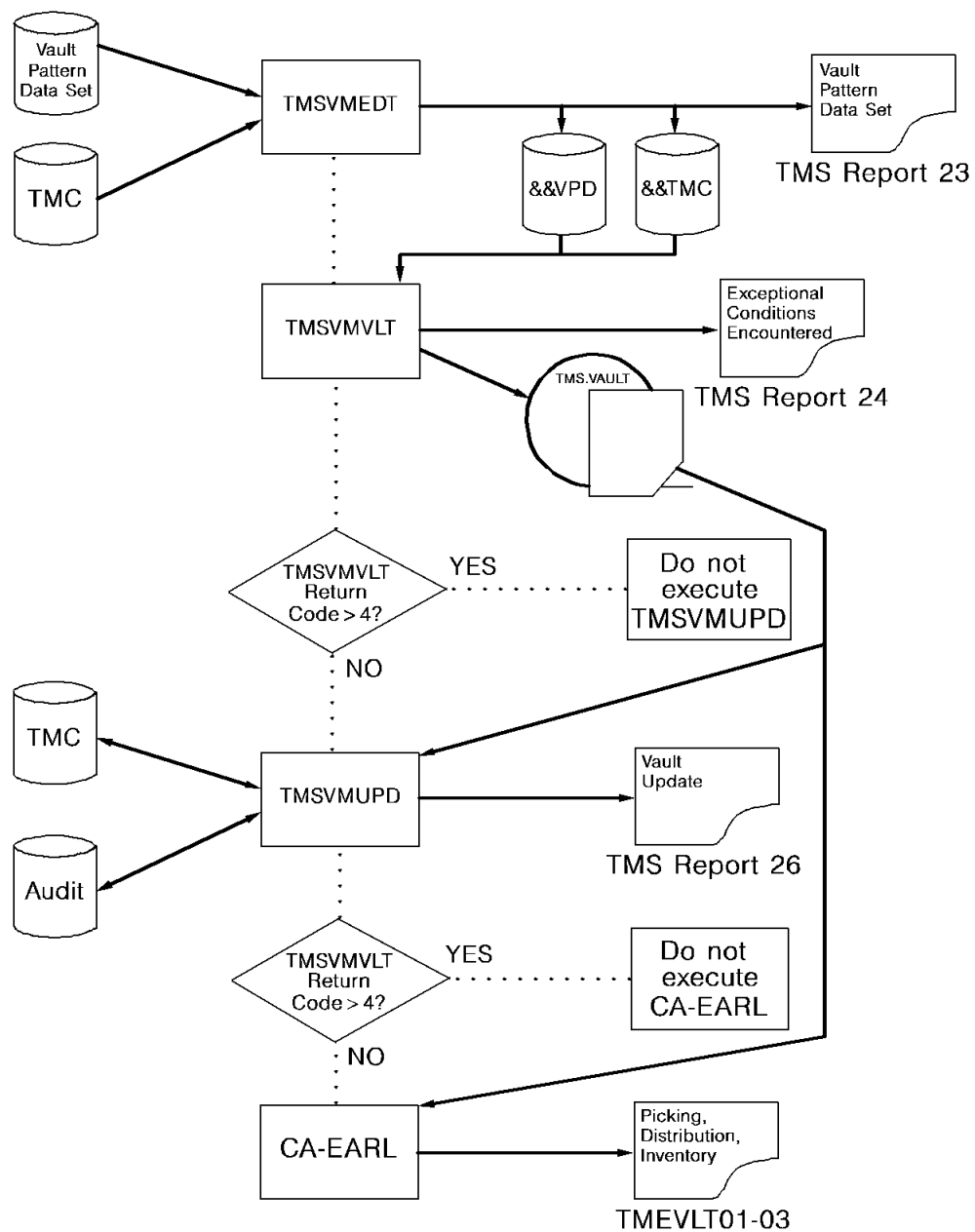
Three reports with site-specified sorting can be generated by three CA Earl steps:

- Picking List for Vault - Describes the volumes you will remove from the library and the off-site vaults.
- Distribution List for Vault - Shows the placement of tapes into a new location or the library.
- Inventory List for Vault - Describes the current location of each tape under VMS control, including (blank)LIB and LIBR.

Reports are also provided regarding the vault pattern used and any exceptions or errors encountered during processing:

- Vault Pattern Data Set - Lists the VPD read by TMSVMEDT and provides a cross reference showing the pattern number assigned and the options requested for each.
- Exceptional Conditions Encountered During Processing - Lists exceptions detected by TMSVMVLT to inform you of any problems or possible errors with the processing.
- TMS Vault Update Listing - Shows each TMC record updated by TMSVMUPD.

The following diagram provides an illustration of a VMS Program Execution:



VMS Overview

VMS consists of the following batch utilities:

TMSVMEDT

TMSVMEDT is the first utility executed in the VMS cycle. It processes the VPD, validating and cross-referencing the site-defined movement data to generate a VPD work file. TMSVMEDT then selects the TMC records eligible for VMS control and generates a TMC work file. TMS Report 23 - Vault Pattern Data Set is created to describe the VPD and any errors it contains.

Note: TMSVMEDT should be executed only after all tapes to be vaulted have been created. Otherwise, movement may not be complete, especially when a multivolume data set is still in creation. When TMSVMEDT selects the TMC records, only those volumes that have already been created and the volume still on the drive can be moved. Volumes created subsequently are not moved until the next VMS execution. However, a second execution of TMSVMEDT will pick up those volumes created subsequent to the first execution.

The CYD parameter in member TMOOPTxx of *hlq.CTAPOPTN* determines cycling options. A PARM=TEST can be used to validate new entries prior to implementation. Optionally, a maximum number of slots may be specified at the VAULT level. TMSVMVLT indicates when this level has been reached. The ADGDG option allows ABARs pseudo GDG files to be treated the same as normal GDGs during the vaulting process.

TMSVMVLT

TMSVMEDT passes the VPD information and the selected TMC records to the TMSVMVLT program, which determines the movement of volumes between locations. TMSVMVLT creates a data set describing this movement for CA Earl to use in reporting, and for the TMSVMUPD program to use in updating the TMC. The VMSOUT data set created by TMSVMVLT should be saved for a reasonable period in the event the reports generated by CA Earl need to be reprinted. TMS Report 24 - Exceptional Conditions Encountered During Processing is produced to inform you of any problems or possible errors associated with the movement of volumes.

TMSVMUPD

Using the data from TMSVMVLT, TMSVMUPD updates the TMC to reflect the new vaulting configuration. TMSVMUPD compares the data set created by TMSVMVLT against the TMC and, if a change is present in the input, performs the update. If the data set name has been changed or the tape has been marked as scratch in the interim between the execution of TMSVMVLT and TMSVMUPD, that TMC record is not updated, and an error message is issued. The updates and any errors are listed on TMS Report 26 - Vault Update Listing.

CA Earl

Using the data from TMSVMVLT, CA Earl can generate the Picking List, Distribution List, and Inventory List. These reports detail the decisions made by TMSVMVLT regarding tape volume movement. TMEVLT01-03 are used to produce these reports. Site-specific CA Earl control statements determine the sort sequences and optional report headings.

Execute VMS

VMS should be executed whenever you have created tapes to be vaulted and are ready to be moved. If VMS is included in the normal CA 1 daily maintenance cycle, it should be executed *after* the TMSCLEAN utility has finished scratch processing. This prevents expired tapes that are being returned from a vault from appearing on the Scratch and Clean List before they have been physically returned to the library.

We recommend that you back up the TMC using TMSCOPY before executing VMS, so the secondary TMC backup created by TMSCOPY can be vaulted immediately.

When TMSVMEDT is executed without a PARM, a U4000-0006 ABEND is issued if any errors are present in the VPD. TMSVMEDT can be executed with a PARM of TEST to validate the VPD without processing the TMC prior to implementing vault pattern changes.

A condition code check should be used to suppress execution of TMSVMUPD if the TMSVMVLT step ends with a return code greater than 4. The TMC is not updated in this instance, and the movement reports are not produced.

Changes to the VPD can be tested by not executing the TMSVMUPD program, or by using PARM=TEST for the TMSVMUPD execution. Updates to the TMC are suppressed but the Picking, Distribution, and Inventory reports can be generated to show the changes.

Tape Volume Selection Criteria

VMS uses the following criteria to select TMC records for processing:

1. Selection Based on File 1 - Selection of a volume by VMS is based on the first file on the tape. VMS does not process volumes other than those reflecting the first data set. If the first file is a multivolume file, or part of a multivolume/multifile *aggregate*, all volumes associated with file 1 are moved.
2. OUTCODE Reflects VMS Location - In addition, if the OUTCODE field in a volume has been manually updated to reflect a VMS location, including (blank)LIB or LIBR, VMS selects the volume even if there is no pattern entry in the VPD. The OUTDATE field must also be manually set.

For nonrecurring or special classes of volumes, the CA 1 Auxiliary Disposition feature, or the online batch or batch update programs, can be used to assign a vault name to the out-of-area code field in the TMC record. VMS processes these records to assign slot numbers and includes them on the reports. They are then shown as residing in the vault until they are expired, when they appear on the Picking List to be returned to the library. Even though these tape volumes are assigned a vault name, they are not under vault control until VMS is executed.

3. Specific volumes or Range of volumes may also be selected. However, in this situation movement is only to one location and the volume(s) selected will remain at that location until the rule is removed.

Tape Volume Rejection Criteria

VMS uses the following criteria to reject TMC records for processing:

1. Scratch, Inactive (DELETE), or ABEND Status - If the FLAG1 value in the TMC volume record reflects scratch (X'04') or inactive (DELETE) (X'02') status, VMS does not select the volume for movement. If a job creating a VMS-controlled data set abends, the FLAG1 value contains X'10'. VMS does not move a tape with this value unless ABEND is coded on the DSN statement.

Note: Not all abends perform normal CLOSE processing. Examples are Sx22, Sx13, and Sx37. In these instances, the FLAG1 ABEND indicator is not set, and manual intervention is required to prevent VMS from controlling the volume.

2. Expired by CA 1, CA 11 Restart and Tracking, or EDM - VMS does not select a volume when the FLAG2 field in the TMC Volume record has the X'01' bit set and the first byte of the DSN is hexzeros. This bit is set for the reason listed below:
 - CA 1 expired the volume due to I/O errors during header label creation.
3. Volume OUTCODEd to Non-VMS Location - If the volume record being processed by VMS has an OUTCODE value other than those defined to VMS, it is not selected.
4. Volume Released from VMS Control (OUTCODE= (blank)VMS) - When VMS releases control of a volume, or if a VMS-controlled tape is manually checked back into the data center and not read for input, the OUTCODE keyword of (blank)VMS is stored in the volume record. VMS does not select records with this OUTCODE unless the REDEFINED keyword has been specified on the DSN statement in the VPD. Once expired, these volumes are eligible for scratch.

Exported Virtual Volume Considerations

Virtual volumes that have been exported or externalized to a physical volume are now processed by the vaulting system. Once the VTX bit x'20' of FLAG5 has been turned on in the TMC record, virtual tapes continue to be moved according to the VPD rule, but slot assignment is bypassed. The slot number is set to a dummy value of 9,999,999 (a "virtual slot"). This virtual-slot number of 9,999,999 will be displayed on any ISPF display, or the TI display and the Vantage/GUI display.

These exported virtual volumes do not appear on picking or distribution list (because there is no physical cartridge to move). They are displayed on the inventory list with the physical container volume (ACTVOL) in place of the slot number preceded by an asterisk (*). So, on the inventory list the slot number might be displayed as *C12345 when C12345 is the container volume that the virtual volume has been exported onto.

Vault Reports

Vault Management reporting, tape movement and vault inventory can be produced using supplied CA Earl members. The output data set produced by the utility TMSVMVLT is used as input to CA Earl to generate the Picking List for Vault, the Distribution List for Vault, and the Inventory List for Vault.

The CA Earl members provided in the source library to create the vault reports (TMS Report-25) are:

- TMEVLT01 - Picking List for Vault
- TMEVLT02 - Distribution List for Vault
- TMEVLT03 - Inventory List for Vault

The reports generated by these CA Earl members reflect the decisions made by TMSVMVLT regarding tape volume movement. The CA Earl members can be customized to meet site-defined sort sequences and report headings. The parameters passed to the vaulting utilities TMSVMEDT and TMSVMVLT can be passed to CA Earl and printed on the report.

TMEVLT01 produces the Picking List for Vault, and reflects those volumes that should be moved from an off-site location or from the tape library. A sample of this report is provided in the section Picking List for Vault.

TMEVLT02 can generate the Distribution List for Vault, and shows where volumes should be placed in a new off-site location or in the library. A sample of this report is provided in the section Distribution List for Vault.

TMEVLT03 creates the Inventory List for Vault, and shows the current location of volumes under VMS control, including volumes in the LIB and LIBR in-house vaults. A sample of this report is provided in the section Inventory List for Vault.

In each of these CA Earl members (TMEVLT01, TMEVLT02, and TMEVLT03), there is a section of commented code describing how to define report heading text that is associated with a specific vault name. This text is printed on the second line of the report. For complete details, see `***Vault name statements to define Vault Desc. in the Report Header**` within each of the CA Earl members. Enter the vault name in all caps.

Picking List for Vault

The following example shows sorting by VOLSER for ease in pulling tape cartridges from a particular location, which in this case is the tape library.

A Picking List sorted by SLOT is useful in the off-site locations.

WEDNESDAY , AUGUST DD, YYYY.222		YOUR COMPANY NAME							PAGE	1
		PARM=PARMUPD=TEST								
TMS REPORT-25		PICKING LIST FOR VAULT (LIBR) CARTRIDGE								
VOLSER	SLOT	DATA SET NAME	NEW VAULT	NEW SLOT	VOL EXPIRATION SEQ DATE	LBL TYPE	CREATE JOBNAME	CREATE DATE	CREATE TIME	PATT USED
061000		ASM2.ARCH.MASTER	DRA1	001	001 PERMANENT	SL-36	ASM2ARCH	YYYY/175	00:00:00	0002
061001		ASM2.BKUP.MASTER	DRA1	002	001 PERMANENT	SL-36	ASM2BKUP	YYYY/175	00:00:00	0002
061004		ADMIN.PAYR.DYL.G0406V00	SAFE	001	001 CATALOG	SL-36	ADMINPDY	YYYY/196	00:00:00	0001
061005		ADMIN.PAYR.WLY.G0088V00	SAFE	002	001 CATALOG	SL-36	ADMINPWY	YYYY/197	00:00:00	0001
061006		ADMIN.PAYR.PAYR.DLY.G0405V00	SAFE	003	001 CATALOG	SL-36	ADMINPDY	YYYY/198	00:00:00	0001
061007		ADMIN.PAYR.DLYBKUP.G0406V00	SAFE	004	001 CATALOG	SL-36	ADMINPDY	YYYY/199	00:00:00	0001
061008		ADMIN.PAYR.DLYBKUP.G0405V00	SAFE	005	001 CATALOG	SL-36	ADMINPDY	YYYY/198	00:00:00	0001
061009		ADMIN.PAYR.MLY.G0045V00	SAFE	006	001 CATALOG	SL-36	ADMINMY	YYYY/211	00:00:00	0001
		8 TAPES								
		8 TAPES								
GRAND TOTAL		8 TAPES								

Distribution List for Vault

In this example the Distribution List was sorted by SLOT for use in off-site locations. A Distribution List sorted by VOLSER is useful for those volumes being returned to the library.

WEDNESDAY , AUGUST		DD, YYYY.222		YOUR COMPANY NAME					PAGE	1
				PARM=PARMUPD=TEST						
TMS REPORT-25		DISTRIBUTION LIST FOR VAULT (DRA1) CARTRIDGE								
VOLSER	SLOT	DATA SET NAME	OLD	OLD	VOL EXPIRATION	LBL	CREATE	CREATE	CREATE	PATT
	NUMBER		VAULT	SLOT	SEQ DATE	TYPE	JOBNAME	DATE	TIME	USED

061000	001	ASM2.ARCH.MASTER	LIBRARY		001 PERMANENT	SL-36	ASM2ARCH	YYYY/175	00:00:00	0002
061001	002	ASM2.BKUP.MASTER	LIBRARY		001 PERMANENT	SL-36	ASM2BKUP	YYYY/175	00:00:00	0002
2 TAPES										
WEDNESDAY , AUGUST		DD, YYYY.222		YOUR COMPANY NAME					PAGE	2
				PARM=PARMUPD=TEST						
TMS REPORT-25		DISTRIBUTION LIST FOR VAULT (SAFE) CARTRIDGE								
VOLSER	SLOT	DATA SET NAME	OLD	OLD	VOL EXPIRATION	LBL	CREATE	CREATE	CREATE	PATT
	NUMBER		VAULT	SLOT	SEQ DATE	TYPE	JOBNAME	DATE	TIME	USED

061004	001	ADMIN.PAYR.DYL.G0406V00	LIBRARY		001 CATALOG	SL-36	ADMINPDY	YYYY/196	00:00:00	0001
061005	002	ADMIN.PAYR.WLY.G0088V00	LIBRARY		001 CATALOG	SL-36	ADMINPWY	YYYY/197	00:00:00	0001
061006	003	ADMIN.PAYR.PAYR.DLY.G0405V00	LIBRARY		001 CATALOG	SL-36	ADMINPDY	YYYY/198	00:00:00	0001
061007	004	ADMIN.PAYR.DLYBKUP.G0406V00	LIBRARY		001 CATALOG	SL-36	ADMINPDY	YYYY/199	00:00:00	0001
061008	005	ADMIN.PAYR.DLYBKUP.G0405V00	LIBRARY		001 CATALOG	SL-36	ADMINPDY	YYYY/198	00:00:00	0001
061009	006	ADMIN.PAYR.MLY.G0045V00	LIBRARY		001 CATALOG	SL-36	ADMINMY	YYYY/211	00:00:00	0001
6 TAPES										
GRAND TOTAL		8 TAPES								

Inventory List for Vault

The Inventory Lists show all tapes currently in VMS-controlled locations, including LIB and LIBR (library). LIB and LIBR are physically the same place, both locations are combined under the heading for LIBR (TAPE LIBRARY). The sort sequence is by DSN.

WEDNESDAY , AUGUST		DD, YYYY.222		YOUR COMPANY NAME				PAGE	1
				PARM=PARMUPD=TEST					
TMS REPORT-25		INVENTORY LIST FOR VAULT (DRA1) CARTRIDGE							
VOLSER	NEW DATA SET NAME	EDM DATE	VOL EXPIRATION	LBL	CREATE	CREATE	CREATE	PATT	
SLOT		MOVED	SEQ DATE	TYPE	JOBNAME	DATE	TIME	USED	
061000	001 ASM2.ARCH.MASTER	YYYY/222	001 PERMANENT	SL-36	ASM2ARCH	YYYY/175	00:00:00	0002	
061001	002 ASM2.BKUP.MASTER	YYYY/222	001 PERMANENT	SL-36	ASM2BKUP	YYYY/175	00:00:00	0002	
2 TAPES									
WEDNESDAY , AUGUST		DD, YYYY.222		YOUR COMPANY NAME				PAGE	2
				PARM=PARMUPD=TEST					
TMS REPORT-25		INVENTORY LIST FOR VAULT (SAFE) CARTRIDGE							
VOLSER	NEW DATA SET NAME	EDM DATE	VOL EXPIRATION	LBL	CREATE	CREATE	CREATE	PATT	
SLOT		MOVED	SEQ DATE	TYPE	JOBNAME	DATE	TIME	USED	
061008	005 ADMIN.PAYR.DLYBKUP.G0405V00	YYYY/222	001 CATALOG	SL-36	ADMINPDY	YYYY/198	00:00:00	0001	
061007	004 ADMIN.PAYR.DLYBKUP.G0406V00	YYYY/222	001 CATALOG	SL-36	ADMINPDY	YYYY/199	00:00:00	0001	
061004	001 ADMIN.PAYR.DYL.G0406V00	YYYY/222	001 CATALOG	SL-36	ADMINPDY	YYYY/196	00:00:00	0001	
061009	006 ADMIN.PAYR.MLY.G0045V00	YYYY/222	001 CATALOG	SL-36	ADMINMY	YYYY/211	00:00:00	0001	
061006	003 ADMIN.PAYR.PAYR.DLY.G0405V00	YYYY/222	001 CATALOG	SL-36	ADMINPDY	YYYY/198	00:00:00	0001	
061005	002 ADMIN.PAYR.WLY.G0088V00	YYYY/222	001 CATALOG	SL-36	ADMINPWY	YYYY/197	00:00:00	0001	
6 TAPES									
GRAND TOTAL									
8 TAPES									

Vaulted Tapes

To bring a tape back from a vault before VMS has released it, use the following recommended procedure. This provides an accurate audit trail and allows VMS to re-vault the volume after it is opened for input.

Recommended Procedure

Perform Online CHECKIN

Once the volume has been brought back to the library, use the CA 1 ISPF or TIQ commands to perform a CHECKIN operation on the volume. This sets the OUTCODE to (blank)VMS, and clear the OUTDATE and the SLOT number.

Read for Input - Next VMS Execution Re-vaults Volume

The tape should be read for the first time to copy the data set to another tape or disk file. Once the tape is OPENed, CA 1 clears the OUTCODE=(blank)VMS to make the tape eligible for re-vaulting with the next execution of VMS. The Picking and Distribution Lists reflect this re-vaulting. Other volumes in a multivolume data set that were checked in are automatically reset and eligible to be returned to the vault.

Note: The RV parameter in the TMOOPTxx member of *hlq.CTAPOPTN* specifies the minimum retention for a volume brought back from a vault when read for input. The expiration date is the higher value of the volume's current EXPDT or the value specified for the RV parameter plus the current date. This prevents the premature scratching of a volume by TMSCLEAN prior to the volume returning to the vault.

Use Vaulted Tapes Without Checking Them In

If a tape is returned from the vault without informing CA 1, no automatic CHECKIN is performed when the volume is OPENed for input. (A volume that has an OUTCODE but no SLOT number is automatically checked in by CA 1 when it is OPENed for input.) If the tape read is the first volume of a multivolume file, this allows TMSCLEAN to scratch the tapes if the EXPDT is in the past. Responsibility for returning the volume to the vault is manual; the next VMS execution still considers the volume vaulted.

You should avoid using vaulted tapes without checking them in because the tape movement is not recorded on the Picking or Distribution Lists, and the Audit trail of activity on the volume is not accurate.

CA 1 does not allow out-coded tapes to be used for output.

Prevent Tapes From Being Re-vaulted

To prematurely remove a volume from VMS control and not re-vault the volume, issue a single CHECKIN command and do not use the volume for input. The (blank)VMS OUTCODE forces the TMSVMEDT program to ignore the record. If the volume is opened for input, however, VMS reassumes control of the volume.

Off-Site Vault Compression

As vaulting activity at your site evolves, you may eventually have large numbers of empty slots between occupied slots in your vaults. To save space, you can re-sequence the vault starting with SLOT 1.

This technique should *not* be used on volumes controlled by vault retention (VR=n) because the OUTDATE is recalculated and causes volumes to be retained too long.

1. Add a New Vault Name

On the Vault Name control statement (VAULTS=) in the VPD, specify a new vault location ID to use in place of the current ID. The current ID should *not* be removed.

2. Redefine Associated DSN Entries

- For each DSN= control statement that uses the vault to be re-sequenced, specify REDEFINED in the VPD.
- Change the vault name on the V= control statement following the DSN= control statement to the new location ID.

3. Execute VMS Job with PARM=TEST for TMSVMUPD

The VMS job should be executed using PARM=TEST in the TMSVMUPD JCL to suppress updating of the TMC. Review the CA Earl reports to determine if the movement is correct.

4. Execute VMS Job

When you are satisfied with the results of the CA Earl reports, remove PARM=TEST from the TMSVMUPD JCL; the next execution of the VMS job establishes the reconfigured patterns. The old vault name, if not used elsewhere, can be then removed from the pattern.

5. Restore the Original Vault Name

If it is necessary to maintain the original vault name, simply leave REDEFINED on the DSN= statement. Change the V= control statement back to the original vault name, and execute the redefined pattern a second time. The work vault name and the REDEFINED keyword can then be removed from the pattern.

Remove Data Sets from VMS Control

When a data set no longer needs to be vaulted and the volumes currently controlled by VMS are returned to the library, follow these steps to remove the data set from VMS control:

Redefine Associated DSN Entries

- Add the keyword REDEFINED to the DSN= control statement.
- Change the V= control statement associated with the DSN= control statement to reflect V=LIB,EXP. (You may need to use a new V= statement if other DSNs that are associated with it are not to be REDEFINED; this prevents invalid movement.) Another method is to use V=vltx,CR=1; as long as there are no creations of the data set on the same day that VMS runs, all volumes are returned to the library with an OUTCODE of (blank)VMS.

Note: Removal of a DSN= entry from the pattern does not direct VMS to return the tapes to the library immediately. The tapes remain in the VMS location until expired, at which time they are returned to the library.

Execute VMS Job with PARM=TEST for TMSVMUPD

The VMS job should be executed using PARM=TEST on TMSVMUPD JCL to suppress updating of the TMC. Review the CA Earl reports to determine if the movement is correct.

Execute VMS Job

When you are satisfied with the results of the CA Earl reports, remove PARM=TEST from the TMSVMUPD JCL; the next execution of the VMS job establishes the new pattern. The DSN= control statement can then be removed from the VPD.

Alternate Procedure

Another method to remove data sets from VMS control is to do the following:

1. Issue a CHECKIN command for all volumes currently out-coded to the vault.
2. Remove the DSN= control statement prior to the next VMS execution.

This method implies that the volumes have already been returned from the off-site location, as they do not appear on any VMS listings once the CHECKIN has been performed and the DSN= statement is removed.

Chapter 7: Using Online Inquiry and Update Facilities

This section contains the following topics:

[CA1 TMC Record Facilities](#) (see page 95)

[CA 1 ISPF Facilities](#) (see page 95)

[CTS ISPF Facilities](#) (see page 96)

[Security](#) (see page 96)

[CA 1 ISPF Facility](#) (see page 100)

[CTS ISPF Facility](#) (see page 124)

[Volume Pool Record Maintenance](#) (see page 128)

[Tape Inquiry \(TI\) Display](#) (see page 136)

[TIQ Online Inquiry/Update](#) (see page 137)

CA1 TMC Record Facilities

CA 1 provides the following methods to inquire on and update TMC records through online facilities:

- CA 1 ISPF facility
- Tape Inquiry (TI) ISPF display
- TIQ Online Inquiry/Update facility
- CAI Graphical Management Interface (CAIGMI)

CA 1 ISPF Facilities

CA 1 ISPF is the *recommended* method for online inquiry and update of the TMC. These online facilities enable you to do the following:

- Inquire/update TMC volume records
- Inquire/update TMC DSNB records
- Inquire by cataloged data set name
- Inquire on TMC and Audit data set control records
- Specify the date format that is accepted and displayed for all external dates
- Inquire/update the Retention Data Set (RDS)
- Inquire/update the Vault Pattern Description Data Set (VPD)
- Build new TMC records

CTS ISPF Facilities

The Common Tape System (CTS) component provides ISPF update and display capabilities for resources that are used by CA 1 and CA TLMS. The CTS ISPF menu is presented when you select the option **5** on the CA 1 ISPF Primary Option Menu. The following resources are maintained using the CTS ISPF panels:

- Inquire or update volume pools.
- Inquire or update volume pool alerts.
- Inquire or update variables that are used by volume pool monitor and eMail facilities.

More information:

[Volume Pool Monitor \(VPM\)](#) (see page 254)

Security

Security is provided to eliminate unauthorized inquiry or update requests. Inquiry only access is allowed for security controls. The CA 1 system password allows the highest possible access to all online facilities and the use of them. Using this password, you can inquire on and you can update password-protected data sets, and you can perform any online functions. If external security is being used, you must be authorized to use this password.

CA 1 provides the capability to define optional online user profiles. CA 1 also provides capability to control the type and degree of access that can be exercised when using all online facilities. Specific TMC update functions, record types, and fields that can be accessed can be defined.

Notes:

- For more information about CA 1 system security and password protection, see the *Programming Guide*.
- The CTS ISPF panels do not use the CA 1 security profile. Restrict the access to the CTS ISPF panels using your external security system (CA ACF2, CA Top Secret, or IBM RACF).

Fields

Some volume record fields may be rejected for update, depending upon the media type of the volume. The following TMC fields can only be updated for WORM tapes:

- WWID (physical WORM tapes only)
- VWWID (virtual WORM tapes only)
- WMC

The following TMC fields can only be updated for non-WORM tapes:

- EDMID (also valid for virtual WORM tapes)
- DATECLN
- USECLN
- CLNCNT
- TRERRC
- TWERRC
- PRERRC
- PWERRC
- TRERRI (also valid for physical WORM tapes)
- TWERRI (also valid for physical WORM tapes)

Not all TMC fields can be updated. Some of these fields are:

- AUCODE
- AUDATE
- AUFLAG1
- AUTIME
- BATCHID
- CPUID
- HOOKID
- USERID

While the FLAG1 field can be updated by TIQ, the scratch indicator (X'04') cannot be set by an online update. The batch TMSCLEAN program and online scratch utility are the only means of marking tapes scratch. Additionally, not all fields that can be updated by TIQ are updatable on CA 1 ISPF panels. This is to avoid inadvertent updates while tabbing about the update panel. The fields that you can update from the ISPF panel are:

- FLAG1
- FLAG2
- FLAG3
- FLAG4
- FLAG5
- FLAG6
- VOLSER

Certain field updates related to multivolume data sets require that the update be performed to the first volume of the set. These fields are:

- ACCT
- B1INT
- BLKSIZE
- CDATE
- CJOB
- CPGM
- CSTEP
- CTIME
- DEN
- DSN
- EXPDT
- FLAG1
- FLAG2
- FLAG3
- LABEL
- LDATE
- LJOB
- LPGM
- LRECL

- LTIME
- OUTCODE
- OUTDATE
- PASSWRD
- RECFM
- SMSMC
- TRTCH
- LSTDSNB

Any manual update to a record results in the FLAG1 X'20' bit being turned on indicating the update. If the update is for a field considered internal, or critical to CA 1, the X'80' bit is also set in FLAG1. The default fields to which this applies are:

- ACTVOL
- B1INT
- EXPDT
- FLAG3
- FLAG4
- FLAG5
- FLAG6
- LSTDSNB
- NEXTVOL
- NUMDSNB
- PASSWRD
- PREVVOL
- SLOT
- SMSMC
- VOLSEQ
- VOLSER
- 1STVOL

Audit related fields are always automatically updated (AUDATE, AUTIME, AUCODE, BATCHID, CPUID, USERID).

CA 1 ISPF Facility

CA 1 ISPF is the *recommended* method for online inquiry and update of TMC records. This section introduces you to CA 1 ISPF and acquaints you with some of the functions available on the CA 1 ISPF Primary Option Menu. This menu is typically accessed through the ISPF/PDF Primary Option Menu that is unique to each site.

Panel Description

Online inquiry and update tasks for CA 1 ISPF are performed through the use of a series of panels. These panels adhere to ISPF panel format and operational conventions; PF keys operate according to the definitions you specify or the defaults assigned at your site.

All CA 1 ISPF panels are formatted to fit on a 24-line by 80-character display. The first three lines of each display use the format shown below:

Title	Short Message
Command/Option/Selection	Scroll
Long Message	

The Title line shows the title of the panel you are using. The Short Message section reflects various information depending on the functions you request, such as confirming an update request, entering the command to view multivolume information, and so forth. When a short message is displayed, you can use F1 to display the Long Message, which provides more detail on the meaning of the short message. The Scroll value can be set to the value that you typically use.

Request CA 1 ISPF Help and Tutorial Facilities

CA 1 ISPF provides a demand help facility and a full tutorial facility. You can access Help by the following actions:

- Pressing the HELP PF key (default is PF1)
- Entering HELP on the command line and pressing the ENTER key

You can access the tutorial by the following actions:

- Entering T on the command line (if it is presented as a panel option, as shown on the following illustration) and pressing the ENTER key
- Entering T next to the task name and pressing the ENTER key

CA 1 ISPF displays a tutorial providing instructions for the panel or task you are using.

```

----- CA 1 rNN.n PRIMARY OPTION MENU --- CA 1/ISPF  START
OPTION ==> T

0 CA 1 ISPF OPTS - Specify CA 1/ISPF options          USERID - USER001
1 INQ/UPD       - OnLine inquiry/update TMC records   DATE   - MMMDD YYYY
2 UTILITIES     - Perform CA 1 utility functions      TIME   - 11:37
3 MAINTENANCE   - Maintain CA 1 configuration tables  GenLvl - 1004 SP00
4 CHECK LEVEL   - Check level of online modules
5 CTS           - Common Tape System menu
T TUTORIAL      - Display information about CA 1 ISPF
X EXIT          - Terminate CA 1 ISPF using log defaults

      Password      ==>
      Test           ==> NO          (Yes or No)

Enter END command to terminate CA 1 ISPF

(C) Copyright 1990, 2010 by CA, Inc.

```

Note: If CA 1 was initialized to the batch active state, the applicable message appears on the Primary Option Menu: **Note:** CA 1 OSI NOT INSTALLED. TAPE I/O NOT BEING TRACKED PROPERLY **.

Query and Update CA 1 ISPF Session Options

CA 1 ISPF optionally updates your ISPF log file for all activities performed. You can print the log file or use the browse function on the ISPF/PDF Primary Option Menu to display the file.

To view or change the default option for the log file, go to the CA 1 ISPF Primary Option Menu and type 0 (for CA 1 ISPF OPTS) and press ENTER. The CA 1 ISPF Session Options Panel is displayed.

The CA 1 ISPF Session Options Panel gives you an opportunity to do the following:

- View or change the default option (Y) for the log file.
- Specify your preferred date format that is in effect for all of your CA 1 ISPF sessions. The date format can be specified as a unique date pattern (for acceptable values, see the *Utilities and Reports Reference Guide*) or as DEFAULT. If DEFAULT is specified, the DATEFMT parameter in the TMOOPTxx member of *hlq*.CTAPOPTN is used.

In the example below, the preferred date format is MMMDD YYYY.

```
----- CA 1/ISPF SESSION OPTIONS -----  
COMMAND ==>  
  
  ISPF Log Option:  
  
    Update Log Dataset ==> Y      (Yes or No)  
  
  Miscellaneous Session Options:  
  
    DATEFMT ==> MMMDD YYYY      Format of dates during session
```

Query and Update TMC Records

To inquire on or update TMC Volume and DSNB records, or inquire on TMC and Audit Control records, go to the CA 1 ISPF Primary Option Menu and type 1 (for INQ/UPD) and press ENTER. The CA 1 ISPF Inquiry/Update TMC Menu is displayed.

You can use this menu to define criteria for accessing information in the TMC. You can access information by:

- Volume serial number or volume serial number range (Volume range inquiry requires READ access to the TMC)
- DSNB number or DSNB number range
- Data set name (This may be a fully qualified data set name or relative GDG specification if cataloged on the MVS catalog; to request a TMC search, enter a fully qualified data set name or data set name prefix delimited by a dash (-), and set the Cataloged option to NO.) An uncataloged data set name search is a sequential search of the TMC.
- Control record request (inquiry only)

Inquiry by Volume Serial Number

The CA 1 ISPF Inquiry/Update TMC Menu allows you to request the display of specific TMC records. In this example, a single volume serial number 760113 is entered to request specific Volume record information from the TMC.

```

----- CA 1 INQUIRY/UPDATE TMC MENU -----
COMMAND ==>

Inquire/update TMC record by volume serial number
  Volser number ==> 760113 to

Inquire/update TMC record by DSNB number
  DSNB number ==> to

Inquire/update TMC record by Data Set Name
  Data Set Name ==>
  Cataloged ==> YES (Yes or No)

Inquire TMC control record
  Yes/No ==> NO

Build new TMC record
  Yes/No ==> NO

```

The CA 1 ISPF Volume Record Display shown next is generated as a result of the inquiry.

The information retrieved from the TMC for Volume record 760113 appears in the CA 1 ISPF Volume Record Display, with the letter I (for Inquiry) at the Command prompt.

Note: When the audit file is in BLOCKSET mode, all audit date and time fields are in GMT/UTC.

```

----- VOLSER INQUIRY DISPLAY      ENTER 'V' FOR MULTI-VOL
COMMAND ==> I                        DATE: MM/DD/YYYY
                                      05.188
VOLSER = 760113      ACTVOL=          SMSMC= BLANKS
DSN   = SYSP.TEST.TESTNL1.AAAAAA.BBBBBBB.BBCCCC DSN17= AA.BBBBBBB.BBCCCC
EXPDT = CATALOG     ACCT= ..40200000
FLAG1 = 41          FLAG2 = C0        FLAG3 = 02        BATCHID= 00 =
FLAG4 = 08          FLAG5 = 00        FLAG6 = 00        HOOKID = FD = SMF 83
EDMID =             WMC = 00000       WWID =             -
CDATE = MM/DD/YYYY CJOB = USERTSTD   CTIME = 1245       CPGM = CTSTAPER
LDATE = MM/DD/YYYY LJOB = USERTSTD   LTIME = 1245       LPGM = CTSTAPER
CSTEP = TAPE1234    CDDNAME= 0U02SL01 CUNIT = 0E8C       LUNIT = 0E8C
OUTDATE= ZEROS      OUTCODE=          SLOT = 0000000     TRERRC = 00000
BTHDATE= MM/DD/YYYY VENDOR = BLANKS   COUNT = 00001     TWERRC = 00000
DATECLN= ZEROS      USECLN = 00000     CLNCNT = 000      TRERRI = 00000
VOLSEQ = 0001        ROBTY =          ROBID = 000       TWERRI = 00000
1STVOL = 760113     NEXTVOL= 760114    PREVVOL=         PRERRC = 00000
NUMDSNB= 0000       1STDSNB= 0000000   LSTDSNB= 0000000  PWERRC = 00000
LABEL = SL          DEN = 38K          TRTCH = 18TK     PRERRI = 00000
RECFM = F           LRECL = 000024     BLKSIZE= 000024   PWERRI = 00000
AUDATE = MM/DD/YYYY AUTIME = 1245     BLKCNT = 000000002
AUCODE = 02         AUFLAG1= 00        CPUID = XE76     USERID = +CATALOG
VOLPERC= 002        FILPERC= 002       COMPRES= 000     CTLGCNT= 001

```

After reviewing the information contained in the keyword fields, you may decide to update one or more of the fields. With the cursor positioned at the command prompt, type U (for Update) and press ENTER. Make any necessary changes to the fields directly on the screen, and press ENTER to submit the update. You must press ENTER a second time to confirm the update before the request is processed.

If a pointer field needs to be updated, enter command P (for Pointers) in the command line of the update panel and perform the update as described above. Pointer fields of the volume record are:

- VOLSEQ
- 1STVOL
- NEXTVOL
- PREVVOL
- NUMDSNB
- 1STDSNB
- LSTDSNB

With the cursor positioned at the Command prompt, type U (for Update) and press ENTER. Make any necessary changes to the fields directly on the screen, and press ENTER to submit the update. You must press ENTER a second time to confirm the update before the request is processed.

Request a Multivolume Listing

If the volume displayed on the CA 1 ISPF Volume Record Display is part of a multivolume data set, you can request a Multiple Volume Listing by entering V at the Command prompt, as shown below:

```

----- VOLSER INQUIRY DISPLAY      ENTER 'V' FOR MULTI-VOL
COMMAND ==> V                      DATE: MM/DD/YYYY
                                   05.188
VOLSER = 760113    ACTVOL=          SSMC= BLANKS
DSN   = SYSP.TEST.TESTNL1.AAAAAA.BBBBBBB.BBCCCC DSN17= AA.BBBBBBB.BBCCCC
EXPDT = CATALOG   ACCT= ..40200000
FLAG1 = 41        FLAG2 = C0        FLAG3 = 02    BATCHID= 00 =
FLAG4 = 08        FLAG5 = 00        FLAG6 = 00    HOOKID = FD = SMF 83
EDMID =           WMC   = 00000    WWID   =       -
CDATE = MM/DD/YYYY CJOB = USERTSTD CTIME = 1245   CPGM  = CTSTAPER
LDATE = MM/DD/YYYY LJOB = USERTSTD LTIME = 1245   LPGM  = CTSTAPER
CSTEP = TAPE1234   CDDNAME= 0U02SL01 CUNIT = 0E8C  LUNIT = 0E8C
OUTDATE= ZEROS     OUTCODE=         SLOT  = 0000000 TRERRC = 00000
BTHDATE= MM/DD/YYYY VENDOR = BLANKS COUNT  = 00001 TWERRC = 00000
DATECLN= ZEROS     USECLN = 00000   CLNCNT = 000   TRERRI = 00000
VOLSEQ  = 0001     ROBTY  =         ROBID  = 000   TWERRI = 00000
1STVOL  = 760113   NEXTVOL= 760114  PREVVOL=       PRERRC = 00000
NUMDSNB= 0000     1STDSNB= 0000000 LSTDSNB= 0000000 PWERRC = 00000
LABEL   = SL       DEN    = 38K     TRTCH  = 18TK  PRERRI = 00000
RECFM   = F        LRECL  = 000024  BLKSIZE= 000024 PWERRI = 00000
AUDATE  = MM/DD/YYYY AUTIME = 1245  BLKCNT  = 0000000002
AUCODE  = 02       AUFLAG1= 00      CPUID   = XE76  USERID  = +CATALOG
VOLPERC = 002      FILPERC= 002     COMPRES= 000   CTLGCNT= 001

```

The Multiple Volume Listing is displayed:

```

----- MULTIPLE VOLUME LISTING ----- Row 1 to 2 of 2
COMMAND ==>                          SCROLL ==> HALF
                                   DATE: MM/DD/YYYY
                                   05.188
      I = Inquiry      U = Update
DATA SET NAME ==> SYSP.TEST.TESTNL1.AAAAAA.BBBBBBB.BBCCCC
VOLUME  VOLUME
CMD  SEQ  SERIAL  VAULT  SLOT  TYPE
0001  760113      0000000  SL
0002  760114      0000000  SL

```

You can inquire on or update any volume that appears in the Multiple Volume Listing by typing I or U in front of the appropriate volume, and pressing ENTER. A full display of that volume appears.

Request a Multiple File Listing

If the volume displayed on the CA 1 ISPF Volume Record Display contains secondary data sets, you can request a Multiple File Listing by entering F at the command prompt, as shown below:

```

----- VOLSER INQUIRY DISPLAY ----- ENTER 'V' OR 'F'
COMMAND ==> F                                DATE: MM/DD/YYYY
                                           05.188
VOLSER = 760116      ACTVOL=                      SMSMC= BLANKS
DSN   = SYSP.TEST.TESTNL1.AAAAAA.BBBBBBB.BBCCCC DSN17= AA.BBBBBBB.BBCCCC
EXPDT = CATALOG      ACCT= ..40200000
FLAG1 = 41           FLAG2 = 40           FLAG3 = 02           BATCHID= 00 =
FLAG4 = 00           FLAG5 = 00           FLAG6 = 00           HOOKID = 44 = CLOSE IO
EDMID =              WMC = 00000          WWID =              -
CDATE = MM/DD/YYYY  CJOB = USERTSTD       CTIME = 1246         CPGM = CTSTAPER
LDATE = MM/DD/YYYY  LJOB = USERTSTD       LTIME = 1246         LPM = CTSTAPER
CSTEP = TAPE1234     CDDNAME= 0U02SL01    CUNIT = 0E8C         LUNIT = 0E8C
OUTDATE= ZEROS       OUTCODE=              SLOT = 00000000      TRERRC = 00000
BTHDATE= MM/DD/YYYY  VENDOR = BLANKS      COUNT = 00002         TWERRC = 00000
DATECLN= ZEROS       USECLN = 00000        CLNCNT = 000         TRERRI = 00000
VOLSEQ = 0002        ROBTY =              ROBID = 000         TWERRI = 00000
1STVOL = 760115      NEXTVOL=              PREVVOL= 760115     PRERRC = 00000
NUMDSNB= 0001        1STDSNB= 0000199     LSTDSNB= 0000199      PWERRC = 00000
LABEL = SL           DEN = 38K            TRTCH = 18TK         PRERRI = 00000
RECFM = F            LRECL = 000024        BLKSIZE= 000024       PWERRI = 00000
AUDATE = MM/DD/YYYY  AUTIME = 1246        BLKCNT = 0000000002
AUCODE = 02          AUFLAG1= 00          CPUID = XE76         USERID = SYSPROG
VOLPERC= 002         FILPERC= 002         COMPRES= 000         CTLGCNT= 000

```

The Multiple File Listing is displayed:

```

----- MULTIPLE FILE LISTING ----- Row 1 to 1 of 1
COMMAND ==>                                SCROLL ==> HALF
                                     I = Inquiry      U = Update
                                     DATE: MM/DD/YYYY
                                           05.188
FILE  VOLUME
CMD  SEQ  SERIAL DSNB   DATA SET NAME
0002  760116 0000199 SYSP.TEST.TESTNL2

```

You can inquire on or update any DSNB that appears in the Multiple File Listing by typing I or U in front of the appropriate DSNB, and pressing ENTER. A full display of that DSNB appears.

Inquiry by DSNB Number

In the panel below, DSNB number 119 is entered to request information for that file.

```
----- CA 1 INQUIRY/UPDATE TMC MENU -----  
COMMAND ==>  
  
  Inquire/update TMC record by volume serial number  
    Volser number ==>          to  
  
  Inquire/update TMC record by DSNB number  
    DSNB number  ==> 119      to  
  
  Inquire/update TMC record by Data Set Name  
    Data Set Name ==>  
    Cataloged    ==> YES      (Yes or No)  
  
  Inquire TMC control record  
    Yes/No      ==> NO  
  
  Build new TMC record  
    Yes/No      ==> NO
```

The panel below is generated as a result of the request. The information retrieved from the TMC for DSNB number 119 appears in the CA 1 ISPF DSNB Record Display, with the letter I (for Inquiry) at the Command prompt. If a file exists on more than one volume, the DSNB Volumes Listing is available for display.

```

----- DSNB INQUIRY DISPLAY -----
COMMAND ==> I                                DATE: MM/DD/YYYY
                                           05.188
DSNB   = 0000119
DSN    = SYSP.TEST.SL.TAPE.FILE8

EXPDT  = CATALOG    ACTIND = 80    BID  = FF    BATCHID= 00 =
FLAG1  = 01 = (RDS)          HOOKID = 14 = OP SL OT

VOLSER = 760104    F1STVSN= 760104    FILSEQ = 00008    SMSMC  = BLANKS
CDATE  = MM/DD/YYYY CJOB   = USERTST4    CTIME  = 1305    CPGM   = CTSTAPER
CSTEP  = TAPSLW    CTLGCNT= 000        FILPERC= 000    COMPRES= 000
CURDSNB= 0000119    PREV   = 0000118    NEXT    = 0000120
RECFM  = F         LRECL  = 000024    BLKSIZE= 000024    BLKCNT = 0000002

AUDATE  = MM/DD/YYYY AUTIME = 1305        USERID = SYSPROG
AUCODE  = 02         AUFLAG1= 00         CPUID  = XE76

DSNB IS LOCATED ON THE FOLLOWING VOLUMES:
VOLSER   760104
SLOT     0000000
VAULT    ==>

```

To update any DSNB field, enter U on the command line and press ENTER to display the update panel. If a pointer field needs to be changed, enter P on the command line of the update panel and press ENTER. Then, make the required changes.

The DSNB record has the following pointer fields:

- VOLSER
- F1STVSN
- FILSEQ
- CURDSNB
- PREV
- NEXT

Inquiry on Control Records

In the following example, the TMC Control record option is set to YES to request a Control record display for TMC and Audit Control records:

- TMSCTL#1 - resides on the TMC
- TMSCTL#2 - resides on the TMC
- TMSCTL#3 - extension of TMSCTL#1 and resides on the TMC
- TMSCTL#4 - resides on the Audit data set

```
----- CA 1 INQUIRY/UPDATE TMC MENU -----  
COMMAND ==>  
  
  Inquire/update TMC record by volume serial number  
    Volser number ==>          to  
  
  Inquire/update TMC record by DSNB number  
    DSNB number   ==>          to  
  
  Inquire/update TMC record by Data Set Name  
    Data Set Name ==>  
    Cataloged    ==> YES      (Yes or No)  
  
  Inquire TMC control record  
    Yes/No       ==> YES  
  
  Build new TMC record  
    Yes/No       ==> NO
```

The CA 1 ISPF Control Record Display shown on the following page is generated as a result of the request.

Information for all Control records is available for inquiry (but not update) in this display:

Pre-Enhanced TMC Display

----- CONTROL RECORD DISPLAY (PAGE 1 OF 3) -----		
COMMAND ==>		
FLAG1 ==> 00	IPL DATE ==> MMDD YYYY	DATE: MMDD YYYY
RANGES ==> 002	IPL TIME ==> 115329	96.099
DSNB ALLOC => 000020	DSNB USED ==> 000004	
DSNB BASE ==> 000050	NEXT DSNB ==> 000015	
NUM AUDIT ==> 005016	1ST AUDIT ==> 002623	NEXT AUDIT => 003193
NEXT LBL ==> 001060	BACIND ==> 00	NEXT SEND ==> 000000
BKUP VOL ==> MVXE90	BKUP DATE ==> MMDD YYYY	BKUP TIME ==> 01357
BKUP DSN ==> DATA.CENTER.CA1.BKUP.TAPE1		
LOW1 ==> 980000	HIGH1 ==> 980040	ADJ1 ==> -979997
LOW2 ==> 980550	HIGH2 ==> 980555	ADJ2 ==> -980506
LOW3 ==> 000000	HIGH3 ==> 000000	ADJ3 ==> 0000000
LOW4 ==> 000000	HIGH4 ==> 000000	ADJ4 ==> 0000000
LOW5 ==> 000000	HIGH5 ==> 000000	ADJ5 ==> 0000000
LOW6 ==> 000000	HIGH6 ==> 000000	ADJ6 ==> 0000000
LOW7 ==> 000000	HIGH7 ==> 000000	ADJ7 ==> 0000000
LOW8 ==> 000000	HIGH8 ==> 000000	ADJ8 ==> 0000000
LOW9 ==> 000000	HIGH9 ==> 000000	ADJ9 ==> 0000000
LOW10 ==> 000000	HIGH10 ==> 000000	ADJ10 ==> 0000000

Enhanced TMC Display

```

----- CONTROL RECORD AND VOLSER RANGE DISPLAY
COMMAND ==>
Row 1 to 7 of 7
SCROLL ==> HALF

FLAG1 ==> 00      IPL DATE ==> YYYYMM,DD      DATE: YYYYMM,DD
RANGES ==> 000    IPL TIME ==> 092417          05.188

DSNB ALLOC => 0000500  DSNB USED ==> 0000002
DSNB BASE ==> 0012405  NEXT DSNB ==> 0000472      1ST VOLDEF => 00012404

NUM AUDIT ==> 00010512 1ST AUDIT ==> 00006351      NEXT AUDIT => 006590
NEXT LBL ==> 000001    BACIND ==> 00          NEXT SEND ==> 000000

BKUP VOL ==> BLANKS    BKUP DATE ==> 2005JUL,05    BKUP TIME ==> 00637
BKUP DSN ==> NULLFILE

AUDIT FLAG => 00      CATVSN ==> 000000

VOLSER RANGES      TMC RELATIVE RECORD NUMBER

RANGE  1:      001500 - 001599      00000003 - 00000102
RANGE  2:      100001 - 100099      00000103 - 00000201
RANGE  3:      AIT000 - AIT100      00000202 - 00000302
RANGE  4:      CA0000 - CA9999      00000303 - 00010302
RANGE  5:      DL2000 - DL2099      00010303 - 00010402
RANGE  6:      I00001 - I01000      00010403 - 00011402
RANGE  7:      I50000 - I51000      00011403 - 00012403

```

Note: For more information about TMC and Audit Control records, see the *Programming Guide*.

Build New TMC Records

Note: The security profile (password) entered on the CA 1 PRIMARY OPTION MENU must provide update authority for all TMC volume and DSNB fields.

In the following example, TMC information is built for a multivolume, multi-data set volume set. The volumes defined are AIVS volumes. The first file (file sequence one) is contained on three volumes (AAAAAA,BBBBBB, and CCCCCC). File two is contained on the third volume (CCCCCC). File three starts on the third volume and is continued on the fourth and fifth volumes (DDDDDD and EEEEEEE). File four is contained on the fifth volume. All of the files are cataloged.

The Build new TMC record option is set to YES.

```
----- CA 1 INQUIRY/UPDATE TMC MENU -----  
COMMAND ==>  
  
  Inquire/update TMC record by volume serial number  
    Volser number ==>                to  
  
  Inquire/update TMC record by DSNB number  
    DSNB number   ==>                to  
  
  Inquire/update TMC record by Data Set Name  
    Data Set Name ==>  
    Cataloged    ==> YES          (Yes or No)  
  
  Inquire TMC control record  
    Yes/No       ==> NO  
  
  Build new TMC record  
    Yes/No       ==> YES
```


The CA 1 ISPF Build New TMC Record display shown on the following page is generated as a result of the request.

```

----- BUILD NEW TMC RECORD -----
COMMAND ==>                                     DATE: MMDD YYYY
                                                99.340

General      Media      Location
Base volume  =          TRTCH  =          Vlt/Area=
Label type   =          Density =          Slot   =
                                      Outdate =

AIVS         = NO          Foreign = NO    (implies AIVS)
Actual volume =          (For AIVS (required) or foreign (optional))
User acctg   =

First File Information
File name    =
Expiration date =
Record format =
Record length =          Catalog file = NO
Block size   =          UCBTYP =          Optional Override
Creation jobname=

Enter V for multiple volumes or enter F for multiple files.
Enter SAVE to accept and prepare to enter a new volume set.
Press END to return to previous level.
Enter CANCEL to ignore changes and return to previous level.

```

For individual field definitions and restrictions, press PF HELP while this screen is shown.

Data for the base volume is filled in. Enter a V on the command line to indicate that additional volume(s) for the first file are needed.

```

----- BUILD NEW TMC RECORD -----
COMMAND ==>                                     DATE: MMDD YYYY
                                                99.340

General      Media      Location
Base volume  &coA. = 001812    TRTCH  = 18TK    Vlt/Area=
Label type    = SL          Density = 38K        Slot   =
                                      Outdate =

AIVS         &coB. = YES          Foreign = NO    (implies AIVS)
Actual volume = AAAAAA (For AIVS (required) or foreign (optional))
User acctg    = USER.ACCTG

First File Information
File name     = YURID01.TEST.FILE1
Expiration date = MMDD YYYY
Record format  = FB
Record length  = 80          Catalog file = YES
Block size     = 3120        UCBTYP =          Optional Override
Creation jobname= CJB0

Enter V for multiple volumes or enter F for multiple files.
Enter SAVE to accept and prepare to enter a new volume set.
Press END to return to previous level.
Enter CANCEL to ignore changes and return to previous level.

```

Note: Two fields, Base Volume A and Actual Volume B, have special rules based on the values given to AIVS and Foreign.

If AIVS and Foreign are both set to NO (not an AIVS volume), then the Base Volume is assumed to be both the internal and external volume serial number for the tape being added. The requirement for Base Volume is that it must be a valid volser in the TMC and must be in delete status. The Actual Volume field is not used.

If AVIS=YES and Foreign=NO, then the Base Volume is the external volser that is assigned to the new volume being added. This volser should be in delete status. If the specified volume is not in delete status, a sequential search of the TMC begins, starting at the specified volser, until a record in delete status is found. Only volumes in delete status may be built or added to the TMC through this facility.

Actual Volume is the actual internal volser found on the VOL1 record of the tape being added. This volser will not be used by the OS catalog and should not be used in JCL. It will only be used by CA 1 at OPEN/Input to ensure the correct tape is mounted.

If AVIS and Foreign are both set to YES, then the same rules apply as above except that in addition to being in delete status, the Base Volume must also have the TMNRS indicator turned on in FLAG4. If not, the TMC is searched sequentially looking for a volume record in delete status that also has the TMNRS indicator turned on.

Enter the first additional volume on the line provided. If more volumes are needed, enter an A in the command column to provide space for adding the next volume.

For AIVS volumes, the volume serial entered here is the actual volume serial of the tape, and may not be the same as the CA 1 assigned external volser. Volumes are assigned by CA 1 from those in delete status starting at or following the base volume given.

For foreign volumes (Foreign=YES), volumes are assigned by CA 1 from those in delete status starting at or near the base volume given. The nonresident indicator (TMNRS bit of TMFLAG4) must also be on.

For volumes that are non-AIVS and non-foreign, the exact volumes must be present in the TMC and in delete status. No searching for substitutes is done.

```

----- ENTER TMC ADDITIONAL VOLUMES ----- ROW 1 TO 2 OF 2
COMMAND ==>                                SCROLL ==> CSR
                                           DATE: DDMMYY YYYY
                                           96.053

File name      = USER001.TEST.FILE1
File sequence  = 0001
First volume   = 001812

A = Add a new volume      D = Delete a volume

Cmd      Volume
        Serial
-        BBBBBB
-        CCCCCC
***** BOTTOM OF DATA *****

```

Enter all additional volumes for the first file before continuing to define additional files.

For individual field definitions and restrictions, press PF HELP while this screen is shown.

When finished entering additional volumes, press PF END to return to the previous screen.

Enter an F in the command line to indicate that additional files are to be defined.

```

----- BUILD NEW TMC RECORD -----
COMMAND ==> F                                DATE: DDMMYY YYYY
                                           96.053

General                               Media                               Location
Base volume      = 001812             TRTCH   = 18TK             Vlt/Area=
Label type       = SL                 Density = 38K             Slot   =
                                           Outdate =

AIVS             = YES                Foreign = NO             (implies AIVS)
Actual volume    = AAAAAA (For AIVS (required) or foreign (optional))
User acctg       = USER.ACCTG

First File Information
File name        = USER001.TEST.FILE1
Expiration date  = DDMMYYYY
Record format    = FB
Record length    = 80                 Catalog file = YES
Block size       = 3120
Creation jobname = CJOB

Enter V for multiple volumes or enter F for multiple files.
Enter SAVE to accept and prepare to enter a new volume set.
Press END to return to previous level.
Enter CANCEL to ignore changes.

```

The screen shown below is used to enter information for additional file(s).

Space is provided to enter information for the first additional file. To create space for another file, enter A in the command column.

The file sequence number and first volume (volume where this file starts) are provided and may not be modified.

For individual field definitions and restrictions, press PF HELP while this screen is shown.

```
----- ENTER TMC ADDITIONAL FILES ----- ROW 1 TO 1 OF 1
COMMAND ==>                                SCROLL ==> CSR
                                           DATE: DDMMYYYY
                                           96.053
Base volume      = 001812

Cmd  (A = Add a new file   D = Delete a file   V = additional volume(s))

- File name      =
  File sequence  = 0002      First volume     = CCCCCC
  Record format  =          Record length    =
  Block size     =          Creation jobname =
  Expiration date =          Catalog file     = NO
-----
***** BOTTOM OF DATA *****
```

After entering information for files two and three, type V in the command column to specify additional volume(s) for file three.

```

----- ENTER TMC ADDITIONAL FILES ----- ROW 1 TO 2 OF 2
COMMAND ==>                                SCROLL ==> CSR
                                           DATE: DDMMYYYY
                                           96.053

Base volume      = 001812

Cmd  (A = Add a new file   D = Delete a file   V = additional volume(s))

- File name      = USER001.TEST.FILE2
  File sequence  = 0002           First volume   = CCCCCC
  Record format  = FB             Record length  = 82
  Block size     = 820            Creation jobname = CJF2
  Expiration date = DDMMYYYY      Catalog file   = YES
-----
V File name      = USER001.TEST.FILE3
  File sequence  = 0003           First volume   = CCCCCC
  Record format  = FB             Record length  = 83
  Block size     = 830            Creation jobname = CJF3
  Expiration date = DDMMYYYY      Catalog file   = YES
-----

```

Enter the volume serial number for the second and third volumes containing file three.

Press PF END when finished entering additional volumes.

```

----- ENTER TMC ADDITIONAL VOLUMES ----- ROW 1 TO 2 OF 2
COMMAND ==>                                SCROLL ==> CSR
                                           DATE: DDMMYYYY
                                           96.053

File name      = USER001.TEST.FILE3
File sequence   = 0003
First volume    = CCCCCC

A = Add a new volume   D = Delete a volume

Cmd      Volume
        Serial

-      DDDDDD
-      EEEEE
***** BOTTOM OF DATA *****

```

To create space for another file, enter A in the command column.

```

----- ENTER TMC ADDITIONAL FILES ----- ROW 2 TO 2 OF 2
COMMAND ==>
Base volume      = 001812
DATE: DDMMYYYY
96.053

Cmd  (A = Add a new file   D = Delete a file   V = additional volume(s))

A  File name      = USER001.TEST.FILE3
   File sequence  = 0003      First volume    = CCCCCC
   Record format  = FB        Record length   = 83
   Block size     = 830       Creation jobname = CJF3
   Expiration date = DDMMYYYY  Catalog file    = YES
-----
***** BOTTOM OF DATA *****

```

The first volume for file four is automatically filled in with the last volume of the previous file (EEEEEE).

Press PF END when finished entering additional file information.

```

----- ENTER TMC ADDITIONAL FILES ----- ROW 2 TO 3 OF 3
COMMAND ==>
Base volume      = 001812
DATE: DDMMYYYY
96.053

Cmd  (A = Add a new file   D = Delete a file   V = additional volume(s))

-  File name      = USER001.TEST.FILE3
   File sequence  = 0003      First volume    = CCCCCC
   Record format  = FB        Record length   = 83
   Block size     = 830       Creation jobname = CJF3
   Expiration date = DDMMYYYY  Catalog file    = YES
-----
-  File name      = USER001.TEST.FILE4
   File sequence  = 0004      First volume    = EEEEEEE
   Record format  = FB        Record length   = 84
   Block size     = 840       Creation jobname = CJF4
   Expiration date = DDMMYYYY  Catalog file    = YES
-----

```

Enter SAVE on the command line to indicate that the definitions for this volume set are done.

```

----- BUILD NEW TMC RECORD -----
COMMAND ==> SAVE                                     DATE: DDMMYYYY
                                                    96.053

General      Media      Location
Base volume  = 001812    TRTCH  = 18TK    Vlt/Area=
Label type   = SL       Density = 38K     Slot   =
                                           Outdate =

AIVS         = YES      Foreign = NO    (implies AIVS)
Actual volume = AAAAAA (For AIVS (required) or foreign (optional))
User acctg   = USER.ACCTG

First File Information
File name     = USER001.TEST.FILE1
Expiration date = DDMMYYYY
Record format  = FB
Record length  = 80      Catalog file = YES
Block size     = 3120
Creation jobname = CJOB

Enter V for multiple volumes or enter F for multiple files.
Enter SAVE to accept and prepare to enter a new volume set.
Press END to return to previous level.
Enter CANCEL to ignore changes.

```

If AIVS was set to YES on the Build New TMC Record screen, the following confirmation screen is displayed. It shows the actual and corresponding external volume serial numbers. Press ENTER to perform the update.

```

----- CONFIRM AIVS ----- Using 001837
COMMAND ==>                               SCROLL ==> CSR
B017 Volume 001837 assigned in place of 001812
                                                    96.053

Press ENTER to CONFIRM update, END or CANCEL to return to previous panel

      Actual Volume   : External Volume

      AAAAAA         : 001837
     BBBBBB         : 001838
      CCCCCC         : 001839
      DDDDDD         : 001840
      EEEEE         : 001841
***** BOTTOM OF DATA *****

```

If AIVS was set to NO on the Build New TMC Record screen, the TMC update takes place without a confirmation screen.

The Build New TMC Record screen is displayed again with most of the fields cleared. If another volume set is to be defined, you can do it now. Otherwise, press PF END to return to the previous menu.

Update the TMC Using the CA 1 ISPF Utility Functions

To access the common librarian command functions, go to the CA 1 ISPF Primary Option Menu, type 2 (for UTILITIES) and press ENTER.

The following functions are available on the CA 1 ISPF Utilities Menu:

ADD

Activates an inactive (deleted) volume serial number to make it available for use. The volume serial number must be in the TMC and in inactive (delete) status.

CHECKIN

Checks in a tape that was previously out-of-area. If the Multivolume Process field on the utilities menu contains a Y, and the volume specified is a multivolume, all volumes are checked in.

CLEAN

Updates cleaning statistics to indicate a tape volume has been cleaned. This function tells CA 1 that a tape has been cleaned before it was marked as needing cleaning on the Scratch and Clean List.

Note: This utility cannot be used for the 3592 WORM tapes.

DELETE

Marks a tape volume as inactive (deleted) when the volume is in scratch status, making the volume unavailable for use and for most reporting.

EXPIRE

Changes the expiration date of a volume to the current system date.

EXTEND

Extends the expiration date of a volume for a specified number of days from the current expiration date.

LABEL

Provides the capability to create an external label request for a volume serial number or a cataloged data set. The external label request being generated is processed by the Online Label Interface. For more information, see Common Tape System. If the Multivolume Process field on the utilities menu contains a Y and the volume specified is part of a multivolume chain, an external label request is generated for every volume in the chain.

RETAIN

Sets the expiration date of a volume to a specified number of days from the creation date.

CHECKOUT

Checks a tape volume out-of-area. The SLOT value should only be updated for non-multivolume data sets. If the Multivolume Process field on the utilities menu contains a Y, and the volume specified is part of a multivolume chain, all volumes on the chain are checked out.

ERASE

Indicates volume has gone through physical degauss. The TMDEGAU setting for TMFLAG4 of the CA 1 TMC volume record is turned on.

SCRATCH

Scratches a volume set immediately. Enter the volume serial number for the base volume of the volume set. The volume set is checked, to make sure it is eligible for being scratched. The utility cannot be used to scratch 3592 WORM tapes. If eligible, a confirmation screen is shown before the scratch is actually performed. If not eligible, a brief message is displayed. For more detailed information, press the PF HELP key.

```

----- CA 1 UTILITIES MENU -----
OPTION ==>

 1 ADD      - Activate an inactive (deleted) volume record
 2 CHECKIN  - Perform the checkin for out-of-area tape
 3 CLEAN    - Update cleaning statistics
 4 DELETE   - Deactivate a scratch volume record
 5 EXPIRE   - Set expiration date to current date
 6 EXTEND   - Extend expiration date of data set from EXPDT
 7 LABEL    - Print external tape label
 8 RETAIN   - Reset expiration date from creation date (CDATE)
 9 CHECKOUT - Check a volume out-of-area
10 ERASE    - Indicate volume has gone through physical degauss
11 SCRATCH  - Scratch a volume immediately

Required Fields :
  VOL ==>                               Multivolume Process ==> (Opt: 2,7,9)
or  DSNB ==>                           (Expire, Extend, Retain)
    DAYS ==>                           (Number of days to Extend/Retain)
    OUTCODE ==>                         (Checkout)
    OUTDATE ==>                         (Checkout - defaults to current date)
    SLOT  ==>                           (Checkout - defaults to zero)

```

Caution

If the ISPF split screen feature is used, one or more of the bottom lines from this panel may not be visible and available for use. PF key assignments, visible by PFSHOW ON, may also cause a problem.

Maintain System Options and Display the Security Table

To inquire on or update the configuration tables, go to the CA 1 ISPF Primary Option Menu, type 3 (for MAINTENANCE) and press ENTER. The CA 1 ISPF Configuration Tables Maintenance Panel is displayed.

This panel provides access to the following functions:

- Display the access passwords defined to the online system and the access criteria allowed for each.
- Display or update system options, such as default retention, mixed expiration date and console time out, list the resident (active) rules, or create a new system options member.
- Display or update scratch tape pool information, including scratch tape pool names, assigned tape ranges, and the access criteria for each, list the resident (active) rules, or create a new scratch tape pool definition member.
- Display or update the set of nonspecific mount rules, list the resident (active) rules, or create a new nonspecific mount rules member.
- Display, update, delete or create new DSN and MGMTCLAS control statements in the Retention Data Set (RDS).
- Display, update, delete or create new Vaults and Vault Management System control statements in the Vault Pattern Description Data Set (VPD).

Note: There must be an existing Vault Pattern data set available to use this facility. If not, create one with at least one VAULTS= statement and at least one pattern before using the CA 1 VPD Rules Edit facility.

```
----- CA 1 CONFIGURATION TABLES MAINTENANCE -----
OPTION ==>

      1  - CA 1 SECURITY Table
      2  - CA 1 SYSTEM Options
      3  - CA 1 SCRATCH Pool Definitions
      4  - CA 1 NSM (Non-Specific Mount) Rules
      5  - CA 1 RDS Rules Edit
      6  - CA 1 VPD Rules Edit

Required fields if requesting OPTION 2, 3, or 4:

Do you wish to use the resident module ? ==> YES   (Yes or No)

The following information must be supplied if NO was entered above:

Parmlib DSN      ==>
Member suffix    ==>      (2 digits)
Allocate new     ==> NO   (Yes or No)
```

Enter option 5 to display the Retention Data Set Rules panel, as shown below.

```
-----Retention Data Set Rules-----
Command ==> _____ DATE: MM/DD/YY
                                04.007
RDS Dataset Name = _____

Confirm a rule delete:  YES

Confirm save of rules:  YES

The name of the fully qualified data set that
contains the RDS rules being edited will need to be provided.
```

Enter 6 to display the Vault Pattern Rules panel, as shown below:

```
----- Vault Pattern Rules -----
Command ==> _____ DATE: MM/DD/YY
                                04.007
1 Maintain Vault Definitions (VAULT=)
2 List Vault Patterns (DSN=, VOL= or CDSN=)

Vault Patterns   : _____

Vault Definitions : _____

Confirm a rule delete:  YES

Confirm save of rules:  YES
```

Each panel provides help text to guide you in using the panel. Enter HELP on the command line or press the PF1 key to obtain help on using the panels.

CTS ISPF Facility

The Common Tape System ISPF menu is presented when you enter **5** on the CA 1 ISPF Primary Option Menu.

```
----- CTS Primary Menu -----  
OPTION ==>  
  
      1 - Variable Maintenance  
      2 - Volume Pool Maintenance  
      3 - Volume Pool Alert Maintenance  
      T - Tutorial  
      Copyright (c) 2012 CA Inc., All rights reserved.
```

Notes:

- The CTS ISPF panels do not use the CA 1 security profile. Restrict access to the CTS ISPF panels using your external security system (CA ACF2, CA Top Secret, or IBM RACF).
- Volume pools, variables, and alerts are saved in the CA 1 VSAM Database (VDB). A control record is used in the VDB to monitor changes to any of these elements.

More information:

[Volume Pool Monitor \(VPM\)](#) (see page 254)

[Using Volume Pool Monitor Alert and Email Facilities](#) (see page 263)

Variable Record Maintenance

The Volume Pool Monitor and email features of CA 1 use variables that are saved in the CA 1 VSAM Database (VDB). Predefined variables are shipped with the product and dynamically updated in the Volume Pool Monitor (VPM) subtask of CTS and selected CA 1 utilities. You can also create variables for extensive customization of emails and WTOs. The variable categories are:

User

Identify variables that the user creates through these ISPF panels. The user creates and updates the variable name and associated data (a character string). These variables can begin with any mixed case alphanumeric character or the underscore (" _") character. Blanks and commas are not allowed in the variable name, however both are allowed in the data portion that is associated with each variable.

Options

Identify variables that configure the VPM subtask or email facilities. These variables begin with OPT.

Product

Identify CA 1 and CTS variables. Selected CA 1 components in your site automatically update these variables. These variables begin with CTS and TMS.

Systems

Identify system variables. These variables are automatically updated to reflect attributes of the system that CA 1 or CA TLMS is running on. These variables begin with SYS.

You can use all of these variable types in the text of emails and WTOs that are issued by the Volume Pool Monitor subtask of CTS or the CTSMail program.

Do not create or delete any variables other than USER variables. You can update OPT variables, but you cannot delete them. CA 1 and CTS components update all other variables.

To create or update a variable, enter **1** on the CTS Primary Menu. The Variable Type Menu appears:

```

----- CTS Variable Type Menu -----
COMMAND ==>
      1  - User Variables
      2  - Options Variables
      3  - Product Variables
      4  - Systems Variables
      5  - All Variables

```

After selecting any of the options on the Variable Type Menu, the CTS Variable List appears including only variables of the type that you selected. Variables appear in ascending alphanumeric sequence by variable name.

```
----- CTS Option Variable List ----- Row 1 of 10
COMMAND ==>                                SCROLL ==> PAGE
Commands: END, CANCEL, UP, or DOWN          Date: yy/10/16
Line Commands: U or S                       Time: 11:05
CMD Variable Name                          Value
-----
  OPT_EMAIL_DEFAULT_DSN                     CAI.SEND
  OPT_EMAIL_DEFAULT_FORMAT                   TEXT
  OPT_EMAIL_DEFAULT_FROM                     ron.davis@abc.com
  OPT_EMAIL_DEFAULT_SUBCHAR                  &
  OPT_EMAIL_DEFAULT_TITLE                    EMAIL FROM CA TAPE MANAGEMENT
  OPT_EMAIL_DEFAULT_TO                       bob.smith@abc.com
  OPT_VPM_DISPLAY_WTO_MAX                    20
  OPT_VPM_RETAIN_HISTORY_DAILY               90
  OPT_VPM_RETAIN_HISTORY_WEEKLY              365
  OPT_VPM_TIMER                              00:30:00
```

To open a detailed view, enter **S** on the line command field (CMD) to the left of the variable name. The CTS Variable Detail Panel opens.

```
----- CTS Variable Detail Panel -----
COMMAND ==>
Name . . : OPT_EMAIL_DEFAULT_FROM
Value . . : RED.SMITH@MEGACO.COM
Last Update
  Timestamp. . : 20xx/271 12.25.34.23
  User ID. . . : SMIRE03
```

The CTS Variable Detail Panel displays the date and time when the variable was last updated on your system. The panel also shows the user ID of the user who made the update. For system and product variables that the product components update, the Last Update information reflects the date and time when the system component ran. The information also reflects the user ID that is associated with the job or task that updated the variable.

Add User Variables

To permit extensive customization of the emails or WTOs, create a user variable. When information is updated, you can update the variable. You do not update all emails or WTOs that use the information.

Follow these steps:

1. From the Variable Type menu, select **1**.
The CTS Variable List panel opens.
2. Enter **A (ADD)** in any line command field (CMD) to the left of the variable name.
The CTS Variable Add panel opens.

```
----- CTS Variable Add Panel -----  
COMMAND ==>  
Name . . : ABC_COMPANY_DEPT_ID  
Value . . : ABC 506309
```

Enter a user variable name and the associated character data value and press Enter.

The variable is saved and the CTS Variable List panel opens.

Note: To cancel the variable addition, enter **CANCEL** or **CAN** in the command line or press PF3 before you press Enter.

Delete User Variables

You can delete variables that you no longer use.

On the CTS Variable List panel, enter **D (DELETE)** in the line command field (CMD) to the left of the variable that you want to delete.

The CTS Variable Delete Panel opens. Press Enter.

Note: You can cancel the delete command. Enter **CANCEL** or **CAN** in the command line or press PF3.

Update User or Options Variables

To change the values of the user or options variables, update the variables.

Follow these steps:

1. On the CTS Variable List panel, enter **U (UPDATE)** in the line command field (CMD) to the left of the variable to be updated.

The CTS Variable Update panel opens.

2. Enter the variable data in the Value field and press Enter.

The variable is saved with the updated value. The CTS Variable List panel opens to the updated variable.

Note: You can cancel the update. Enter **CANCEL** or **CAN** in the command line or press PF3.

Volume Pool Record Maintenance

The Volume Pool Monitor feature of CA 1 enables you to create sets of tape volumes. The sets are known as volume pools. You can monitor the availability of tapes in those sets in various states of use – Active, Scratch, or Not-in-Service. You can also monitor the availability of new tapes that have never been used. You create and monitor volume pools in ISPF and they are saved in the CA 1 VSAM Database (VDB).

Create or Update a Volume Pool

To add or modify a set of tape volumes, create or update a volume pool.

Follow these steps:

1. In the CTS Primary Menu, enter **2**.

The Volume Pool List panel opens. This panel shows all volume pools that are defined for this shared CA 1 environment. The panel also shows the number of ranges for each volume pool and the total number of volumes for the volume pool.

----- Volume Pool List -----			Row 1 of 15
COMMAND ==> ADD			SCROLL ==> PAGE
Commands: 'END', 'CANCEL', 'UP', 'DOWN', 'ADD'			Date: xx/09/28
Line Commands: 'A', 'D', 'R', 'S'			Time: 16:19
CMD	Volume Pool ID	Num of Ranges	Total Volumes
-	-----	-----	-----
	ABC_pool	4	11,404
	DEVLPOOL	3	100
	ISYS_POOL2	1	0
	Pool_A	3	200
	QAP00L	3	200
	QAP00L2	1	100
	Virtual_Pool	1	100
	SY01_POOL_1	3	41,250
	SY01_POOL_2	1	27,042

2. Enter **ADD** in the command line at the top.

Note: If you already defined at least one volume pool, you can enter **A** in the line command (CMD) field.

The VPM Range Add panel opens.

----- VPM Range Add Panel -----	
COMMAND ==> ADD	
Pool ID	. . New_Pool
LO VSN.	. . 000100
HI VSN.	. . 000199

3. Enter the volume pool name in the Pool ID field and the volume serial number (VSN) range in the LO and HI fields and press Enter.

The Volume Pool List panel opens.

4. Add more ranges for this pool by specifying an **A** on the line command (CMD) field.
5. Enter **R** in the command line.

The Volume Pool Range List panel opens. This panel includes a scrollable list of all ranges that are defined for the volume pool.

```

----- Volume Pool Range List ----- Row 1 of 2
COMMAND ==>                                SCROLL ==> CSR
Commands: 'END', 'CANCEL', 'UP', 'DOWN'      Date: yy/10/22
Line Commands: 'A', 'D', 'S'                 Time: 16:13
Volume Pool ID: New_Pool
CMD Low      High      Volumes
-----
      000100  000199      100
      VT0010  VT0099       90
***** Bottom of data *****

```

6. Add more ranges with the **A** line command or delete a range with the **D** command.

Notes:

- Use caution when defining new volume pools or new volume ranges. The new volume ranges can trigger alerts that are associated with percentage value conditions.
- The counters that are associated with a new volume pool or volume pool range are correct after the VPM subtask is stopped and started or the REBUILD command is issued. The TMC is then rescanned and all volume pool counters are rebuilt. New volume pools and volume pool ranges show 0 counters before the TMC is scanned.

How to Monitor Volume Pool Activity

After you create volume pools, you can view the counts of volumes in the pools on the CTS ISPF Volume Pool panels. To see totals for all volumes in all ranges for a volume pool, enter **S** on the Volume Pool List panel. The Volume Pool Detail panel opens.

```

----- VPM Pool Detail Panel -----
COMMAND ==>
Pool ID . . . : New_Pool
              Total      Active      Scratch  Out of Srv  Never used
-----
Current counts:  190        151        39         0         0
Highest counts:         151        39         0         0
Lowest counts:         151        39         0         0
Counts Updated: yyyy/296 16.11.12.13
              ---- Counts are for all Volume Ranges in this Pool ----
Last Update
Timestamp . . : yyyy/286 09.25.32.73
User ID. . . : SMITHJ5

```

The VPM subtask creates the counters that are presented in this display. VPM scans the TMC when it is first initialized or when the VPM REBUILD command is requested. As VPM scans the TMC, it examines each volume and categorizes it as Active, Scratch, or Out of Service. A total volume count for the volume pool is maintained as well as a separate count for Never Used volumes. The Active, Scratch, and Out of Service counts add up to the total number of volumes in the volume pool. The Never Used count is separate.

The Counts Updated field shows the date and time when the VPM subtask last updated the counters.

To monitor the activity for a specific range in the volume pool, enter an **R** in the line command on the volume pool list. The Volume Pool Range List panel opens. On this panel, enter **S** to display the counters for a specific range in the Volume Pool Range Detail panel.

```

----- VPM Range Detail Panel -----
COMMAND ==>
Pool ID . . . : New_Pool
Range . . . . : 000100 - 000199
              Total      Active      Scratch  Out of Srv  Never used
-----
Current counts:  100        61        39         0         0
Highest counts:         61        39         0         0
Lowest counts:         61        39         0         0
Counts Updated: yyyy/296 16.11.12.11
Last Update
Timestamp . . : yyyy/286 09.16.48.39
User ID. . . : SMITHJ5

```

For both the Volume Pool Detail Panel and the VPM Range Detail Panel, three sets of counters are maintained: Current counts, Highest counts, and Lowest counts. The Current counts row presents current counts for the volumes in the pool or range. The Highest counts and Lowest counts rows present the highest and lowest number of volumes for the pool or range. The numbers reflect the state of the last synchronization with the TMC at the startup of the VPM subtask or when a REBUILD command was issued.

Volume Pool Alert Maintenance

The Volume Pool Monitor feature enables you to create alerts that trigger the sending of an email or issue a WTO message to the console when user-specified conditions are met. You can associate alerts with volume pools and product variables that are updated by selected CA 1 jobs or tasks. You can create and update alerts in ISPF. You save alerts in the CA 1 VSAM Database (VDB).

Create or Update an Alert

To be alerted when certain conditions that you specify are met, create an alert.

Follow these steps:

- 1. On the CTS Primary Menu, enter **3**.

The VPM Alert List panel opens.

```
----- VPM Alert List ----- Row 1 of 19
COMMAND ==> SCROLL ==> PAGE
Commands: 'END', 'CANCEL', 'UP', 'DOWN', 'ADD' Date: yy/09/28
Line Commands: 'A', 'D', 'S', 'U' Time: 16:46
CMD Alert ID Pool or Variable ID
-----
LOW_SCRATCH_01 New_Pool
LOW_SCRATCH_CRITICAL ZOOMPOOL
BOBSPool BOBSPool
FAISALS_SCRATCH_LOW DEVLPOOL
FAISALS_SCRATCH_LOW2 DEVLPOOL
Johns_CA1_Alert BADCNT
TMSCLEAN_Exception-1 TMSCLEAN_Count
TMSCLEAN_Exception-2 TMSCLEAN_Count
TMSCLEAN_Exception-3 TMSCLEAN_RC
A VicksAlert01 Pool_A
```

- 2. On the VPM Alert List panel, enter **ADD** in the command line at the top. If you already have defined at least one alert, enter **A** in the line command (CMD) field.

When you specify **A** on the CMD field, the detailed variables that are associated with that alert are copied to your new alert as defaults. You can then modify the variables.

When you specify **A** in the CMD field or **ADD** in the command line at the top, the VPM Alert Add Panel opens.

----- VPM Alert Add Panel -----	
COMMAND ==>	
Alert ID. . .	My_Alert_Name
Pool ID . . .	POOL_ABC
Variable. . .	Scratch_Count
Oper.	LT
Value	500
Send.	WT0001
Interval. . .	ONCE

- 3. Specify the required values using the following rules. You can create an alert using the variables that are associated with a specific volume pool (identified in Pool ID on this screen). The alert can also test the condition of a user variable that you created.

Alert ID

Specifies a name for this alert. Use case-sensitive alphanumerics. The Alert ID can be from 1 to 32 characters long and can also include dashes and underscores.

Pool ID

Specifies the volume pool name. If the alert is not associated with a volume pool, leave this field blank. The Pool ID can be from 1 to 32 characters long and can also include dashes and underscores.

Variable

Specifies the name of the variable that you want to test. For alerts associated with volume pools, use one of the following names:

Active_Count

Active_Percent

Scratch_Count

Scratch_Percent

Out_of_Service_Count

Out_of_Service_Percent

Never_Used_Count

Never_Used_Percent

If the variable is not associated with a volume pool, enter any user or product variable with a numeric value. Do not set up alerts using variables with character strings. For example, you can specify a product variable such as TMSVMUPD_TAPES_VAULTED because a numeric value is associated with it. You cannot specify a product variable such as TMSCOPY_VOLUME_SERIAL_NUMBER because it is not a numeric value.

Oper

Defines the operand that you want to use in the variable test. Use one of the following values:

LT

less than

LE

less than or equal

EQ

equal

GE

greater than or equal

GT

greater than

Value

Specifies the value that you want to test against when performing the comparison.

Send

Specifies the member name of the WTO or email that you want to send. This field must be a member of the CAI.SEND data set. The field must conform to the email or WTO construction rules. The data set name of CAI.SEND must be specified in the VPM Option variable OPT_EMAIL_DEFAULT_DSN.

Interval

Specifies how often the alert is issued. Use one of the following values as the interval:

ONCE

The alert WTO or email is issued only once when the alert condition is first met.

WHILE

The alert WTO or email is issued each time that the alert is tested and the alert condition is met. How often the alert is tested is based on the setting of the VPM Option variable OPT_VPM_TIMER.

HH

Specifies the number of hours to wait between issuing an alert WTO or email while the alert condition is met. The alert condition is tested based on the setting of OPT_VPM_TIMER. HH hours must have passed since this variable was last issued and the condition was met.

4. Use the **U** line command on the VPM Alert List panel.

The alert updates. The fields for the Update panel are the same as the Add panel. The Last Update Timestamp and User ID are also presented. The Last Update Timestamp and User ID fields are not updateable. They reflect the last time that maintenance to this alert was performed and the user who did the maintenance. All preceding fields except for the Last Update fields and the alert ID can be updated.

Note: To change an alert ID, create an alert. Use the line command **A** on the alert that you are trying to rename. The action creates an alert that is populated with the old variables. To delete an alert, use the **D** line command on the VPM Alert List panel.

Note: The ISPF application does not precheck whether the variables that you specify in the Pool ID, Variable, and Send fields exist. You can construct alerts that reference volume pools, variables, and CAI.SEND data set members that do not exist and express conditions that are never met. Enter the correct name. For volume pools and variables, mixed case is supported for these fields and the case must match exactly.

More information:[Create a WTO](#) (see page 276)[Create an Email](#) (see page 278)

Tape Inquiry (TI) Display

A quick method for checking on the status of a CA 1 cataloged tape data set from the ISPF 3.4 Data Set List (DSLST) display is also provided. This panel shows basic information from the TMC for the cataloged data set in a format that is consistent with other ISPF 3.4 data displays. You can quickly get information about a cataloged data set without starting the full CA 1 ISPF application.

By entering "TI" on the line command field for a specific data set presented in the DSLST panel, the following panel displays:

```
----- CA 1 TAPE INQUIRY -----  
  
Data Set Name . . . : USER001.TESTMOD.PDS1  
  
General Data                               Creation Information  
Volume Serial. . . : 202583                Date . . . . . : YYYY/MM/DD  
Alternate Volume . : A02583                Time . . . . . : 01:00:00  
Media type . . . : 3480-18TK              Jobname . . . . : IEBDG  
Record Format. . . : F                    Program . . . . : CTSTAPER  
Record Length. . . : 24                  Last Used Information  
Block Size . . . : 24                    Date . . . . . : YYYY/MM/DD  
Number of blocks . : 4                    Time . . . . . : 00:03:00  
Percent utilized . :                      Jobname . . . . : CATMG  
Batch ID/Hook ID . :                      Expiration Information  
Status . . . . . : ACTIVE                 Expire Date. . . : YYYY/MM/DD  
  
File / Volume Set                         Vault Management Information  
Base Volume. . . . : 202583                Outcode. . . . . : XA11  
Sequence Number. . : 1                    Slot . . . . . : 0012345  
Total Files in Set : 1                    Outdate. . . . . : YYYY/MM/DD  
Secondary volumes: 102584 102585 102586 102587 102588 (2 MORE)  
User Data: ACCOUNT 000455
```

Note: The above display is read-only.

TIQ Online Inquiry/Update

The TIQ Online Inquiry/Update facility provides an alternate online access method that is supported on more terminals: TSO or CICS transaction, CA Roscoe, system console (started task) and CA 7 Job Management console. Inquiries can be based on the volume serial number, the DSNB number, or the cataloged data set name. The built-in security feature can be tailored to fit your installation needs.

The automatic timeout limit for the system console and TSO facilities is defined by the TW parameter in the TMOOPTxx member of *hlq.CTAPOPTN*.

Note: When the audit file is in BLOCKSET mode, all audit dates and times are in GMT/UTC.

TMC Record Access

TMC volume, DSNB and control records can be accessed when the following information is supplied:

Volume record

Defines the volume serial number or cataloged data set name.

DSNB record

Defines the DSNB number or cataloged data set name.

Control record

Defines the control record identifier.

When inquiring by cataloged data set name, the TMC volume record for that data set is displayed if it is the first file. The DSNB record is displayed for the data set if it is a secondary file.

All fields of a volume or DSNB record are displayed and are available for subsequent updating, except the scratch bit field, which cannot be modified. When an inquiry is performed on a multivolume data set or multi-data set volume, a list of secondary volume or DSNB records can optionally be displayed. Control records can only be displayed; updating is not permitted.

Specific Field Access

Any field within a volume or DSNB record can be displayed and updated by specific request. Access is accomplished by specifying the volume serial or DSNB number and the fields to be displayed.

Up to 10 keywords (fields) can be specified on a single inquiry/update command. Special utility commands (see Special Utility Commands) typically update more than one field in the TMC, and these fields count toward the limit of 10.

Note: For more information about keywords and their meanings, see the *Programming Guide*.

TIQ Online Inquiry and Update

Each console type (MVS, CA Roscoe, TSO, and so forth) requires a *driver* program to handle the interface between the terminal and TIQ. For example, the driver program for the MVS console, TMSTOSTQ, is invoked by a started task, TMSOSTQ, while the driver for a TSO terminal, TMSTSO, is called from a CLIST.

Once activated, TIQ issues the following message:

System Console	Non-System Console
IEETMS1 ENTER PASSWORD	IEETMS1 ENTER PASSWORD
nn IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)	IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)

The value nn is the message number of the MVS console request for input. The password must be entered. Three attempts to enter a correct password are allowed. If an incorrect password is entered, TIQ responds with:

System Console	Non-System Console
nn IEETMS1 INVALID PASSWORD	IEETMS1 INVALID PASSWORD

Once a valid password is entered, the following message is issued:

System Console	Non-System Console
nn IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)	IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)

In the event a volume has been password-protected through the Data Set Security feature, any attempt to inquire on or update the volume record without providing the assigned password generates the following prompt:

System Console	Non-System Console
nn IEETMS1 ENTER VOLUME PASSWORD	IEETMS1 ENTER VOLUME PASSWORD

Three attempts to specify the correct password are allowed before the session is terminated.

TMC Inquiry by Volume Serial Number

This type of inquiry is used when the volume serial number is known but the data set on the volume is not, or when the data set is not cataloged. Enter the following command:

System Console	Non-System Console
nn,'076777'	076777

If the volume requested is the first volume of a multivolume data set, the volume information is displayed and the following prompt is issued:

System Console	Non-System Console
ENTER V TO REVIEW MULTIPLE VOLUMES	ENTER V TO REVIEW MULTIPLE VOLUMES
nn IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)	IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)

If the volume requested contains multiple files, the volume information is displayed and the following prompt is issued:

System Console	Non-System Console
ENTER F TO REVIEW MULTIPLE FILES	ENTER F TO REVIEW MULTIPLE FILES
nn IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)	IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)

A response of **V** to the multivolume prompt provides a list of volumes associated with the data set, while a response of **F** to the multiple file prompt displays a list of the DSNBs associated with the other files on the volume.

The keyword **LONG** requested on a volume display (volser, **LONG**) causes a general display of each volume in the group if the volume serial requested is the first in a set, and then a display of all associated DSNB records. Up to 60 lines of information can be processed in a single request. When the number of lines in the display exceeds this limit, the following prompt is issued:

System Console	Non-System Console
ENTER P TO CONTINUE OR NEXT REQUEST	ENTER P TO CONTINUE OR NEXT REQUEST
nn IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)	IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)

A response of **P** to the prompt continues the display.

TMC Inquiry by DSNB Number

Inquiry on a DSNB record can be performed by DSNB number when the data set name is not known or not cataloged:

System Console	Non-System Console
nn,'DSNB 2'	DSNB 2

TMC Inquiry on Control Records

Inquiry on the contents of TMC Control records can be performed with the following command:

System Console	Non-System Console
nn,'CTLx'	CTLx

The value **x** is the Control record number to be displayed. Valid numbers are 1, 2, 3 and 4.

TMC Inquiry by Cataloged Data Set Name

This inquiry function can be used when the data set name is known and has an entry in the MVS catalog but the volume serial number/DSNB number is unknown:

System Console	Non-System Console
nn,'data.set.name[(gg)]'	data.set.name[(gg)]

Specification of a relative generation (gg) is permitted, such as DATA.SET.NAME(0) or DATA.SET.NAME(-5). Response from TIQ is as follows:

```
DSN=data.set.name.GooooVuu d-TRK,FS=nnn
```

The value oooo is the generation number (if applicable), and the value uu is the version number. The value of d-TRK indicates how many tracks of data were written on the tape (7, 9, 18, 36, 128, 256, or 384) and FS=nnn is the file number. If the data set is on the first file, TIQ returns the volume record display. If it is a secondary data set, the contents of the DSNB record associated with this file are displayed.

If there is no entry in the system catalog for the data set entered, TIQ issues the following message:

```
LOCATE FAILED-DSN=data.set.name INQUIRY REJECTED
```

If there is an entry but the volume serial number is not defined to the TMC, the response from TIQ is:

```
DATA SET NOT ON TMC VOLUME INQUIRY REJECTED
```

TMC Inquiry on Specific TMC Record Fields

If only certain fields in a record are required for display, the specified keyword name for that field may be passed with the volume serial or DSNB number. Up to 10 keywords can be specified on a single request.

Note: For more information about the fields, see the *Programming Guide*.

System Console

```
nn, 'vvvvvv,keyword1,keyword2,...,keywordn'
```

Non-System Console

```
vvvvvv,keyword1,keyword2,...,keywordn
```

TIQ responds with the following:

```
VOLSER=vvvvvv DSN=data.set.name  
keyword1=value1  
.  
.  
.  
keywordn=valuen  
INQUIRY COMPLETE
```

TMC External Tape Label Request Generation

The EXLBL inquiry command can be used to generate an external label request using the Online Label Interface for processing and printing the labels. The label request can be generated for a volume serial number or cataloged data set name, using the following inquiry command:

vvvvvv,EXLBL - generates label request by volume serial number

dsname,EXLBL - generates label request by cataloged data set name

The data set name may be specified as a relative generation, for example, 'DATA.SET.NAME(-1)'. You must have authorization in the Security Table to be able to execute this command. TIQ responds with:

ENTER V TO PRINT MULTIPLE VOLUMES

If the volume is part of a multivolume data set, and then responds with:

LABEL REQUEST FOR VOLSER=xxxxxx PROCESSED BY ONLINE LABELS INTERFACE
INQUIRY COMPLETE

External label requests can be generated for any volume within the TMC ranges. Information for the label is extracted directly from the TMC record.

If the Online Label Interface is not available at the time a label is requested, the following message is generated when a label is requested.

LABEL REQUEST FOR xxxxxx FAILED -ONLINE LABELS INTERFACE INACTIVE
INQUIRY COMPLETE

Update the TMC with TIQ

TIQ allows updates to keyword fields in the TMC records.

Note: The supported keywords and their meanings are described in the *Programming Guide*.

When an update request is issued for a TMC record, CA 1 first responds with the current status and contents of the field(s) for which the update was requested. No change is made to a volume or DSNB record until the change is confirmed by the operator. The operator is always provided an opportunity to confirm or cancel the update request prior to any action being taken by TIQ on a TMC record.

When TIQ is initialized, parameters can be passed to define whether updates can be performed only if the data set name currently stored in the TMC record is supplied with the update request. If the data set name verification is not established, and a data set name is provided in an update request, the data set name field is modified as specified.

Volumes that are under CA 1 Data Set Security password protection can be updated only if security requirements are met.

Updates to certain fields of multivolume data sets can be performed *only* on the first volume in the set. The update is applied to all chained volumes. Update is prohibited for TMC records associated with tapes owned by CA Dynam/T.

Some volume record fields may be rejected for update, depending upon the media type of the volume.

The following TMC fields can only be updated for WORM tapes:

- WWID (physical WORM tapes only)
- VWWID (virtual WORM tapes only)
- WMC

The following TMC fields can only be updated for non-WORM tapes:

- EDMID (also valid for virtual WORM tapes)
- DATECLN
- USECLN
- CLNCNT
- TRERRC
- TWERRC
- PRERRC
- PWERRC
- TRERRI (also valid for physical WORM tapes)
- TWERRI (also valid for physical WORM tapes)

Update Volume Records

The following command is used to request a field update:

System Console

```
nn, 'U, vvvvvv, [DSN=current.dsn, ]keyword1=value1, . . . , keywordn=valuen
```

Non-System Console

```
U, vvvvvv, [DSN=current.dsn, ]keyword1=value1, . . . , keywordn=valuen
```

TIQ responds with:

```
VOLSER=vvvvvv DSN=data.set.name
      keyword1 oldvalue WILL BE CHANGED TO value1
      .
      .
      .
      keywordn oldvalue WILL BE CHANGED TO valuen
```

REPLY 'U' OR 'R'

IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)

A response of U allows TIQ to update the record while a response of R rejects the update request. U or R must be provided before another inquiry or update transaction can be requested. Should the response be R, TIQ issues the message:

UPDATE REJECTED

Update DSNB Records

The following command can be issued to request a DSNB field update:

System Console

```
nn, 'U,DSNB  
xxxx, [DSN=current.dsn], keyword1=value1, ..., keywordn=valuen'
```

Non-System Console

```
U,DSNB xxxx, [DSN=current.dsn], keyword1=value1, ..., keywordn=valuen
```

The value xxxx is the number of the DSNB record to update. TIQ responds with the following:

```
keyword1 oldvalue WILL BE CHANGED TO value1  
.  
.  
.  
keywordn oldvalue WILL BE CHANGED TO valuen  
REPLY 'U' or 'R'  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

As with volume record updates, a reply of U directs TIQ to perform the update, and a reply of R rejects the request.

The UPDATE REJECTED message is also produced if one of the input parameters is in error. In either case, the entire update is rejected and all parameters in the update must be reentered correctly.

Special Utility Commands

As a time-saving feature, special update and inquiry commands are provided for frequently recurring activities. You must be authorized (through the Security Table) to issue these commands.

Volume Record Commands

ADD

Activates an inactive (deleted) volume serial number to make it available for use. The volume serial number must be in the TMC and in inactive (delete) status.

CHECKIN

Checks in a tape that was previously marked out-of-area.

CLEANED

Updates cleaning statistics to indicate that a tape volume has been cleaned. This function tells CA 1 that a tape was cleaned before it was marked as needing cleaning on the Scratch and Clean List.

Note: The CLEANED utility cannot be used for 3592 WORM tapes.

DELETE

Marks a tape volume as inactive (deleted) when the volume is in scratch status, making the volume unavailable for use and for most reporting.

EXLBL

Generates an external label request for a volume serial number or a cataloged data set. The label request is routed to the Online Label Interface for processing and printing.

EXPIRE

Changes the expiration date of a volume to the current system date.

EXTEND

Extends the expiration date of a volume for a specified number of days from the current expiration date.

RETAIN

Sets the expiration date of a volume to a specified number of days from the creation date.

Note: The CHECKOUT command is not available for TIQ. To check out a tape, update the volume record with the desired OUTCODE. The OUTDATE is automatically updated to the system date.

DSNB Record Commands

EXPIRE

Changes the expiration date for the DSNB to the current date.

EXTEND

Extends the expiration date of a DSNB for a specified number of days from the expiration date.

RETAIN

Sets the expiration date of a DSNB to a specified number of days from the creation date.

In addition to the special utility commands, a stand-alone tailoring command is available for use with all displays:

DATEFMT='fmt'

Specifies the date format that is accepted for all external dates.

The special utility commands and the DATEFMT command are discussed in detail on the following pages. They are presented in alphabetical order for easy reference.

Note: Updates to volume or DSNB record fields cannot be combined with the special utility requests discussed in the next section.

ADD Command—Activate a Volume Record Marked as Inactive

Use the ADD command to activate an inactive (deleted) volume serial number to make it available for use. The volume serial number must be in the TMC and in inactive (delete) status.

From the system console, this command has the following format:

```
nn, 'U, vvvvvv, ADD[ , DSN=dsname] '
```

From a non-system console, this command has the following format:

```
U, vvvvvv, ADD[ , DSN=dsname]
```

If the tape is not in inactive status, TIQ responds with:

```
VOLSER=vvvvvv DSN=dsname  
VOLUME IS NOT IN DELETE STATUS  
UPDATE REJECTED VOLSER=vvvvvv  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

If the record is in inactive status, TIQ responds with:

```
VOLSER=vvvvvv DSN=* THIS TAPE IS IN DELETE STATUS *  
VOLUME WILL BE CHANGED TO ACTIVE STATUS  
REPLY 'U' OR 'R'  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

CHECKIN Command—Check in a Tape Marked Out-of-area

Use the CHECKIN command to check in a tape that was previously marked out-of-area.

This command has the following format:

From the system console, this command has the following format:

```
nn, 'U, vvvvvv, CHECKIN[ , DSN=dsname ] '
```

From a non-system console, this command has the following format:

```
U, vvvvvv, CHECKIN[ , DSN=dsname ]
```

If the tape is part of a multivolume chain, the CHECKIN is required to be performed against the first volume and updates all volumes in the chain. TIQ responds with the following:

```
VOLSER=vvvvvv DSN=dsname  
OUTCODE=oldvalue WILL BE CHANGED TO BLANKS  
OUTDATE=oldvalue WILL BE CHANGED TO ZEROS  
SLOT=oldvalue WILL BE CHANGED TO HEXZEROS  
REPLY 'U' OR 'R'  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

If the tape being checked in was under CA 1 Vault Management control (a tape with an OUTCODE and a SLOT number), then the field update to OUTCODE is changed to (blank)VMS, allowing the tape not to be recycled to the off-site vault until it has first been read for input.

CLEANED Command—Update Cleaning Statistics

Use the CLEANED command to update cleaning statistics to indicate that a tape volume has been cleaned.

This function tells CA 1 that a tape was cleaned before it was marked as needing cleaning on the Scratch and Clean List.

Note: The CLEANED utility cannot be used for 3592 WORM tapes.

CA 1 marks tapes as needing to be cleaned during scratch processing (TMSCLEAN) based on user-defined criteria. Should it be necessary to clean a tape before TMSCLEAN indicates it is required, the CLEANED command should be issued to prevent TMSCLEAN from marking the volume prematurely.

From the system console, this command has the following format:

```
nn, 'U, vvvvvv, CLEANED[ , DSN=dsname ] '
```

From a non-system console, this command has the following format:

```
U, vvvvvv, CLEANED[ , DSN=dsname ]
```

TIQ responds with the following:

```
VOLSER=vvvvvv DSN=dsname  
USECLN=oldvalue WILL BE CHANGED TO current.use.count  
DATECLN=oldvalue WILL BE CHANGED TO current.date  
CLNCNT=oldvalue WILL BE CHANGED TO oldvalue+1  
FLAG1=oldvalue WILL BE CHANGED TO value  
TRERRC=oldvalue WILL BE CHANGED TO HEXZEROS  
TWERRC=oldvalue WILL BE CHANGED TO HEXZEROS  
PRERRC=oldvalue WILL BE CHANGED TO HEXZEROS  
PWERRC=oldvalue WILL BE CHANGED TO HEXZEROS  
REPLY 'U' OR 'R'  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

Note: The CLEANED utility cannot be used for 3592 WORM tapes.

DATEFMT Command—Specify the Date Format

Use the DATEFMT='fmt' command to specify the date format that is accepted for all external dates. The date pattern indicated by 'fmt' is limited to 10 bytes and must be enclosed in single quotes. All dates that are entered, printed or displayed will appear in this format.

Note: For more information about valid date formats, see the *Utilities and Reports Reference Guide*.

From the system console, this command has the following format:

```
nn,DATEFMT='fmt '
```

From a non-system console, this command has the following format:

```
DATEFMT='fmt '
```

If the date format is valid, TIQ responds with:

```
DATEFMT='fmt '  
DATE  FORMAT SET  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

If the date format is invalid, TIQ responds with:

```
DATEFMT='fmt '  
** INVALID DATE  FORMAT ***
```


DELETE Command—Mark a Tape Volume as Inactive

Use the DELETE command to mark a tape volume as inactive (deleted) when the volume is in scratch status, making the volume unavailable for use and for most reporting.

The DELETE command can be used, assuming the volume is currently in scratch status.

From the system console, this command has the following format:

```
nn, 'U, vvvvvv, DELETE [ , DSN=dsname] '
```

From a non-system console, this command has the following format:

```
U, vvvvvv, DELETE [ , DSN=dsname]
```

If the volume is not in scratch status, TIQ responds with:

```
VOLSER=vvvvvv DSN=dsname  
VOLUME IS NOT IN SCRATCH STATUS  
UPDATE REJECTED  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

If the volume is in scratch status, the response is:

```
VOLSER=vvvvvv DSN=dsname  
VOLUME WILL BE CHANGED TO DELETE STATUS  
REPLY 'U' OR 'R'  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

EXLBL Command—Generate an External Label Request

Use the EXLBL command to generate an external label request for a volume serial number or a cataloged data set. The label request is routed to the Online Label Interface for processing and printing.

From the system console, this command has the following format:

```
nn, '980001, EXLBL '
```

From a non-system console, this command has the following format:

```
980001, EXLBL
```

TIQ responds with the following:

```
980001, EXLBL  
LABEL REQUEST FOR VOLSER=xxxxxx PROCESSED BY ONLINE LABELS INTERFACE
```

Note: For more information, see [Common Tape System](#) (see page 187).

EXPIRE Command—Change Expiration Date to Current Date

Use the EXPIRE command to change the expiration date of a volume or DSNB record to the current date.

From the system console, this command has the following format:

```
nn, 'U, vvvvvv, EXPIRE[ , DSN=dsname] '  
nn, 'U, DSNB xxxx, EXPIRE[ , DSN=dsname] '
```

From a non-system console, this command has the following format:

```
U, vvvvvv, EXPIRE[ , DSN=dsname]  
U, DSNB xxxx, EXPIRE[ , DSN=dsname]
```

TIQ responds with the following:

```
VOLSER=vvvvvv DSN=dsname  
EXPDT=oldvalue WILL BE CHANGED TO current.date  
REPLY 'U' or 'R'  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

EXTEND Command—Extend the Expiration Date

Use the EXTEND command to extend the expiration date of a volume for a specified number of days from the current expiration date.

Extending the expiration date of a volume or DSNB record a specified number of days from the expiration date can be performed with the following commands:

From the system console, this command has the following format:

```
nn, 'U, vvvvvv, EXTEND, dddd[ , DSN=dsname] '  
nn, 'U, DSNB xxxx, EXTEND, dddd[ , DSN=dsname] '
```

From a non-system console, this command has the following format:

```
U, vvvvvv, EXTEND, dddd[ , DSN=dsname]  
U, DSNB xxxx, EXTEND, dddd[ , DSN=dsname]
```

The value dddd is the number of days to extend the expiration date. TIQ responds with the following:

```
VOLSER=vvvvvv DSN=dsname  
EXPDT oldvalue WILL BE CHANGED TO oldvalue+dddd  
REPLY 'U' OR 'R'  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

RETAIN Command—Set Expiration Date from Creation Date

Use the RETAIN command to set the expiration date of a volume to a specified number of days from the creation date.

Setting the expiration date of a volume or DSNB record to a specified number of days from the creation date can be performed with the following commands.

From the system console, this command has the following format:

```
nn, 'U, vvvvvv, RETAIN, dddd[ , DSN=dsname] '  
nn, 'U, DSNB xxxx, RETAIN, dddd[ , DSN=dsname] '
```

From a non-system console, this command has the following format:

```
U, vvvvvv, RETAIN, dddd[ , DSN=dsname]  
U, DSNB xxxx, RETAIN, dddd[ , DSN=dsname]
```

The value dddd is the number of days to retain the data set from the tape's creation date. TIQ responds with the following:

```
VOLSER=vvvvvv DSN=dsname  
EXPDT=oldvalue WILL BE CHANGED TO creation.date+dddd  
REPLY 'U' OR 'R'  
IEETMS1 AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

Console-Unique Procedures

This section contains initialization and termination procedures from the supported consoles.

Initialize and Terminate TIQ from an MVS Console

The task TMSOSTQ is initialized by executing a procedure from SYS1.PROCLIB (or a procedure library accessible to JES) using the START command. This task invokes the program TMSTOSTQ. To establish communication with the TIQ facility, the following command is issued:

```
START TMSOSTQ[ ,ROUTCDE=nn] [ ,VERIFY=NODSN] [ ,DATEFMT='fmt' ]
```

ROUTCDE is used by MCS users only. Replace with the MCS route code of the console which issued the START command if the route code differs from the default route code in the procedure.

VERIFY=NODSN is used to allow the updating of a TMC record without verifying the data set name for that record. If the data set name is being changed, NODSN must be specified. Otherwise, the DSN= statement must be specified twice, once for verification, and again for the update. If this parameter is not specified, the default value (VERIFY=DSN) is assumed and each update will require data set name verification.

DATEFMT='fmt' specifies the date format that is accepted for all external dates. The date pattern indicated by 'fmt' is limited to 10 bytes and must be enclosed in single quotes. All dates that are entered, printed or displayed will appear in this format. Dates in Julian format (yyyy/ddd) are also accepted.

TMSOSTQ can be terminated by any one of the three following procedures:

- Reply C to the outstanding TMSOSTQ request for input (nn,'C'), where nn is the message number of the TMSTIQ request for input.
- Enter CANCEL TMSOSTQ from the calling console.
- Cease requests to TMSOSTQ and allow TMSOSTQ to time out. (The CA 1 parameter TW in member TMOOPTxx in *hlq*.CTAPOPTN defines the TIQ timeout limit.) TMSOSTQ terminates itself.

When TMSOSTQ is terminated by one of the three above methods, TMSOSTQ responds with the following:

```
UPDATE/INQUIRY COMPLETE  
TMS INQUIRY CANCELLED
```

Initialize and Terminate TIQ with TSO

TIQ Online Inquiry/Update using TSO is available if the following CLIST, member name TSOTIQ, is added to the CLIST library.

```
PROC 0 D(DSN) P(PROMPT) DATEFMT(DEFAULT)
CALL 'CAI.CTAPLINK(TMSTSO)' '&D.,&P.,DATEFMT=' '&DATEFMT' '.
END
```

&D. is the data set name verification option parameter. A data set name verification option value of NODSN is used to allow updating of a TMC record without verifying the data set name for that record. If the data set name is being changed, NODSN must be specified. Otherwise, the DSN= parameter must be specified twice, once for verification, and again for the update. If DSN is specified, each update will require data set name verification.

&P. is the prompt for the access password parameter. The option parameter value PROMPT causes the prompt for the access password prior to invoking the external security interface. NOPROMPT should only be used when the optional security exit, TMSXITS, has been coded to supply a default access password.

&DATEFMT. represents the date format option for this session. In the example above, the date format pattern is set to DEFAULT. You can specify a preferred date format by replacing DEFAULT with the desired pattern, such as DATEFMT('MMMDD YYYY'). &DATEFMT. must be preceded by two single quotes and followed by three single quotes.

Aside from the initialization of TMSTSO, all user inputs and TMSTSO outputs are identical to those used for TMSTOSTQ, except that no outstanding system message reply numbers are used, and the IEETMS1 message prefixes are eliminated.

To initialize TMSTSO, issue the following command from the TSO READY mode or the SPF COMMAND function:

```
TSOTIQ [D(DSN|NODSN)] [P(PROMPT|NOPROMPT)] [DATEFMT('fmt')]
```

When TSOTIQ becomes active, the following message is issued:

```
IEETMS1 ENTER PASSWORD
AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

A password must be entered. Three attempts to enter the password are allowed before TIQ terminates. Once a correct password is entered, the following message is issued:

```
AWAITING REQUEST/REPLY (ENTER C TO EXIT CA 1 INQ/UPD SESSION)
```

Initialize and Terminate TIQ with CA Roscoe

TIQ Online Inquiry/Update using CA Roscoe is available if the module RSSCTIQ0 is in the CA Roscoe library.

Aside from the initialization and termination of TIQ, all user inputs and TIQ outputs are identical to those used for non-MVS console users of TMSTOSTQ.

Communication with TIQ is established by issuing one of the following commands from the CA Roscoe terminal:

RUN TIQ,/NODSN/[NODSN]

All TIQ commands are entered conversationally from the terminal.

RUN TIQ-A,/NODSN/

All TIQ commands are read from the current AWS.

RUN TIQ-A,dsname,/NODSN/

All TIQ commands are read from the CA Roscoe library data set dsname.

NODSN is used to allow updating of a TMC record without verifying the data set name for that record. If the data set name is being changed, NODSN must be specified. Otherwise, the DSN= statement must be specified twice, once for verification, and again for the update. If this parameter is not specified, the default value is assumed, and each update will require data set name verification.

To switch from conversational terminal entry to entry from the AWS or CA Roscoe library data set dsname, enter an asterisk (*) at any point during the terminal conversion.

TIQ can be terminated by any one of the following methods:

- Enter C to terminate request
- END of DATA from current AWS
- END of DATA from specified CA Roscoe library data set dsname
- Press ATTENTION (CLEAR) at any time

Chapter 8: Using the CA Dynam/T Interface

This chapter describes the CA Dynam/T Interface and identifies its unique operations and utilities. The material supplied is specifically directed to the use of this interface with CA 1 and toward performing common procedures under MVS. Operational and procedural information is supplied for the CA 1 systems operator and tape librarian.

This section contains the following topics:

[About the CA Dynam/T Interface](#) (see page 159)

[Communications](#) (see page 160)

[Database Synchronization](#) (see page 161)

[CA Dynam/T Interface Utilities](#) (see page 162)

About the CA Dynam/T Interface

The CA Dynam/T interface provides the means for sharing a common set of volumes with one common scratch pool between these two tape management systems.

When a volume from the common scratch pool is written on, the tape management system under which the volume was opened for output owns the volume. The owning system sends information to the sharing system, notifying it that tape is no longer available for use as a scratch tape. The owning tape management system can provide all of its services (such as vault management) for the volumes it owns. It is also responsible for scratching the volume when it expires and notifying the other system(s) that the volume is available for use.

When notification is received about a volume used by the other system(s), the non-owning system records the information about the volume and provides real-time protection for it. None of the other services provided by that tape management system are available for that volume (except some reporting and batch maintenance utilities). When notification is received that the volume is scratched, the volume is marked as scratch and is once again available to each system.

Communications

Communication between CA 1 and CA Dynam/T is accomplished through the use of queues allocated on a shared DASD device. Two queues are required to support the interface. Each system has a *sending* queue used to send transactions to the other system, and a *receiving* queue used to receive transactions from the other system; the sending queue for one system serves as the receiving queue for the other. Transactions are placed on the queue by one system and processed by the other.

Queue serialization between CA 1 and CA Dynam/T is accomplished using a technique referred to as record *locking*. This method allows serialization of access to the queue without using ENQs or hardware RESERVEs.

Queues are allocated on a DASD device and then formatted using the utility DYNFORMAT. Another utility, the DYNSYNC, is provided to aid in analyzing and resolving queue-related problems.

Note: For more information about these utilities, see the *Utilities and Reports Reference Guide*.

Database Synchronization

To exercise the CA Dynam/T interface, each system must define the same range(s) of volume serial numbers in its respective catalog. If the CA Dynam/T catalog contains volumes not defined in the CA 1 Tape Management Catalog (TMC), the CA 1 utilities TMSXTEND or TMSFORMT must be executed to extend the TMC and add the volumes. If the CA 1 TMC contains volumes not defined to CA Dynam/T, the undefined volumes are added to the CA Dynam/T catalog automatically when the database synchronization utility DYN SYNC is executed.

After being initially defined during installation, the databases of each interfacing system must be synchronized to establish that there are no differences. This synchronization must be maintained throughout all activity to ensure the integrity of the databases. Should queue-related problems prevent information from being shared between CA 1 and CA Dynam/T, a synchronization utility is provided to help resolve discrepancies between the databases.

An extract utility is supplied with each system to convert database information into a common format that can be compared to determine if any discrepancies exist between the systems. CA recommends that the CA Dynam/T VM extract be put on tape, as VM cannot write to a sequential data set using MVS access methods. Any other extract(s) is stored on DASD.

Volumes in delete status are not placed in the extract. If these volumes are active under CA Dynam/T, the synchronization utility generates a message indicating that the volume could not be found on the extract for CA 1.

The synchronization utility accepts the extracts from each system as input and performs the analysis. All discrepancies are indicated on an exception report and information relative to resolving the discrepancies is sent to the appropriate system for corrective action through separate data sets.

Each system provides a product-specific update utility used to apply synchronization information to its respective database. Any updates that cannot be performed are detailed in error reports. These errors may require additional research and manual correction.

CA Dynam/T Interface Utilities

The following utilities are unique to the operation of the CA Dynam/T interface:

DYNCOMM

Sends and receives transaction records using the common communications queue for the CA Dynam/T interface.

DYNEXTRA

Generates the TMC database extract to be used by DYNSYNC for the CA Dynam/T interface.

DYNRELSE

Releases the CA Dynam/T interface's common communications queue that has been locked as a result of a system failure.

DYNSYNC

Processes the database extracts from CA 1 and CA Dynam/T to determine differences for the CA Dynam/T interface.

DYNUPDTE

Processes the output from DYNSYNC which is used to update the TMC to resolve discrepancies for the CA Dynam/T interface.

Parameters for the CA Dynam/T Interface are in the TMODYNxx member in *hlq.CTAPOPTN*.

Note: For more information about the CA Dynam/T Interface, see the *Programming Guide*.

Chapter 9: Automated Tape Libraries and Virtual Tape Support

This section contains the following topics:

[Tape Robotics](#) (see page 163)
[TMC Fields Used to Track Tapes in Automated Tape Libraries](#) (see page 164)
[The IBM Tape Library](#) (see page 165)
[IBM Virtual Tape Server \(VTS\)](#) (see page 170)
[IBM Basic Tape Library Support \(BTLS\)](#) (see page 177)
[Share an IBM ATL or VTS](#) (see page 179)
[CA Vtape Virtual Tape System](#) (see page 183)
[Sun/STK Tape Libraries](#) (see page 185)
[Oracle/STK Virtual Storage Manager \(VSM\)](#) (see page 185)

Tape Robotics

Mainframe tape processing has evolved from labor-intensive manual tape processing into a simplified process, using Automated Tape Libraries (ATL) and Virtual Tape (VT) systems. This eliminates the hassles of retrieving reel tapes to mount on manual drives.

Using tape robotics, you can now quietly and efficiently perform mount and unmount requests for high capacity cartridges. Virtual Tape systems have further increased efficiency by eliminating the need to write to physical tapes at all. Instead, tape data is written to volumes in a disk cache. These virtual volumes are later consolidated and stacked on physical tape volumes and exported from the virtual tape system for disaster recovery or other purposes. CA 1 supports each new phase of tape technology whenever one is introduced, supporting the latest tape media types for both robotics and virtual tape systems.

Automated Tape Libraries(ATL) and Virtual Tape systems are now the standard in most installation situations. ATLs offer the following benefits:

- Faster mount times
- Minimization of tape handling, reducing the possibility of operator errors
- Easier tracking of tape inventory

Virtual tape provides an extension to the benefits of ATLs by providing these additional benefits:

- Improved use of physical tape cartridges
- Instantaneous dismount and rewind processing

These benefits drive the acceptance of robotics and virtual tape systems as the industry standard for efficient tape processing.

Note: The terms ATL, library and robot are used interchangeably. The meaning of these terms are all intended to be the same.

TMC Fields Used to Track Tapes in Automated Tape Libraries

CA 1 provides two fields in the Tape Management Catalog (TMC). These fields enable you to track the residency of tapes in ATLs and virtual tape systems. A third field is provided, but is only used for virtual tape solutions to record the physical volume serial number that a virtual tape is written to. The following fields are provided:

ROBTY (Robot type)

Identifies the manufacturer (IBM, STK and others) and if the robot is a virtual tape system. Specific bit settings are defined for the leading robotics and virtual tape vendors. The following is a partial list:

- X'01' IBM
- X'02' STK
- X'81' IBM Virtual Tape System
- X'82' STK Virtual Storage Manager
- X'88' CA Vtape Virtual Tape System

Note: For a complete list, see the *Programming Guide*.

ROBID (Robot identifier)

A robotic device indicator used to distinguish between multiple robots of the same type in an installation. This field is a field you assign. Typically, if an installation has three robots of the same type they are assigned the identifiers of 1, 2, and 3, however the identifiers can be assigned any number up to 255.

ACTVOL (Actual volume serial number)

The container volume serial number is used to record the physical volume serial number that one or more virtual volumes are written on. This field is not used for basic ATL support.

These fields may be updated using the TMSUPDTE utility as well as the ISPF and TIQ user interfaces. However, the fields are primarily updated by the user exits provided by CA or by the robotic system vendor. Exits are invoked when volumes are entered into or removed from the library, providing an up-to-the-minute status on volume residency.

The IBM Tape Library

The IBM TotalStorage Tape Library is IBM's main cartridge tape library for the mainframe. The IBM Tape Library also supports connectivity to open systems and is the framework for IBM's virtual tape system, the IBM TotalStorage Virtual Tape Server (VTS).

This section describes how to configure CA 1® Tape Management with the IBM Tape Library when managed by OAM and with host-attached physical devices. For more information about configuring CA 1® Tape Management with IBM VTS, see the section IBM Virtual Tape Server later in this chapter. The IBM Tape Library may also be configured in Basic Tape Library Support (BTLS) mode, in which case the OAM exits are not installed. For information on managing an IBM Tape Library in BTLS mode, see the section IBM Basic Tape Library Support (BTLS) later in this chapter.

Integration between CA 1 and the IBM Tape Library or the IBM VTS is based on four exits/options that are used to communicate with the robot:

- CBRUXENT—IBM OAM Volume Entry (Usermod CTSJUCBX)
- CBRUXEJC—IBM OAM Volume Eject (Usermod CTSJUCBX)
- CBRUXVNL—IBM OAM Volume Not In Library (Usermod CTSJUCBX)
- ROBSCR—CA 1 option (see member TMOOPTxx) to notify OAM when a tape volume is scratched.

Object Access Method (OAM) is a part of IBM's DFSMSdfp component and is designed to provide management of data objects rather than files. OAM is also used to manage IBM's tape libraries as part of an overall storage hierarchy, based on DFSMS System Managed Storage concepts and facilities. The OAM address space also maintains the Tape Configuration Database (TCDB), which defines the IBM tape volumes and libraries that are managed. The TCDB is made up of one or more VOLCATs, a special type of ICF.CATALOG which is specifically marked as a volume catalog.

OAM includes a component, Library Control Services (LCS) which is used to manage the movement and tracking of tapes in IBM automated tape libraries. CA 1 interfaces to OAM through requests made to LCS.

There are four basic functions available to support the IBM ATL with the current version of CA 1:

1. Ensure that the status of new cartridges entering the IBM ATL are correct
2. Synchronize the CA 1 TMC and the IBM ATL database (TCDB)
3. Eject any cartridges that are needed for off-site storage from the IBM ATL.
4. Change the status of cartridges as they are scratched.

These functions are accomplished in different ways depending on whether the IBM ATL is managed as an SMS managed tape device or using the much less common Basic Tape Library Support (BTLS).

To configure your installation to access the IBM ATL as an SMS managed tape library, you must customize the OAM exits CBRUXENT and CBRUXEJC. The CBRUXVNL exit is optional. The details of each process are described in the following section.

Note: Additional calls to these exits are performed when an IBM VTS is being managed. The additional calls related to virtual tape processing are discussed in the section IBM Virtual Tape Server (VTS) later in this chapter.

IBM CBRUXENT Volume Entry Exit

The OAM CBRUXENT exit is executed whenever a tape cartridge is inserted into the ATL. The CBRUXENT exit, which is provided by CA 1 determines the status of the tape within the TMC, then allows or disallows the entry of the tape and sets the status in the TCDB to scratch or private based on the status in the TMC. When installed, the CBRUXENT exit updates TMC fields ROBTY and ROBID to indicate the tape is in the ATL. By default, a foreign tape may not be entered into the ATL. If this is attempted, the tape is left in *insert* status. This exit must be customized to specify the SMS name of each robot in an internal table for both IBM ATL and IBM VTS implementations.

The CBRUXENT exit calls the module CTSQSTS, to get the status of the TMC volume being entered into the IBM ATL or VTS.

CTSQ returns the following codes:

0

Tape is in scratch status.

4

Tape is non-scratch and can be appended.

8

Tape is full (part of a multivolume chain) or owned by an EDM or CA Dynam/T.

12

Tape is marked BAD by CA-9/R+ or tape is in delete status and thus out of service.

16

Tape is not in the TMC (foreign tape).

20

There was an error trying to obtain the status of the tape.

IBM CBRUXEJC Volume Eject Exit

For basic IBM ATL processing, the OAM CBRUXEJC exit is invoked when a tape is physically ejected from the ATL. The sample CBRUXEJC exit that is provided by CA 1 clears the ROBTY and ROBID fields to indicate that the volume doesn't reside in any robot. For basic IBM ATL support, no changes to the CBRUXEJC sample exit are required.

If the eject operation fails (because the volume is in use as an example), OAM redrives the CBRUXEJC a second time. The second run indicates that the eject did not actually occur. CBRUXEJC updates the ROBTY field to either x'01' (physical tape) or x'81' (virtual tape). The volume remains in the robot. A subsequent attempt to eject the volume can be successful. The ROBTY field is cleared then.

IBM CBRUXVNL Volume Not in Library

The OAM CBRUXEJC exit is invoked when a requested tape is not found in the ATL. This provides you with an opportunity to insert the tape into the ATL and retry allocation without canceling the job. Use of this exit is strictly optional. If you select to use this exit, be aware that OAM also calls this exit for DASD that is offline and for all tapes (even round reels) not in an SMS managed IBM robot.

No updates to CA 1 or OAM are performed based on this exit. The only time to install and use this exit is to allow operations to insert a cartridge back into the IBM ATL and re-drive the allocation instead of simply allowing the job to have a JCL error or abend. If you have an IBM ATL with similar devices both inside and outside the robot (so any tape can be mounted either inside or outside the ATL) then there is no need to use this exit.

This exit must be customized for your environment. You must determine what processing needs to be done if the tape you are requesting is not in the library. For example, if you have 3490 drives outside the IBM ATL and have only 3590 drives inside the IBM ATL, un-comment out two lines that compare the CA 1 returned density with the value 3590. If you do not un-comment out these two lines then all allocations for the IBM ATL are driven through the CBRUXVNL exit.

CA 1 ROBSCR

Use the ROBSCR option in TMOOPTxx of *hlq*.CTAPOPTN to indicate that the robotic (virtual) tape system is notified when a tape volume has gone scratch. Which robotic or virtual tape system to notify is based on the TMROBTY field in the TMC.

If the ROBSCR option is set to YES, and the TMROBTY field in the TMC record for the scratched volume is set to x'01' or x'81', then OAM is called when that specific volume is scratched.

If ROBSCR is set to YES and CA Vtape or Oracle/STK are used, the CA Vtape or Oracle/STK library must be either link-listed in the system linklist or added as a STEPLIB in the TMSCLEAN proc (TMSCLEAN proc step).

CA 1 CTSSYNC Utility Support for the IBM ATL

Use the CTSSYNC utility to communicate with OAM using LCS services for both the basic ATL and the VTS. This is in the main interface module between CA 1® Tape Management and OAM. For basic ATL support, CTSSYNC is used to eject tapes and can also be used to synchronize the TCDB with the TMC. Additional functionality to support the VTS is described in the IBM VTS section below.

CTSSYNC can perform eject or synchronize functions on either a range of volumes or a list of volumes. The utility can accept input from both control statements and parameter values. CA 1® Tape Management provides CA Earl report member TMEEJECT of *hlq*.CTAPEARL to read the output of the vault system program TMSVMVLT and format eject statements for volumes selected to be moved to a vault.

IBM ATL Recommendations and Procedures

The following usermod found in *hlq*.CTAPJCL must be installed to provide support for the IBM ATL:

CTSJUCBX

Applies your customized source to CBRUXENT, CBRUXEJC (and optionally CBRUXVNL) to your z/OS linklib.

The following CA 1® Tape Management System Parameters and procedures should also be set:

- **ROBSCR=YES**—set in the TMOOPTxx member of CTAPOPTN to enable real-time scratch processing.
- **WRKFLS=NO**—set in the TMOOPTxx member of CTAPOPTN. This causes CA 1 to assign a one-day retention to all work files.
- If **YSVC=YES**, update your security system to give OAM update access to the resource YSVCUNCDC.

Run your normal daily CA 1 maintenance jobs. When TMSCLEAN runs, it notifies the ATL of the tapes that have been scratched. After your job for vault management, run TMEEJECT to build control statements for CTSSYNC to EJECT the cartridge from the ATL. As cartridges are returned from the vault, you can insert them back into the ATL.

Notes:

1. CA 1 must be active before other software products that use tape or look for the CA 1 mount messages. This includes, but is not limited to OAM for the IBM robotic devices.
2. CA 1 does not support scratch subpooling in the IBM OAM controlled ATL/VTS.
3. If you run the IBM ATL in a Peer-to-Peer configuration (with a second IBM ATL off-site mirroring your ATL on-site), the procedures are the same. Do not run the TMEEJECT job to create the eject list. In Peer-to-Peer configurations the elapsed run time of TMSCLEAN may be extended due to delays in communicating with the off-site ATL. If the elapsed time for TMSCLEAN becomes too long, you may optionally set SCRATCH to NO and run CTSSYNC with a parameter of SYNC to cause the status of recently scratched volumes to be synchronized with the TMC. This results in the scratch volumes being updated in the OAM TCDB to scratch status.

IBM Virtual Tape Server (VTS)

This section documents how to configure CA 1 to work with the IBM TotalStorage Virtual Tape Server (VTS). For information on the basic IBM ATL and a general introduction to CA 1 robotics support, see sections TMC Fields Used to Track Tapes in Automated Tape Libraries and IBM Tape Library.

As with the IBM ATL, integration between CA 1 and the IBM VTS is based on four exits/options that are used to communicate with the robot:

- **CBRUXENT**—IBM OAM Volume Entry
- **CBRUXEJC**—IBM OAM Volume Eject
- **CBRUXVNL**—IBM OAM Volume Not In Library
- **ROBSCR**—CA 1 option (see member TMOOPTxx) to notify OAM when a tape volume is scratched

IBM VTS - CBRUXENT Volume Entry Exit

The OAM CBRUXENT exit is executed whenever a tape cartridge is inserted into the ATL, or when a virtual volume is imported from a physical tape or defined to the VTS. The CBRUXENT exit provided by CA 1 determines the status of the tape within the TMC. CBRUXENT then allows or disallows the entry of the tape and sets the status in the TCDB to scratch or private based on the status in the TMC. When installed, the CBRUXENT exit provided with CA 1 updates TMC fields ROBTY and ROBID to indicate the tape is in the ATL. By default, a foreign tape may not be entered into the ATL. If this is attempted, the tape is left in insert status. This exit must be customized to specify the SMS name of each robot in an internal table for both ATL and VTS implementations.

Note: For more information on CA 1 provided CBRUXENT exits, see IBM ATL CBRUXENT Volume Entry Exit

IBM VTS - CBRUXEJC Volume Eject Exit

The CBRUXEJC exit is invoked when a physical tape is ejected and when a virtual tape is exported. Modify this exit if you are using the VTS with the import and export functions. A pseudo DSN needs to be added to the code so that CA 1 can track the physical tapes used during the export. If you are using a VTS without the import/export feature, or if you are using an ATL, modification to the CA 1 supplied version is not necessary.

During an export function, the following actions are taken on the virtual volumes that are being exported:

- CA 1 clears the OUTCODE, OUTDATE, SLOT and ROBID fields
- ACTVOL is updated with the physical VOLSER that the virtual volumes are being exported to
- FLAG5 is set to 20 (VTX - Virtual Tape Exported)

The following actions are taken on the physical volume (also referred to as the container volume):

- The physical tape is taken out of scratch or delete status
- The EXPDT is set to PERM
- The DSN is set to the pseudo data set name that you have specified in exit CBRUXEJC
- FLAG3 is set to 20 (EDM - External Data Manager)
- Creation date (CDATE) is set to today's date
- The OUTCODE and SLOT in the physical tape are set to match the vaulting data from the first virtual volume that was exported to this physical volume
- OUTDATE is set to the current date

The pseudo DSN that is assigned to the physical volume should be added to the Vault Pattern Data Set to define the vault movement that is desired for exported VTS volumes. The vault system utility TMSVMEDT has been changed to skip all volume records that have the VTX (volume exported bit x'20') on in FLAG5.

IBM VTS - CBRUXVNL Volume Not in Library

The OAM CBRUXVNL exit is invoked when a requested tape is not found in the ATL, providing an opportunity to insert the tape into the ATL and retry allocation without canceling the job. Use of this exit is strictly optional. If you select to use this exit, you should be aware that OAM also calls this exit for DASD that is offline and for all tapes (even round reels) not in an SMS managed IBM robot.

There are no CA 1 OAM updates performed based on this exit.

This exit is only installed and used to allow operations an opportunity to insert a cartridge (or import a virtual volume) back into the ATL/VTS and re-drive the allocation rather than simply allowing a JCL error or abend to occur during the job. If you have a VTS without the IMPORT/EXPORT feature this exit is not necessary (all virtual volumes are always inside the robot.)

This exit must be customized for your environment. You must also determine what processing needs to be done if the tape you are requesting is not in the library. For example, if you have 3490 drives outside the ATL and have only 3590 drives inside the ATL, un-comment out two lines that compare the CA 1 returned density with the value "3590". If you do not un-comment out these two lines, then ALL allocations for the ATL are driven through the CBRUXVNL exit.

CA 1 ROBSCR Support for the IBM VTS

Use the ROBSCR option in TMOOPTxx of *hlq*.CTAPOPTN to indicate that the robotic (virtual) tape system should be notified when a tape volume has gone scratch. Which robotic or virtual tape system to notify is based on the TMROBTY field in the TMC.

If the ROBSCR option is set to YES, and the TMROBTY field in the TMC record for the scratched volume is set to x'01' or x'81', then OAM is called when that specific volume is scratched.

If ROBSCR is set to YES and CA Vtape or Oracle/STK are used, the CA Vtape or Oracle/STK library must be either link-listed in the system linklist or added as a STEPLIB in the TMSCLEAN proc (TMSCLEAN proc step).

CA 1 CTSSYNC Utility Support for the IBM VTS

Use the CTSSYNC utility to communicate with OAM using LCS services for the basic ATL and for the VTS. It is the main interface module between CA 1 and OAM. CTSSYNC is used to eject tapes or to perform import/export functions for a VTS. It can also be used to synchronize the TCDB with the TMC.

When using CTSSYNC to import or export tapes from the VTS one of the following additional DD statements is required:

- Export operations require you to specify an EXPORT DD
- Import operations require you to specify an IMPORT DD

These DD statements must point to a unit in the VTS. The list of virtual volumes to be imported or exported is passed to the VTS by temporarily using a virtual volume in scratch status that is allocated with the EXPORT or IMPORT DD statement. For this reason, the virtual volume written on must reside in the same VTS that the volumes are being exported from or imported to. Only one IMPORT or EXPORT can be run at a time.

CTSSYNC can perform eject, export, import or synchronize functions on either a range of volsers or a list of volsers. CA 1 provides the following CA Earl report members in *hlq*.CTAPECPB to create control statements that can be processed by CTSSYNC:

TMEJECT

Creates export or eject control statements from the output of the vault management system.

TMEIMPRT

Creates import control statements for a specified physical volser you specify. If some of the virtual volumes are in scratch status, CA 1 builds the control statements so that the actual data on the virtual volume is not copied to cache.

Two following CA Earl report members list VTS physical volumes and assist in creating TMSUPDTE transactions to manage scratching the physical volume when all virtual volumes have expired:

TMEACTVL

Lists the physical tapes that have been used for export processing and the virtual tapes that have been put on each one. If the physical tape is no longer needed, (that is, if all virtual volumes have been imported or all virtual volumes have expired,) it produces TMSUPDTE control statements to help automate the scratching of the physical tape. Once the tape has been scratched you manually insert the physical tape into the library as an eligible stacking volume.

Note: For information, see IBM's *IBM VTS User Guide*.

TMEVTINV

Lists the physical tapes and the associated exported virtual volumes.

Recommendations and Procedures for IBM VTS Without IMPORT/EXPORT

The following usermod must be installed to provide support for the IBM VTS without the IMPORT/EXPORT feature:

- CTSJUCBX—Applies your customized source to CBRUXENT, CBRUXEJC (and optionally CBRUXVNL) to your z/OS linklib

The following CA 1 system parameters and procedures should also be set:

- ROBSCR=YES—Set in TMOOPTxx member of CTAPOPTN to enable real-time scratch processing.
- WRKFLS=NO—Set in TMOOPTxx member of CTAPOPTN. This causes CA 1 to assign a 1 day retention to all work files.
- If YSVC=YES you must update your security system to give OAM update access to resource YSVCUNCD.

Run your normal daily CA 1 maintenance jobs and when TMSCLEAN runs, it notifies the VTS of which tapes have been scratched. All tapes (virtual volumes) remain within the VTS.

Recommendations and Procedures for IBM VTS with IMPORT/EXPORT

The following usermod must be installed to provide support for the IBM VTS in configurations where the IMPORT/EXPORT feature is used:

- CTSJUCBX—Applies your customized source to CBRUXENT, CBRUXEJC (and optionally CBRUXVNL) to your z/OS linklib.

The following CA 1 system parameters and procedures should also be set:

- ROBSCR=YES—Set in TMOOPTxx member of CTAPOPTN
- WRKFLS=NO—Set in TMOOPTxx member of CTAPOPTN. This causes CA 1 to assign a 1 day retention to all work files.
- If YSVC=YES you must update your security system to give OAM update access to resource YSVCUNCD.

If you are vaulting data sets that are on virtual tapes and they go to different vaults, some changes to CBRUXEJC and TMEEJECT need to be made. In CBRUXEJC, add logic to compare the DSN (and optionally other data fields) and set different pseudo DSNs for the physical tapes to be created. In TMEEJECT add a destination on the EXPORT control statements for CTSSYNC depending on the pseudo DSN that was set:

```
EXPORT , vvvvvv , VLT1
```

```
EXPORT , vvvvvv , VLT2
```

The vvvvvv is replaced with the VOLSER to be exported and VLT1 and VLT2 are examples of separate vaults you want physical volumes exported to. These names can be the vault name or any other meaningful destination for your site.

When CTSSYNC is run with these control statements, all tapes with VLT1 are exported together and all tapes with VLT2 are exported onto another tape. This gives you the separation so that you can have exported tapes going to different vaults. If you are going to keep the physical tape off-site until it expires you do not need to add a vault pattern for the pseudo DSNs. To keep them a number of days, have a new vault pattern with the pseudo DSN.

Use the following steps for Vaulting when using the IMPORT/EXPORT feature:

1. Run TMEACTVL to build control statements for TMSUPDTE to expire physical tapes when all virtual volumes stacked upon them have been imported or scratched.
2. Run TMSUPDTE with the control statements from TMEACTVL. The volumes are changed to "scratch" status the next time TMSCLEAN runs.
3. Run your daily CA 1 job for vaulting (TMSVMEDT, TMSVMVLT, and TMSVMUPD).
4. Run TMEEJECT to build the CTSSYNC control statements to eject the physical tapes in an ATL and export the virtual tapes.
5. Run CTSSYNC to do the ejects or exports.
6. Run TMEIMPRT to build the CTSSYNC control statements to do the IMPORT for the tapes returned from the vault, after they have been received back in the library and are ready to be put back into the VTS
7. Run CTSSYNC using the control statements from TMEIMPRT.

Exported Virtual Volume Considerations

Virtual volumes that have been exported or externalized to a physical volume are now processed by the vaulting system. Once the VTX bit x'20' FLAG5 has been turned on in the TMC record, virtual tapes continue to be moved according to the VPD rule, but slot assignment is bypassed. The slot number is set to a dummy value of 9,999,999 (a "virtual slot"). This virtual slot number of 9,999,999 will be displayed on any ISPF display or the TI display and the Vantage/GUI display.

These exported virtual volumes do not appear on picking or distribution list (there is no physical cartridge to move). They are displayed on the inventory list with the physical container volume (ACTVOL) in place of the slot number preceded by an asterisk (*). So, on the inventory list the slot number might be displayed as *C12345 when C12345 is the container volume that the virtual volume has been exported onto.

IBM Basic Tape Library Support (BTLS)

With the IBM ATL configured in BTLS mode, most functions are performed in batch using IDCAMS with the LIBRARY command. In addition, the CTSBTLS program is linked as IDCLI04, the name of the Set Category Installation Exit.

To ensure that the status is correct on new cartridges being entered into a IBM BTLS ATL, a list of all tapes currently in the INSERT category are processed through the Set Category Installation Exit. The exit determines their current status in the CA 1 TMC, and then sets the cartridges to the correct category (PRIVATE, SCRTCH, or ERROR).

To allow the synchronization between CA 1 and the BTLS database, an inventory of all cartridges currently in the ATL is created. This list is passed through the same exit, which again determines their current status in the tape management system, and then sets the cartridges to the correct category.

To eject tapes and to set tapes to a category of SCRTCH after they have been scratched by the Tape Management System, lists of volumes are used as input to an IDCAMS job. The IDCAMS job then sets the category for all tapes in the list to a specified value (for example XEJECT or SCRTCH for example.)

IBM BTLS Insert Tapes

After cartridges have been inserted into a BTLS, they are set to a category of INSERT. A list of all tapes currently in the INSERT category can be obtained in machine-readable form (LIBOUT DD) through the INVENTORY command. This list can then be read as input (CTSINPUT DD) by the new IDCLI04 exit and have the category changed to either PRIVATE, SCRTCH, or ERROR.

The following is an example:

```
//STEP1      EXEC  PGM=IDCAMS
//SYSPRINT DD   SYSOUT=A
//LIBOUT     DD   DSN=&. &TEMP. , DISP=(MOD,PASS) ,
                  DCB=(RECFM=FB,LRECL=80) ,
//           SPACE=(CYL,(1,1))
//CTSINPUT DD   DSN=*.LIBOUT,DISP=(OLD,PASS)
//SYSIN      DD   *
                LIBRARY INVENTORY UNIT(ALL) CATEGORY(INSERT)
                LIBRARY SETCEXIT UNIT(ALL)
/*
```

IBM BTLS - Synchronize the IBM ATL and TMC

On an occasional basis (once a week or once a month) it is a good idea to synchronize the BTLS and the Tape Management Catalog. This can be done by creating an inventory (LIBOUT DD) of all tapes in the BTLS. This list can then be read as input (CTSINPUT DD) by the new IDC104 exit and have the category changed to either PRIVATE, SCRTCH or ERROR.

The following is an example:

```
//STEP1      EXEC  PGM=IDCAMS
//SYSPRINT DD   SYSOUT=A
//LIBOUT     DD   DSN=&. &TEMP . , DISP=(MOD,PASS) ,
                  DCB=(RECFM=FB,LRECL=80) ,
//           SPACE=(CYL,(1,1))
//CTSINPUT DD   DSN=*.LIBOUT,DISP=(OLD,PASS)
//SYSIN      DD   *
    LIBRARY INVENTORY UNIT(ALL) CATEGORY(INSET)
    LIBRARY INVENTORY UNIT(ALL) CATEGORY(SCRTCH)
    LIBRARY INVENTORY UNIT(ALL) CATEGORY(PRIVATE)
    LIBRARY SETCEXIT UNIT(ALL)
/*
```

IBM BTLS - Eject Cartridges

Once a list of volumes to be sent off-site has been created, it must be "fed" as input to IDCAMS. The category may be either XEJECT (for the convenience station) or XEJECTB (for the bulk station).

The following is an example:

```
//STEP1      EXEC  PGM=IDCAMS
//SYSPRINT DD   SYSOUT=A
//LIBIN      DD   DSN=VOLUME.LIST.TO.BE.EJECTED,DISP=OLD
//SYSIN      DD   *
    LIBRARY SETCAT UNIT(123) CATEGORY(XEJECT)
/*
```

Note: The LIBIN file must be a Fixed Block file, any record length, with the VOLSER(s) starting in column 1. You can use CA Earl to generate this list.

IBM BTLS - Scratch Cartridges

Once the daily scratch job has been run (TMSCLEAN), the output from that job can be used as input to IDCAMS. Like the EJECT process, the list is used as input (using the LIBIN DD statement) and the category is changed to SCRTCH.

```
//STEP1    EXEC  PGM=IDCAMS
//SYSPRINT DD   SYSOUT=A
//LIBIN     DD   DSN=DAILY.SCRATCH.LIST,DISP=OLD
//SYSIN     DD   *
      LIBRARY SETCAT UNIT(123) CATEGORY(SCRTCH)
/*
```

Note: The LIBIN file must be a Fixed Block file, any record length, with the VOLSER(s) starting in column 1. You can use CA Earl to generate this list.

Share an IBM ATL or VTS

The IBM ATL and VTS can be attached to multiple systems in two main configurations. The first is to partition the box – make it look like two separate boxes – each section attached to a different system or share the ATL or VTS with different systems. In fact, you can partition the box and have one partition shared between some systems and have the second partition shared between different systems.

In most cases, when the ATL or VTS is going to be shared between different operating systems, the box is partitioned. One partition is used by z/OS (most likely shared by a couple of z/OS systems) and one partition is used by the other operating system (z/VM for example). In a partition mode, the ATL/VTS operates as if it were two separate boxes and tape volumes cannot be shared. So tapes created in one partition cannot be used in the other partition.

For more information about sharing or partitioning an ATL or VTS, it is strongly recommended that you review the DFSMS Object Access Method Planning, Installation, and Storage Administration Guide for Tape Libraries.

When sharing an ATL or VTS between z/OS systems there are four different ways to configure your CA 1 and OAM environment:

- Common CA 1 TMC and common OAM TCDB (Tape Configuration Database, also known as the VOLCAT)
- Separate CA 1 TMCs and separate OAM TCDBs
- Common CA 1 TMC and separate OAM TCDBs
- Separate CA 1 TMCs and a common OAM TCDB

Each of these presents slightly different requirements and can meet different needs. The section following provide some items within the CA 1 supplied exits that may need customization depending on your environment.

Common CA 1 TMC and Common OAM TCDB

This is probably the most common way to share an ATL or VTS. All z/OS systems share a common TMC and a common TCDB. In this environment, very few modifications to the CA 1 supplied entry and eject exits are needed. To enter foreign tapes into the robot (cartridges not defined to CA 1,) change the entry (CBRUXENT) exit. In the CA 1-supplied exit (CTSUXENT in the *hlq*.CTAPSAMP library) there is a small branch table after label ASKCA, where there is a branch to label RC0, RC4, RC8, RC12, RC16 and RC20 made. This can be done using the usermod CTSJUCBX.

To read foreign tapes inserted into the Tape Library, the easiest modification is to change the branch table. So, in the case of a foreign tape the branch is made to RC8 instead of to RC16. Do this by simply changing "B RC16" to "B RC8" for foreign tapes. If you are not reading foreign tapes inside the ATL (for example, if the VTS has no native-attached physical devices) then there is no need to modify the entry or eject exits supplied with CA 1.

Separate CA 1 TMCs and Separate OAM TCDBs

This is another very common way to share an ATL or VTS when the Tape Library is not actually partitioned. This is almost the same as having the Tape Library partitioned, but provides the user with more flexibility in terms of adjusting the size (number of volumes or number of devices) on each side without having to redefine the partitions. For basic operation, the CA 1® Tape Management supplied entry and eject exits do not require modifications. However, if foreign tapes are inserted into the ATL and used on one of the systems, a more complicated version of the entry exit is needed.

For an example, we assume that CPUA and CPUB share a Tape Library with separate TMCs and TCDBs, and we want to allow foreign tapes to be used as input on CPUA. In this case, the entry and eject exits for CPUB remain unchanged. CPUA cartridges are considered foreign to CPUB, when CPUB is notified that a CPUA cartridge is inserted. The CPUA cartridge is ignored. When CPUA is notified it assigns the correct status (SCRATCH or PRIVATE) and takes ownership of the cartridge.

However, in the same example the entry exit on CPUA needs to be modified and is more complicated. In the CA 1–supplied entry exit (CTSUXENT in the *hlq*.CTAPSAMP library), a code needs to be added at label RC16. This can be done using the usermod CTSJUCBX.

If the volume inserted is defined to CPUB (which means a table needs to be created or a set of compares needs to be done to see if the volser matches those defined to CPUB,) then the instruction LA R15,UxEIGNOR needs to be performed (followed by the branch to EXIT). However, if the tape is not defined to CPUB (again, based on the volser) branch to label RC8 (which puts the foreign tape into PRIVATE status and take ownership of it on CPUA.) Depending upon the volser standards on CPUB, this can be very simple (look for a cartridge starting with B0 or B1) or very complicated.

Common CA 1 TMC and Separate OAM TCDBs

This method is not recommended, however it can be done. This provides the user with the ability to logically partition the Tape Library (with separate TCDBs) but still have a common CA 1 TMC. This creates a problem because the entry exit (CTSUXENT in *hlq*.CTAPSAMP library) needs to be modified on both systems, and the ROBSCR scratch option should NOT be enabled (ROBSCR=NO). The entry exit on CPUTA needs to have all the CPUB cartridges defined to it, and not take ownership of them (even though they are defined to the TMC.) This additional code is best to be performed prior to the "ASKCA" label because all of the cartridges are known to both systems (because they share a common TMC). So, on CPUTA before the ASKCA label, you must compare the volser and see if it is owned by CPUB. If it is, branch to label RC16 (leave it in insert status for CPUB to take ownership). This also applies to CPUB, the exit must look for CPUTA cartridges and branch to label RC16 for them (so CPUTA takes ownership of its cartridges.) If you want to share a common exit on both systems, the exit needs to be modified to look at the SMFID or SYSTEM-NAME and determine if the volser should be owned or not based on the combination of which system it is running on, and which tape is being inserted.

It is best to disable the real-time scratch interface by setting ROBSCR=NO in the TMOOPTxx member. Instead, break the scratch list created by TMSCLEAN and into two separate lists. One list consists of CPUTA owned scratch tapes and the other is a list of CPUB owned scratch tapes. Then, on each system run CTSSYNC taking in the correct list of scratch tapes (CTSSYNC can be run with PARM=SYNC and the sysin is simply a list of cartridges.)

Separate CA 1 TMCs and Common OAM TCDB

Important! This method is not recommended and should be avoided if at all possible.

CPUTA and CPUB have a common OAM TCDB. Any tape can be mounted to satisfy a scratch request on either system. By having different categories defined on each system (this is done by modifying SYS1.PARMLIB members as explained in the OAM PISA guide), the problem of shared scratch pools can be removed. However, it is still possible to request a specific-volume from either system for any volume. CA 1 only knows about its tapes and not about other system tapes. Other system tapes appear as FOREIGN. Using EXPDT=98000 you can read (or update) active tapes from the other system. In this case, you must tightly restrict the use of EXPDT=98000 processing. If you don't, anyone that has access to EXPDT=98000 processing can request tapes from the other system and read, modify or both, the existing data. The full 44-character data set name checking is not available. External security rules are not sufficient.

The entry and eject exits can be used as shipped with CA 1 without modification and the real-time scratch processing performed with the ROBSCR option can be run on both systems.

CA Vtape Virtual Tape System

CA Vtape Virtual Tape System is a CA software-based virtual tape solution that enables you to improve the efficiency of your tape storage environment. CA Vtape Virtual Tape System uses your existing tape and DASD hardware to implement virtual tape with mainframe-class reliability and performance.

The CA Vtape Virtual Tape System interface to CA 1 is internal and does not require user exits for use. A CA Vtape Virtual Tape System parameter defined in the VTPARMS member of PARMLIB is used to activate this internal interface. When the Tape Management System Parameter is set to CA1 or AUTO (and CA 1 is discovered on the system), CA Vtape Virtual Tape System activates this internal interface to provide the following benefits:

- The expiration date for the virtual volumes is retrieved directly from CA 1 and used in backstore processing to group together virtual volumes with similar expiration attributes.
- During externalization processing, the container field (ACTVOL) in the virtual volume is updated in the TMC to reflect the physical volume it is written to.

In most cases, CA Vtape Virtual Tape System virtual volumes are handled as real volumes. CA 1 remains in control of scratch processing, which must be coordinated with CA Vtape Virtual Tape System. CA Vtape Virtual Tape System provides a job in *hlq*.INSTALL(CA1) to perform scratch synchronization. This job must be scheduled to run after the daily TMSCLEAN run.

If ROBSCR is set to YES and CA Vtape is used, the CA-Vtape library must be either link-listed in the system linklist or added as a STEPLIB in the TMSCLEAN proc (TMSCLEAN proc step).

The CA Vtape Virtual Tape System CA 1 Interface

Be aware of the following when setting up an interface between CA Vtape Virtual Tape System and CA 1:

- You must define an exclusive range of virtual volumes in the TMC to be used by CA Vtape Virtual Tape System.
- CA Vtape Virtual Tape System writes virtual volumes to the physical container volume during externalization processing as secondary files. A DSNB record is used for each virtual volume written. Extend the TMC to define.
- DSNBs are not freed until all virtual volumes on the physical tape are expired.
- The CA Vtape Virtual Tape System RECYCLE utility can be run to recover fragmented physical tape space by copying active virtual volumes to new backstore tapes. This process also frees up DSNBs.
- CA 1 can report on the physical volumes externalized and the virtual volume to physical volume mapping. CA 1 CTAPECPB member TMEVTINV can be executed to create this report. This information is important in disaster recovery planning.

Note: For more information about the interface between CA Vtape Virtual Tape System and CA 1, see the *CA Vtape Virtual Tape System User Guide*.

Sun/STK Tape Libraries

CA 1 supports all major Sun/STK tape libraries for mainframe environments. This includes but is not limited to the following:

- PowderHorn
- TimberWolf 9740
- SL8500

CA 1 integrates with STK libraries through the STK Host Software Component (HSC) Exit 6 (SLSUX06). HSC Exit 6 is documented in the Sun/STK HSC System Programmer's Guide. A sample SLSUX06 exit is provided in CAI.CTAPJCL for customers who do not have an existing EXIT 6 routine and only need support for updating the ROBTY and ROBID. This version of the exit provides basic code to update the ROBTY and ROBID when volumes are entered into the robot (define processing) and to clear ROBTY and ROBID when the volumes are ejected (delete processing).

Once the ROBTY field is updated automatically when a cartridge is entered into an Oracle/STK library, the new ROBSCR option in CTAPOPTN can be used to notify HSC when a tape is put into scratch status.

If ROBSCR is set to YES and Oracle/STK is used, the Oracle/STK library must be either link-listed in the system linklist or added as a STEPLIB in the TMSCLEAN proc (TMSCLEAN proc step).

Oracle/STK Virtual Storage Manager (VSM)

The Sun/STK Virtual Storage Manager (VSM) is integrated with CA 1® Tape Management through exits provided by STK. VSM maintains all consistent information in the HSC control data set (CDS,) which must be synchronized with the TMC. This integration is documented in the STK documentation set.

Note: For more information about configuring the VSM, see the *Sun/STK HSC System Programmer's Guide* and *Sun/STK HSC Installation Guide for MVS*.

Chapter 10: Common Tape System

This section contains the following topics:

[About the Common Tape System](#) (see page 187)

[CTS Initialization](#) (see page 188)

[CTS Started Task](#) (see page 189)

[CTS Commands](#) (see page 190)

[CA 1 Automatic Pointers Error Correction \(APEC\)](#) (see page 191)

[CA 1 Data Base Services \(DBS\)](#) (see page 198)

[CA 1 Health Checker Service \(HCK\)](#) (see page 200)

[Online Label Interface \(LAB\)](#) (see page 202)

[Distributed Tape Support \(DTS\)](#) (see page 233)

[CTS Scheduler \(SCHD\)](#) (see page 241)

[CA 1 SMF Queue \(SMFQ\)](#) (see page 249)

[Tape Map \(TMAP\)](#) (see page 252)

[Volume Pool Monitor \(VPM\)](#) (see page 254)

About the Common Tape System

The Common Tape System (CTS) provides a set of interfaces to all CA tape management products. CTS consists of a started task, utilities, and interface programs. CTS is started by commands provided in the member CTSSTART in *hlq.CTAPOPTN*. In the ENFCMDS member of CA Common Services *hlq.PPOPTION* which CAIENF reads, specify 'S CTS' to have CAIENF start CTS. In the CTS startup procedure, the CTSSTART member in *hlq.CTAPOPTN* is read and the CTS system starts.

The VPM subtask is unique in that it obtains options from the VSAM Database (VDB). If you use this subtask, the VPM dynamically allocates the VDB.

CTS Initialization

The following example shows the commands used to start selected CTS tasks:

```
/* * * * * * * * * * * * * * * * * * * * * * * * * * * */
/*          CA-COMMON TAPE SYSTEM STARTUP COMMANDS          */
/*          CA 1  STARTUP PARAMETERS                          */
/* * * * * * * * * * * * * * * * * * * * * * * * * * * */
SET    LOG                /* MAINTAIN A LOG OF MESSAGES      */
SET NOSNAP                /* DO NOT LOG DIAGNOSTICS         */
SET NOWTOR                /* NO OUTSTANDING WTOR           */
START DBS                 /* AUTO START DATABASE SERVICE TASK */
START LAB                 /* AUTO LABEL PRINT TASK          */
START APEC                /* AUTO POINTERS ERROR CORRECTION */
```

SET LOG

Indicates that CTS retains a message log.

SET NOSNAP

Indicates that the option for recording snap dumps is turned off.

SET NOWTOR

Indicates that the optional outstanding CA\$F000R WTOR is not issued. Commands are entered through the console MODIFY command instead.

START DBS

Indicates that the Tape management Database Services (DBS) subtask is attached.

START LAB

Indicates that the Tape management Online Label Interface (LAB) is attached.

START APEC

Indicates that the Automatic Pointers Error Correction subtask is attached.

CTS Started Task

```

/**-----**
/**
/**          CA COMMON TAPE SYSTEM
/**          'STARTED TASK' PROCEDURE
/**
/**-----**
//CTS      PROC CTAPECPB='CAI.CTAPECPB',    CA COMMON SOURCE LIBRARY
//          LAB1='A',                        LABEL PTR#1 CLASS
//          LAB2='A',                        LABEL PTR#2 CLASS
//          LAB3='A',                        LABEL PTR#3 CLASS
//          LBL='TMELBLS',                    LABEL FORMAT MEMBER
//          LINKLIB='CAI.CAILIB',            CA COMMON LOAD LIBRARY
//          OPTS='hlq.CTAPOPTN',            CA COMMON OPTION LIBRARY
//          PRM=,                             EXEC TIME COMMANDS
//          SOUT='*',                         OUTPUT SYSOUT CLASS
//          SPIN='A',                         SPINOFF OUTPUT SYSOUT CLASS
//          START='CTSSTART',                CTS STARTUP OPTIONS
//          WRKUNIT='SYSDA'                  DASD WORK UNIT TYPE
//CTSGO     EXEC PGM=CTSMAN,
//          PARM='&PRM'.,
//          REGION=3M,
//          TIME=1439
//SPINOFF   OUTPUT CLASS=&SPIN.              REQUIRED STATEMENT
//OUTPUTP1  OUTPUT CLASS=&LAB1.
//OUTPUTP2  OUTPUT CLASS=&LAB2.
//OUTPUTP3  OUTPUT CLASS=&LAB3.
//STEPLIB   DD DSN=&LINKLIB.,
//          DISP=SHR
//SYSPRINT  DD SYSOUT=&SOUT.
//CTSSYSIN  DD DSN=&OPTS(.&START).,
//          DISP=SHR
//SYSOUT     DD SYSOUT=&SOUT.
//EARLOBJ   DD UNIT=&WRKUNIT.,
//          SPACE=(TRK,2)
//EARLLIB   DD DSN=&CTAPECPB.,
//          DISP=SHR
//SORTIN    DD DUMMY
//SORTOUT   DD DUMMY
//SYSIN     DD DSN=&CTAPECPB(&LBL).,
//          DISP=SHR

```

CTS may be tested by running it in batch mode. If run in batch mode, ENF and CCI which are used by the LAB subtask do not need to be active.

CTS Commands

The CTS address space is created by starting the CTS task. The CTS task is a multitasking supervisor, which provides support for CTS Services. The CTS task provides the means for starting, stopping, and controlling CTS subtasks. The CTS task also provides display and diagnostic facilities. Except where otherwise indicated, the following command-processing rules are documented:

- The CTS task processes commands from Console Modify, CAI.CTAPOPTN member CTSSSTART, EXEC PARM=, CTS internal message facility, and reply to an outstanding WTOR.
- Commands from any source are free-form in MACRO type and/or TSO type syntax.
- The syntax is a command-word followed by parameters and separated by either blanks or commas. For example, both of the following commands are valid:


```
F CTSV,MSG,VPM, 'DISPLAY P00LS'
```

```
F CTSV,MSG VPM 'DISPLAY P00LS'
```
- The first comma after CTSV is required by MVS console command processing.
- TSO style comments (/*.....*/) can appear anywhere in the command line or alone on the command line.
- Parameters can be keyword, positional, or value and they can be specified in any order.
- Keyword parameters can be of either the KWD= or KWD() style.
- Parameters which contain blanks, commas, or special characters must be enclosed in '.....' or (.....).

CA 1 Automatic Pointers Error Correction (APEC)

Use the Automatic Pointers Error Correction (APEC) subtask of the Common Tape System (CTS) to inspect and repair volume and data set chaining errors found in the CA 1 TMC in real time. Use the APEC subtask (program TMSAPEC) and TMSPTRS batch utility to identify invalid internal CA 1 multivolume and multi-data set chains and errors that exist in the free chain of unused DSNBs. In most cases invalid or broken chains may occur as a result of invalid manual updates, however a system failure during CA 1 processing may also cause invalid chains, or if the TMC could not be restored with all available Audit data (as in a disaster recovery). Volumes in delete or scratch status are not processed for pointer errors.

Note: For more information about volume and data set chaining structures and possible errors, see the *Utilities and Reports Reference Guide*.

Unlike TMSPTRS, which scans for pointer errors, generates reports, and control statements, the TMSAPEC program can identify additional problems and is able to correct pointer errors directly in the TMC. APEC can also be run in a "NOUPDATE" mode, causing it to only report on errors found and to generate control statements for the TMSUPDTE, TMSUDSNB, and TMSAGGR utilities. In addition to the errors identified by TMSPTRS, APEC also verifies that all of the DSNBs on the free chain are truly free.

All messages created by TMSAPEC are in the same format as those created by TMSPTRS. Some messages may indicate a range of volumes or DSNBs, while messages from TMSAPEC are for a single volume or DSNB. Because APEC is updating the TMC in real time, the errors identified by APEC may differ from TMSPTRS. The following errors are fixed by APEC: ERR24, ERR37, ERR41, ERR42, ERR43, ERR44, ERR45, ERR51, ERR55, ERR56, ERR78, ERR79, and ERR67. When APEC is run with the NOUPDATE option, one pointer error can cause the generation several messages. When run with the UPDATE option ON, the same pointer error may only generate a single error message.

APEC is designed to run as a background task that allows users to control how frequently and how long the task is to be run. Use control statements to prevent the task from running during specified periods of high activity and pacing logic is provided to prevent over-utilization of CPU resources.

APEC in a Shared TMC Environment

Logic is provided to prevent APEC from running on any system in the shared TMC environment while any of the following utilities are in use:

- TMSPTRS
- TMSCLEAN
- TMSSPLIT
- TMSMERGE

If any of these utilities start while APEC is scanning or if any of these utilities are running when a scan starts, APEC stops scanning until the next scheduled start time.

If the TMC has to be moved, before you start TMSINIT to establish the new location, stop TMSAPEC before running TMSINIT. If TMSAPEC is active when TMSINIT is run, unpredictable results may be obtained, for example an S130 abend or errors reading from or writing to the TMC.

The APEC subtask identifies users or jobs that have incorrectly updated volume or data set chains. Messages issued when invalid chains are discovered include the userid of the last updater of the invalid TMC record. Check the output of APEC task at regular intervals to identify the users incorrectly performing updates to the TMC.

If you share the TMC with multiple systems, APEC must be run on one system at a time to avoid possible duplicate updates.

TMSAPEC issues an enqueue for exclusive ownership of a resource, SCOPE=SYSTEMS, QNAME=TMSQNAME, RNAME=TMSAPEC or RNAME=TMSAPEC-*vvvvv-data.set.name.tmc*. This resource should be GLOBALLY managed and shared by all systems sharing the TMC using either MIM or GRS. The RNAME depends on if you enabled the Unique_RNAME feature either via TMSXTEND or TMSUDSNB.

Startup Procedure

APEC can be started manually or automatically when CTS is started. The recommended method is to have CTS automatically start the APEC subtask.

Automatic Startup

To automatically start the APEC subtask each time CTS is started, add the following command to the CTSSTART member in *hlq.CTAPOPTN*:

```
SET TASK(APEC) PGM(TMSAPEC) PARM(yyyyyy)
START APEC
```

yyyyyy

Specifies the initialization parameters described in APEC Initialization Parameters.

Manual Startup

Once the CTS task has been started, you can communicate with CTS by using a MODIFY command or by WTOR. The following is an example of how to manually start the APEC subtask under CTS by using the MODIFY command:

```
F CTS,SET TASK(APEC) PGM(TMSAPEC) PARM(yyyyyy)
F CTS,SET START APEC
```

where 'yyyyyy' is replaced with initialization parameters described in APEC Initialization Parameters.

APEC Initialization Parameters

The following parameters are supported for the APEC subtask:

```
DELAY=shh
STIME=hhmm
ETIME=hhmm
TRUN=t
UPDATE | NOUPDATE
```

DELAY=*shh*

Specifies the amount of time in seconds and hundredths of a second, to wait between scanning individual volume or DSNB chains. This parameter is used to control the speed of a scan. When more time is specified, the scan runs slower and has a lower CPU use while scanning.

Default: 002

Example: If a 1/10 of a second delay is requested, use DELAY=010.

STIME=*hhmm*

Sets a time of day when a scan is automatically started. The start time is specified as local time (not GMT), in hours and minutes and must be less than 2400.

Default: 9:00am (0900)

Example: If a scan is to start at 11:30 pm every day, use STIME=2330

ETIME=hhmm

Sets a time of day when automatic scanning is to stop and can only be set if the STIME parameter is used to set a start time. The end time is specified as local time (not GMT), in hours and minutes and must be less than 2400. If a scan has not completed before the end time is reached the scan is halted.

Default: 5:00pm (1700)

Example: To stop automatic scanning at 8:45am, use ETIME=0845.

TRUN=t

Used to set the number of automatic scans that are to take place between the start time and end time. TRUN may only be specified if both STIME and ETIME have been set. Valid TRUN values are between 1 and 9. The number of scans requested will be distributed across the time period specified in the STIME and ETIME parameters. The TRUN parameter is recommended to prevent running this subtask more than necessary. In most installations, two passes of the TMC (TRUN=2) is recommended.

Default: 1

Example If 4 scans are to be run between the start time and end time a parameter of TRUN=4 would be used

UPDATE|NOUPDATE

Tells the APEC task whether to update the TMC or generate control statements that can be used by utilities TMSUPDTE, TMSAGGR, and TMSUDSNB. A parameter of UPDATE causes the TMC to be directly updated and will not create control statements for the above utilities. A parameter of NOUPDATE causes the TMC to not be updated, but generates control statements that can be used by the above utilities.

Default: UPDATE

Update Parameters While TMSAPEC is Active

Parameters may be temporarily updated while TMSAPEC is active through the use of the CTS Message (MSG) command. Parameters updated this way will remain in effect until TMSAPEC is terminated or the parameter is again changed. If a scan is currently running and a new, later STIME is specified, the current scan will not be stopped, however, if a new ETIME is specified that can be honored, it will. The DELAY and TRUN parms may also be applied to an active scan.

Use the following format for all parameters issued to the APEC subtask:

F CTS,MSG APEC, 'yyyyyy'

YYYYYY

Specifies the desired APEC parameters.

APEC Startup Examples

This example starts the APEC subtask started automatically each time CTS is started, and sets the following options:

- The scanning window is 11:30pm to 3:30am
- The window performs 2 scans
- There is a 0.05 second delay between scanning chains
- The window updates the TMC directly

Add the following parameters to the CTSSTART member in *hlq*.CTAPOPTN:

```
SET TASK(APEC) PGM(TMSAPEC)
SET TASK(APEC) PARM(STIME=2330,ETIME=0330,TRUN=2,DELAY=005,UPDATE)
START APEC
```

This example manually starts the APEC subtask with the following options:

- The scanning window is 8:00am to 2:00pm
- The window performs 1 scan
- There is a 0.15 second delay between scanning chains
- The TMC is not updated directly - transactions for the TMSAGGR, TMSUDSNB, and TMSUPDTE utilities are created.

```
F CTS,SET TASK(APEC) PGM(TMSAPEC)
F CTS,SET TASK(APEC) PARM(STIME=0800,ETIME=1400,DELAY=015,NOUPDATE)
F CTS,START APEC
```

In this example the APEC subtask starts automatically each time CTS is started with the following options:

- The scan starts at 10:00am every day
- The window performs 1 scan
- There is no delay between scanning chains
- The TMC is updated directly

Add the following parameters to the CTSSTART member in *hlq*.CTAPOPTN:

```
SET TASK(APEC) PGM(TMSAPEC)
SET TASK(APEC) PARM(STIME=1000,UPDATE)
START APEC
```

In this example the APEC subtask uses default initialization parameters each time CTS is started.

Specify the following parameters in the CTSSTART member in *hlq.CTAPOPTN*:

```
SET TASK(APEC) PGM(TMSAPEC)
START APEC
```

JCL Considerations for CTS Started Task

When the NOUPDATE parameter is specified the TMSUPDTE, TMSUDSNB, and TMSAGGR DD statements must be added to the CTS started task. If the NOUPDATE parameter is used and the DDs are not found, the APEC subtask will not start. If the UPDATE parameter is used and the DDs are in the CTS started task, the APEC subtask will ignore the DDs. CA recommends that these data sets be defined as generation data sets to allow the output of each scan to be saved separately. Define enough generations to temporarily retain at least all scans run in a single day. Do not write to these data sets using DISP=MOD processing. This will duplicate the transactions to fix the errors already given. The TMSRPT28 DD is required in both UPDATE and NOUPDATE mode.

The following DD statements are available:

TMSUPDTE

Creates a sequential data set which contains control statements for utility TMSUPDTE, used for initialization or correction of DSN17 and FLAG3 bit TMULTIF.

TMSUDSNB

Creates a sequential data set which contains control statements for utility TMSUDSNB, used for the initialization of field DSNBFVSN.

TMSAGGR

Creates a sequential data set which contains control statements for utility TMSAGGR to put unchained DSNB records back to the DSNB free chain.

TMSRPT28

Required to print the APEC log.

Example:

```
//TMSUPDTE DD DSN=tmsupdte.control.statements(+1),
//          DISP=(NEW,CATLG,DELETE),SPACE=(space),
//          DCB=BLKSIZE=n*80,UNIT=unit           Note
//TMSUDSNB DD DSN=tmsudsnb.control.statements(+1),
//          DISP=(NEW,CATLG,DELETE),SPACE=(space),
//          DCB=BLKSIZE=n*80,UNIT=unit           Note
//TMSAGGR  DD DSN=tmsaggr.control.statements(+1),
//          DISP=(NEW,CATLG,DELETE),SPACE=(space),
//          DCB=BLKSIZE=n*80,UNIT=unit           Note
//TMSRPT28 DD SYSOUT=*
```

Note: The BLKSIZE can be any multiple of the LRECL(80). Additional JCL updates may be needed to meet your site's installation standards. A large number of control cards can be generated if the NOUPDATE option is used and this is the first time that TMSAPEC has been run.

APEC Commands

Commands are sent to APEC using the CTS Message command (MSG), and can be issued at any time while TMSAPEC is active.

Commands issued to the APEC subtask have the following format:

F CTS,MSG APEC,*command*

STOP

Stops all scans, closes all files, and terminates the APEC task. Scanning will not restart until the APEC task is restarted. See the note following the example.

CANCEL

Requests that the APEC task cancel itself with a USER-111 abend. All scans are stopped and all files are closed. Scanning will not be restarted until the APEC task is restarted. See the note following the example.

SCAN

Starts one scan of the TMC.

HALT

Stops all scans and closes the TMC. A scan stopped by HALT cannot be RESUMED. Scanning of the TMC will restart at the next scheduled start time and will start from the beginning of the TMC. See the note following the example.

PAUSE

Pauses a scan, during which time the TMC remains open and the scan can be restarted from the point at which it was stopped.

RESUME

Restarts a scan that has been paused. RESUME can only restart a scan that was stopped by a PAUSE; it cannot restart a scan that was stopped by a HALT.

STATUS

Displays status information in one CTS log about the state of the APEC task and displays all parameters that have been set.

Example: Status information

```
ACTIVE - NOT SCANNING
NOUPDATE
DELAY: 0.05 SECOND
START TIME: 18:00
END TIME: 21:45
# OF SCANS IN WINDOW: 03
# OF SCANS STARTED: 00000002
# OF SCANS COMPLETED: 00000001
# OF ERRORS FOUND BY LAST SCAN: 00000004
# OF ERRORS FIXED BY LAST SCAN: 00000002
TOTAL ERRORS FOUND SINCE APEC STARTED: 00000001
TOTAL ERRORS FIXED SINCE APEC STARTED: 00000001
APEC STARTED: 05:57:11
WAITING
```

Note: If a scan of the TMC does not complete and NOUPDATE was specified, the control statements for TMSUPDTE, TMSUDSNB, and TMSAGGR may not have been created successfully. Do not use an incomplete set of control statements for these utilities.

CA 1 Data Base Services (DBS)

Before starting the DBS subtask, ensure that TMSINIT has started and CA 1 is either ACTIVE or BATCH ACTIVE.

If CA 1 is currently ACTIVE or BATCH ACTIVE when it is invoked, DBS opens both the CA 1 Tape Management Catalog (TMC) and the AUDIT file dynamically and waits on any future commands.

If CA 1 is *not* ACTIVE or BATCH ACTIVE when it is invoked, DBS waits and periodically tests the status of CA 1 again. If DBS has made several unsuccessful attempts at initializing and CA 1 is still not active, a message is issued to the operator and it continues waiting until either CA 1 is activated or DBS is stopped.

To use the DBS task, you do not need ENF and CCI.

Startup Procedure

DBS can be started manually or automatically when CTS is started. The recommended method is to have CTS automatically start the DBS subtask.

Automatic Startup

To automatically start the DBS subtask when CTS is started, add the following command to the CTSSYSIN DD in the member CTSSTART file in the CTS procedure.

```
START DBS
```

Once the CTSSYSIN DD in the member CTSSTART file is updated to contain the command listed above, CTS automatically starts the DBS subtask when CTS is started.

Manual Startup

Once the CTS task has been started, you can communicate with CTS by using a MODIFY command or by WTOR.

The following example shows how to manually start the DBS subtask under CTS using the MODIFY command:

```
F CTS,START DBS
```

DBS Commands

Commands are sent to DBS using the CTS Message command (MSG).

The following format should be used on all commands issued to the DBS subtask:

MSG DBS,xxxxxxxxxx

Where xxxxxxxxxx is the DBS command and must contain one of the following values:

STOP

To request DBS to close all files and terminate.

CANCEL

To request DBS to cancel itself with a User-111 Abend.

STATUS

To request DBS to provide the current status of CA 1, the TMC and AUDIT files.

OBTAIN TMC|AUDIT

To request DBS to OPEN either the TMC or the AUDIT file.

RELEASE TMC|AUDIT

To request DBS to CLOSE either the TMC or the AUDIT file.

Note: The OBTAIN and RELEASE commands require a parameter of either TMC or AUDIT to define the name of the file to be opened or closed. For this reason, the command and parameter entered must be enclosed within single quotes or parentheses, for example, 'OBTAIN TMC'.

CA 1 Health Checker Service (HCK)

The CTSCHK subtask of the Common Tape System provides support for the CA 1 health checks when you do not want to run the CA Common Services CAHCHECK address space. The CA 1® Tape Management health checks run under the IBM Health Checker for z/OS to do the following:

- Identify potential problems (CA 1 and other) in your z/OS environment.
- Recommend how to fix the potential problems.

This subtask is always started when the CTS address space starts. If you are not using the CA Common Services CAHCHECK address space, all available CA 1® Tape Management health checks are automatically discovered and started. If the CA Common Services CAHCHECK address space is available, TMSINIT automatically discovers and starts the CA 1 health checks in the CAHCHECK address space.

Startup Procedure

The HCK subtask is started when the CTS address space starts. No control statements are necessary to run this subtask.

Operator Commands

REINIT Command

F CTS,MSG,HCK,REINIT

This command will request that HCK stop and restart all available CA 1 health checks. After applying CA 1 maintenance that introduces new CA 1 health checks, this command can be used to cause the new health checks to be started. You do not have to stop and start the entire CTS address space.

START Command

F CTS,START,HCK

This command starts the HCK subtask. The HCK subtask is always started when the CTS address space initializes. Therefore, this command is needed only after you issue a STOP command and when you want the health check processing to resume.

STOP Command

F CTS,STOP,HCK

This command stops the HCK subtask. CA 1 health check processing will not be performed until either the CTS task is restarted or the START command is issued.

Online Label Interface (LAB)

External gummed labels are generated through the Online Label Interface (LAB) subtask of CTS.

Gummed labels produced for CA 1 from the various label generating programs (TMSLBLPR, TMSDISP, online requests and system option LAB=YES), use the Online Label Interface (LAB) to generate labels.

The Online Label Interface is made up of three components:

- The input processor (CTSLBLIN) which is used to obtain each input record and pass the information to the appropriate CA Earl program.
- CA Earl which is used to process the input records, format the label image and call the output processor. Member TMELBLS in CAI.CTAPEARL is the CA Earl example provided to generate the default gummed labels for CA 1. This member can be copied and modified to customize labels for your site.
- The output processor (CTSLBLOT) which is used to write a formatted label image to one or more output locations.

Processing Steps

The CTS procedure provided with this version uses member TMELBLS in CAI.CTAPEARL to create labels. The following example shows the code contained in the TMELBLS member with a discussion of the various sections. If modification of this is required to customize labels for your site, you can modify this code or make a copy for modification. Verify that the LBL parameter in the CTS procedure points to the name of the member that you want to use.

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!  SAMPLE OUTPUT LABELS FOR REELS AND CARTRIDGES                                !
!                                     1                                           !
!  THE FOLLOWING FORMATS ARE BEING PRODUCED IN THIS SAMPLE:                     !
!                                                                              !
!  REELS:                                                                      !
!    .....1.....2.....3.....4.....5.....6                                !
!                                                                              !
!  1  DSN.....                                                                    !
!  2                                                                              !
!  3    CJOB....          CDATE.....                                           !
!  4                                                                              !
!  5  DEN. TRTC LRECL. BLKSIZ RCFM EXPDT.....                                !
!  6                                                                              !
!  7          VOLSER VSEQ LBL          UNIT                                     !
!                                                                              !
!  CARTS:                                                                      !
!    .....1.....2.....3.....4.....5.....6                                !
!                                                                              !
!  1  DSN.....                                                                    !
!  2                                                                              !
!  3                                                                              !
!  4    CJOB....          CDATE.....                                           !
!  5          EXPDT.....                                                       !
!  6  DEN. TRTC LRECL. BLKSIZ RCFM                                           !
!  7                                                                              !
!  8  VOLSER VSEQ LBL          UNIT                                             !
!                                                                              !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```

OPTION NODUMP                                2
COPY LABELDEF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! OUTPUT LABEL IMAGE   10 ROWS BY 80 COLUMNS                                     !
!                                                                                   !
! "ROWS" DEFINE THE NUMBER OF LINES IN EACH LABEL FROM                          !
! THE FIRST LINE OF ONE LABEL TO THE FIRST LINE IN THE                         !
! NEXT LABEL.                                                                !
!                                                                                   !
! "COLUMNS" DEFINE THE NUMBER OF CHARACTERS IN EACH ROW.                      !
!                                                                                   !
! "LINE" DEFINES THE INDIVIDUAL ROW IN THE ARRAY.                             !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

DEF OUTPUT_LABEL (X 802) = ' ' 'OUTPUT' 'LABEL' 3
                                         4
DEF ROWS              = OUTPUT_LABEL 001-001 B 'ROWS'
DEF COLUMNS          = OUTPUT_LABEL 002-002 B 'COLUMNS'
DEF LINE  ARRAY 10    = OUTPUT_LABEL 003-082 X   5

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! WORK LINES FOR LABEL REEL IMAGE - 80 COLUMNS WIDE !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

DEF WORK_LINE (X 80) = ' '

DEF REEL01_DSN      = WORK_LINE      001-044 X
DEF REEL03_CJOB     = WORK_LINE      005-012 X
DEF REEL03_CDATE    = WORK_LINE      031-040 X

DEF REEL05_DEN      = WORK_LINE      002-005 X
DEF REEL05_TRTCH    = WORK_LINE      007-010 X
DEF REEL05_LRECL    = WORK_LINE      012-017 N
DEF REEL05_BLKSIZE  = WORK_LINE      019-024 N           6
DEF REEL05_RECFCM   = WORK_LINE      026-029 X
DEF REEL05_EXPDIT   = WORK_LINE      031-040 X

DEF REEL07_VOLSER   = WORK_LINE      016-021 X
DEF REEL07_VOLSEQ   = WORK_LINE      023-026 N
DEF REEL07_LABEL    = WORK_LINE      028-030 X
DEF REEL07_LUNIT    = WORK_LINE      037-040 X

```

```

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!  WORK LINES FOR LABEL CART IMAGE - 80 COLUMNS WIDE !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

DEF CART01_DSN      = WORK_LINE      001-044 X
DEF CART04_CJOB     = WORK_LINE      002-009 X
DEF CART04_CDATE    = WORK_LINE      015-024 X
DEF CART05_EXPDT    = WORK_LINE      015-024 X
DEF CART06_DEN      = WORK_LINE      002-005 X
DEF CART06_TRTCH    = WORK_LINE      007-010 X
DEF CART06_LRECL    = WORK_LINE      012-016 N
DEF CART06_BLKSIZE  = WORK_LINE      019-024 N
DEF CART06_RECFM    = WORK_LINE      026-029 X
DEF CART08_VOLSER   = WORK_LINE      002-007 X
DEF CART08_VOLSEQ   = WORK_LINE      009-012 N
DEF CART08_LABEL    = WORK_LINE      014-016 X
DEF CART08_LUNIT    = WORK_LINE      024-027 X

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!  THE FOLLOWING ROUTINE WILL GENERATE AND PRINT 10 CART SETUP LABELS  !
!  USING SETUP OUTPUTS DEFINED IF THE VALUE IN 'SETUP_REQUESTED' = 'Y'.!
!                                                                           !
!  IF REEL SETUP LABELS ARE PREFERRED, CHANGE THE VALUE IN THE         !
!  'SETUP_TYPE' FIELD TO 'REEL'.                                       !
!                                                                           !
!  IF MORE THAN 10 SETUP LABELS ARE DESIRED, CHANGE THE VALUE IN THE   !
!  'SETUP_MAXIMUM' FIELD TO THE DESIRED VALUE.                         !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

DEF SETUP_REQUESTED  (X 1)  = 'Y'
DEF SETUP_TYPE       (X 4)  = 'CART'
DEF SETUP_COUNT      (P 2.0) = 0                                     7
DEF SETUP_MAXIMUM    (P 2.0) = 10

IF SETUP_REQUESTED NOT = 'Y'
  GOTO GETTAG
ENDIF

SET SETUP_REQUESTED = 'N'

```

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! THE FOLLOWING OUTPUT ASSIGNMENTS ARE USED FOR 'SETUP LABELS' ONLY.  !
!                                                                           !
! NOTE: OUTPUT ASSIGNMENTS ARE SET HERE FOR 'SETUP LABELS' AND LATER  !
!       FOR ALL INPUT 'LABEL REQUESTS'.                               !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
SET PRT1_REQ = 'N'      NOTE - SET PRINTER1 FILE
SET PRT2_REQ = 'N'      NOTE - SET PRINTER2 FILE
SET PRT3_REQ = 'N'      NOTE - SET PRINTER3 FILE
SET WT01_REQ = 000      NOTE - SET WTO ROUTE CODE    8
SET WT02_REQ = 000      NOTE - SET WTO ROUTE CODE
SET WT03_REQ = 000      NOTE - SET WTO ROUTE CODE
SET CCI_DEST = ' '      NOTE - SET CCI DESTINATION

SETUP_LOOP:

SET SETUP_COUNT = SETUP_COUNT + 1

IF SETUP_COUNT > SETUP_MAXIMUM
    GOTO GETTAG
ENDIF
```

```

IF SETUP_TYPE = 'CART'
  SET OUTPUT_LABEL = ' '
  SET ROWS = 10
  SET COLUMNS = 80
  SET WORK_LINE = ' '
  SET CART01_DSN = 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
  SET LINE(/1/) = WORK_LINE

  SET WORK_LINE = ' '
  SET CART04_CJOB = 'XXXXXXX'
  SET CART04_CDATE = 'XXXXXXXXXX'
  SET LINE(/4/) = WORK_LINE
  SET WORK_LINE = ' '
  SET CART05_EXPDT = 'XXXXXXXXXX'
  SET LINE(/5/) = WORK_LINE
  SET WORK_LINE = ' '
  SET CART06_DEN = 'XXXX'
  SET CART06_TRTCH = 'XXXX'
  SET CART06_LRECL = 999999
  SET CART06_BLKSIZE = 999999
  SET CART06_RECFM = 'XXXX'
  SET LINE(/6/) = WORK_LINE
  SET WORK_LINE = ' '
  SET CART08_VOLSER = 'XXXXXX'
  SET CART08_VOLSEQ = 9999
  SET CART08_LABEL = 'XXX'
  SET CART08_LUNIT = 'XXXX'
  SET LINE(/8/) = WORK_LINE
  SET REQUEST = 'L'
  CALL CTSBLOT USING OUTPUT_LABEL COMM_AREA
  GOTO SETUP_LOOP
ENDIF

```

9

```

IF SETUP_TYPE = 'REEL'
  SET OUTPUT_LABEL = ' '
  SET ROWS = 10
  SET COLUMNS = 80

  SET WORK_LINE = ' '
  SET REEL01_DSN = 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
  SET LINE(/1/) = WORK_LINE

  SET WORK_LINE = ' '
  SET REEL03_CJOB = 'XXXXXXX' 10
  SET REEL03_CDATE = 'XXXXXXX'
  SET LINE(/3/) = WORK_LINE

  SET WORK_LINE = ' '
  SET REEL05_DEN = 'XXXX'
  SET REEL05_TRTCH = 'XXXX'
  SET REEL05_LRECL = 999999
  SET REEL05_BLKSIZE = 999999
  SET REEL05_RECFM = 'XXXX'
  SET REEL05_EXPDT = 'XXXXXXXXXX'
  SET LINE(/5/) = WORK_LINE
  SET WORK_LINE = ' '
  SET REEL07_VOLSER = 'XXXXXX'
  SET REEL07_VOLSEQ = 9999
  SET REEL07_LABEL = 'XXX'
  SET REEL07_LUNIT = 'XXXX'
  SET LINE(/7/) = WORK_LINE

  SET REQUEST = 'L'
  CALL CTSLBLT USING OUTPUT_LABEL COMM_AREA
  GOTO SETUP_LOOP
ENDIF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! THE FOLLOWING STATEMENTS READ THE NEXT INPUT RECORD !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

GETTAG:
GET TAPEDB

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! IF END-OF-FILE IS REACHED - CALL CTSLBLT TO CLOSE ALL FILES !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

IF TAPEDB = 'E'
  SET REQUEST = 'C'
  CALL CTSLBLT USING LABEL_RECORD COMM_AREA
  GOTO E0J
ENDIF

```



```

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! IF A SPINOFF REQUEST IS READ - CALL CTSBLLOT TO PROCESS IT                !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

IF REQUEST = 'S'
    CALL CTSBLLOT USING LABEL_RECORD COMM_AREA
    GOTO GETTAG
ENDIF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! SELECT (OR BYPASS) INPUT RECORDS AT THIS POINT      11                    !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! THE FOLLOWING OUTPUT ASSIGNMENTS ARE USED FOR 'LABEL REQUESTS' ONLY.      !
!                                                                           !
! NOTE: OUTPUT ASSIGNMENTS ARE SET HERE FOR ALL 'LABEL REQUESTS' AND      !
!       IN THE SETUP LABELS SECTION FOR 'SETUP LABELS'.                    !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
SET PRT1_REQ = 'N'      NOTE - SET PRINTER1 FILE
SET PRT2_REQ = 'N'      NOTE - SET PRINTER2 FILE
SET PRT3_REQ = 'N'      NOTE - SET PRINTER3 FILE
SET WT01_REQ = 000      NOTE - SET WTO ROUTE CODE
SET WT02_REQ = 000      NOTE - SET WTO ROUTE CODE    12
SET WT03_REQ = 000      NOTE - SET WTO ROUTE CODE
SET CCI_DEST = ' '      NOTE - SET CCI DESTINATION

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! IF A LABEL IMAGE IS READ - CALL CTSBLLOT TO PRINT THE LABEL IMAGE        !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

IF REQUEST = 'L'
    CALL CTSBLLOT USING LABEL_RECORD COMM_AREA
    GOTO GETTAG
ENDIF

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! IF A VOLUME OR DATA RECORD IS READ AND IT CONTAINS A CARTRIDGE          !
! DENSITY, FORMAT A LABEL IMAGE FOR CARTRIDGES.                          !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

IF REQUEST = 'V' OR REQUEST = 'D'
    IF DEN = '38K' OR DEN = '38KC'                                         13
        SET OUTPUT_LABEL          = ' '
        SET ROWS                   = 10
        SET COLUMNS               = 80

```

```
!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-1 !
!!!!!!!!!!!!!!!!!!!!

          SET WORK_LINE          = ' '
          SET CART01_DSN          = DSN
          SET LINE(/1/)           = WORK_LINE

!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-4 !
!!!!!!!!!!!!!!!!!!!!

          SET WORK_LINE          = ' '
          SET CART04_CJOB          = CJOB
          SET FUNCTION = 'TO_EXTERNAL'
          CALL CTSEDATE USING FUNCTION CART04_CDATE CDATE DATE_FMT
          SET LINE(/4/)           = WORK_LINE

!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-5 !
!!!!!!!!!!!!!!!!!!!!

          SET WORK_LINE          = ' '
          SET FUNCTION = 'TO_EXTERNAL'
          CALL CTSEDATE USING FUNCTION CART05_EXPDT EXPDT DATE_FMT
          SET LINE(/5/)           = WORK_LINE

!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-6 !
!!!!!!!!!!!!!!!!!!!!

          SET WORK_LINE          = ' '
          SET CART06_DEN          = DEN
          SET CART06_TRTCH         = TRTCH
          SET CART06_LRECL         = LRECL
          SET CART06_BLKSIZE       = BLKSIZE
          SET CART06_RECFM         = RECFM
          SET LINE(/6/)           = WORK_LINE

!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-8 !
!!!!!!!!!!!!!!!!!!!!

          SET WORK_LINE          = ' '
          SET CART08_VOLSER        = VOLSER
          SET CART08_VOLSEQ        = VOLSEQ
          SET CART08_LABEL         = LABEL_TYPE
          SET CART08_LUNIT         = LUNIT
          SET LINE(/8/)           = WORK_LINE
```

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! CALL CTSLBLOT WITH A LABEL REQUEST TO PRINT THE LABEL IMAGE           !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

16

```
      SET REQUEST = 'L'
      CALL CTSLBLOT USING OUTPUT_LABEL COMM_AREA
      GOTO GETTAG
    ENDIF
  ENDIF
```

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! IF ANY OTHER VOLUME OR DATA RECORD IS PROCESSED, ASSUME IT CONTAINS !
! A REEL DENSITY AND FORMAT A LABEL IMAGE FOR REELS.                   !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
  IF REQUEST = 'V' OR REQUEST = 'D'
    SET OUTPUT_LABEL      = ' '
    SET ROWS              = 10
    SET COLUMNS          = 80
```

17

```
!!!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-1 !
!!!!!!!!!!!!!!!!!!!!!!
```

```
      SET WORK_LINE      = ' '
      SET REEL01_DSN     = DSN
      SET LINE(/1/)      = WORK_LINE
```

```
!!!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-3 !
!!!!!!!!!!!!!!!!!!!!!!
```

```
      SET WORK_LINE      = ' '
      SET REEL03_CJOB     = CJOB
      SET FUNCTION = 'TO_EXTERNAL'
      CALL CTSEDATC USING FUNCTION REEL03_CDATE CDATE DATE_FMT
      SET LINE(/3/)      = WORK_LINE
```

```
!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-5 !
!!!!!!!!!!!!!!!!!!!!

      SET WORK_LINE          = ' '
      SET REEL05_DEN         = DEN
      SET REEL05_TRTCH       = TRTCH
      SET REEL05_LRECL       = LRECL
      SET REEL05_BLKSIZE     = BLKSIZE
      SET REEL05_RECFM       = RECFM
      SET FUNCTION = 'TO_EXTERNAL'
      CALL CTSEDATE USING FUNCTION REEL05_EXPDT EXPDT DATE_FMT
      SET LINE(/5/)          = WORK_LINE

!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-7 !
!!!!!!!!!!!!!!!!!!!!

      SET WORK_LINE          = ' '
      SET REEL07_VOLSER      = VOLSER
      SET REEL07_VOLSEQ      = VOLSEQ
      SET REEL07_LABEL       = LABEL_TYPE
      SET REEL07_LUNIT       = LUNIT
      SET LINE(/7/)          = WORK_LINE

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! CALL CTSBLLOT WITH A LABEL REQUEST TO PRINT THE LABEL IMAGE          !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

      SET REQUEST = 'L'
      CALL CTSBLLOT USING OUTPUT_LABEL COMM_AREA
      GOTO GETTAG
ENDIF
END
```

1

The first part of the code is a comment section showing the output format of labels produced by default. Refer to Label formats for REELS and CARTS for additional information.

2

The COPY command brings in the CA EarI label and record definitions from members LABELDEF, CTEARLCM and CTEARLCN from CAI.CTAPECPB.

3

This section of code defines the storage array for the labels. The definition may be larger than the actual storage requirements.

4

The size for OUTPUT_LABEL is calculated by (rows * columns)+2.

5

The LINE array should be set to the number of rows used to calculate OUTPUT_LABEL.

For example:

```
DEF OUTPUT_LABEL (X 1282) = ' ' 'OUTPUT' 'LABEL'
.
.
DEF LINE ARRAY 16 = OUTPUT_LABEL 003-082 X
```

This defines a storage area that can accommodate labels up to 16 rows by 80 columns.

6

Defines the work lines used to build the label array. For ease in understanding the variable names in this section are defined as ttttnn-ffff where tttt indicates REEL or CART, nn is the two-digit line number within the label and ffff describes the value to be printed.

The variables are defined as sub-strings of WORK_LINE at the positions indicated.

For example:

The fourth line of the cartridge label displays the creating job in columns 2-9 and the create date in columns 15-24.

7

The program allows you to produce setup labels if they are desired. The delivered sample automatically produces 10 setup labels in a CART format. If the REEL format is preferred, the number of setup labels needs to be changed or the setup labels are to be bypassed entirely, modify the default values in this section.

8

Defines the output routine setup labels only. The values can be changed manually in this member or by SMP using USERMOD TMELBLS.

Note: For more information about the values, see Output Assignments.

9

This is the code to produce CART setup labels. If you have modified the actual label output and want the setup labels to reflect your changes, you must modify this code also.

10

The code to produce REEL setup labels. If you have modified the actual label output and want the setup labels to reflect your changes, you must modify this code also.

11

Logic can be added at this point to select or bypass labels based on input information. For example, TMSLBLPR is producing labels for an audit shared between three CPUs - XE01, XE02, PD03. PD03 is a test machine so no labels are required for those tapes.

For example:

```
IF CPUID='PD03'  
  GOTO GETTAG  
ENDIF
```

This bypasses records based on the CPUID field (PD03) in the input record.

12

Defines the output routine for non-setup labels. The values can be changed manually in this member or by SMP using USERMOD TMELBLS.

Note: For more information about the values, see Output Assignments.

13

These SET commands determine the actual number of rows and columns to be used and can be modified to site standards. Verify that the data fields defined at 6 are contained within the new label size.

The SET OUTPUT_LABEL statement clears the storage area.

14

The beginning of the section that moves the data into the work line and stores the completed line into the corresponding element of the array named LINE.

15

CTSEDATE is the CA Earl date function. The DATE_FMT parameter indicates the date format to be used on your label. The default is defined by the CA 1 parameter DATEFMT in the TMOOPTxx member of hlq.CTAPOPTN. If you want a specific format for the label, use the SET command before the CALL to indicate the format.

For example:

```
SET DATE_FMT = 'mmdd yyyy'
```

This example produces a label with a create date in the format AUGDD YYYY.

16

Set the request indicator to L and print the label image.

17

Begins the section for REEL tape labels. The logic for this section is the same as 14 through 16 above.

Modify Labels

This section gives an example to modify the code to produce the following cartridge label. The company name has been added to the label, the DSN moved to line 3, the dates on the label are displayed in the format MMMDD/YYYY, no setup labels are generated and the actual labels are routed to WTO route code 13. The labels are defined as 12 rows by 50 columns and the extra rows are used for alignment. REEL type labels are not modified but are routed to WTO route code 15. The lines added/modified are indicated by the sequence numbers which correspond to the code in TMELBLS.

CARTRIDGE LABEL:

```

      ....+....1....+....2....+....3....+....4....+....5....+....6
-----
1 |      COMPUTER ASSOCIATES INTERNATIONAL, INC.      |
2 | |                                                    |
3 | DSN.....|
4 | CJOB....  CDATE....|
5 |           EXPDT....|
6 | DEN  TRTC LRECL. BLKSIZ RCFM|
7 | |                                                    |
8 | VOLSER VSEQ LBL      UNIT|
-----

```

CODE UPDATES:

```

DEF OUTPUT_LABEL (X 602) = ' ' 'OUTPUT' 'LABEL'
DEF LINE_ARRAY 12       = OUTPUT_LABEL 003-052 X
DEF WORK_LINE (X 50)    = ' '
DEF CART01_COMPANY      = WORK_LINE 005-043 X
DEF CART03_DSN           = WORK_LINE 001-044 X
DEF SETUP_REQUESTED     (X 1) = 'N'
SET DATE_FMT            = 'MMMDD/YYYY'
SET WT01_REQ            = 013
SET ROWS                = 12
SET COLUMNS            = 50
SET CART01_COMPANY      = 'COMPUTER ASSOCIATES INTERNATIONAL,INC.'
!!!!!!!!!!!!!!!!!!!!!!
! FORMAT LINE-3 !
!!!!!!!!!!!!!!!!!!!!!!

SET WORK_LINE           = ' '
SET CART03_DSN          = DSN
SET LINE(/3/)           = WORK_LINE

SET WT01_REQ            = 015
SET ROWS                = 12
SET COLUMNS            = 50

```

User Data Access

User data is a free-form text that can be passed to your CA Earl label formatter. Real-time Auxiliary Disposition provides this capability when the user data option is specified. See [Real-time Auxiliary Disposition](#) (see page 68). Up to 51 characters of free-form text may be passed for each volume. The length of the user data is provided in the field LABEL_USERDATA_LEN, and the actual text is provided in the field LABEL_USERDATA.

The following is an example on modifying TMEBLS to move this user data to the output label.

Add the following line to the section titled 'WORK LINES FOR LABEL CART IMAGE - 80 COLUMNS WIDE' after 'DEF CART01_DSN ...'

```
DEF CART02_USR = WORK_LINE      001-051 X
```

Add the following lines to the SETUP LABELS section for CARTs:

```
SET WORK_LINE = ' '  
SET CART02_USR= 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'  
SET LINE(/2/) = WORK_LINE
```

In the format for line 2 for cartridges, code the following:

```
!!!!!!!!!!!!!!!!!!!!  
! FORMAT LINE-2 !  
!!!!!!!!!!!!!!!!!!!!  
  
SET WORK_LINE      = ' '  
SET CART02_USR     = LABEL_USERDATA  
SET LINE(/2/)      = WORK_LINE
```


Input, Output and Label Examples

The following is a description of how to code the //INPUT DD to the CTS job stream. It will point to the CTS input as described below.

```
*-----*
*      1. EACH LABEL REQUEST MUST START WITH THE LABEL REQUEST TYPE      *
*      KEYWORD ('VOLUME', 'LABEL' OR 'SPINOFF').                          *
*                                                                           *
*      2. EACH LABEL REQUEST AND ALL KEYWORDS ASSOCIATED WITH THE         *
*      LABEL REQUEST MUST EITHER FIT ON ONE CARD (WITH NO                *
*      CONTINUATION CHARACTER IN COLUMN 72 OR EACH CARD INPUT            *
*      ASSOCIATED WITH A LABEL REQUEST MUST CONTAIN A                   *
*      CONTINUATION CHARACTER IN COLUMN 72 IF KEYWORDS FOLLOW.            *
*                                                                           *
*      3. CARD IMAGE RECORDS ENDING IN A COMMA MUST CONTAIN A            *
*      CONTINUATION CHARACTER IN COLUMN 72.                               *
*                                                                           *
*      4. COMMENTS MUST CONTAIN AN ASTERISK (*) IN COLUMN 1 AND          *
*      NOT BE INTERMIXED WITH CONTINUED 'USERDATA' OR 'LABEL'           *
*      KEYWORD VALUES.                                                  *
*                                                                           *
*      5. IF VALUES FOR KEYWORD'S 'USERDATA' OR 'LABEL' ARE             *
*      CONTINUED FROM ONE RECORD TO THE NEXT, COLUMN 71 WILL            *
*      BE USED AS THE LAST COLUMN OF THE DATA ON CONTINUED              *
*      RECORDS AND COLUMN 1 OF THE NEXT RECORD WILL BE USED AS           *
*      THE NEXT COLUMN OF THE DATA UNLESS THE ENDING DELIMITER          *
*      IS FOUND. THE ENDING DELIMITER MAY BE PLACED ANYWHERE            *
*      IN COLUMNS 1 THROUGH 71.                                         *
*                                                                           *
*      6. EACH LABEL REQUEST (AND ALL KEYWORDS ASSOCIATED WITH THE        *
*      LABEL REQUEST) WILL BE VALIDATED. IF NO ERRORS OCCUR,            *
*      THE LABEL REQUEST WILL BE PROCESSED AND SENT TO CA-EARL.          *
*                                                                           *
*      7. THE 'USERDATA' KEYWORD IS ONLY VALID WITH A 'VOLUME'           *
*      LABEL REQUEST. THE FOLLOWING ARE SAMPLE VALID REQUESTS:          *
*                                                                           *
*      SAMPLES. . .                                                    72 *
*      1.  VOLUME=123456,PRINTER=(1,3),WTOROUTE=(1,15,20)              *
*                                                                           *
*      2.  VOLUME=123456,                                             X *
*          PRINTER=(1,3),                                           X *
*          WTOROUTE=(1,15,20)                                         *
*                                                                           *
*      3.  VOLUME=123456,USERDATE='.....X' *
*          .....',CCIDEST='.....' *
*                                                                           *
*      4.  LABEL=10,80,'.....X' *
*          .....',PRINTER=2 *
*                                                                           *
```

```
*
*      5.    SPINOFF
*      6.    SPINOFF, PRINTER=(1,3)
*
*      7.    STATUS
*-----*
```

Output Assignments

Up to seven outputs can be requested on each label image. Three different printers can be requested (PRINTER1, PRINTER2 and PRINTER3), up to three unique WTO Route Codes, and additionally a CAICCI destination.

Unless the CTS started task procedure in CAI.CTAPPROC is modified, the particular printer is dynamically allocated as a SYSOUT using the OUTPUT statements OUTPUTP1, OUTPUTP2, OUTPUTP3 for printers PRINTER1, PRINTER2 and PRINTER3 respectively.

If a subsequent SPINOFF request is processed, all printers that were initially dynamically allocated are closed (and freed), reallocated and reopened. This frees the current SYSOUT to be printed.

Fields: PRT1_REQ, PRT2_REQ and PRT3_REQ

Representing requests for the PRINTER1, PRINTER2, and PRINTER3 files respectively, these fields may contain a value of:

Y

Specifies that the label be output to the respective printer.

N

Specifies that the respective printer is not to be used.

Fields: WTO1_REQ, WTO2_REQ and WTO3_REQ

These fields can be used to request up to three WTO route codes to be used to print the label image. Valid values for these fields are 000 through 128. Zero in the option indicates that the specific WTO route code is not to be used for the request.

Using WTO route code 011 may conflict with warning and error messages issued by the input and output processors and is discouraged.

Field: CCI_DEST

To generate the labels on one CPU and route the label image to another CPU to print, use the CAICCI Destination assignment.

If this is the case, have two LAB subtasks running under CTS; one on the sending CPU and one on the receiving CPU. Additionally, the two CPUs need to be on the same CAIENF/CAICCI network to facilitate the cross-CPU communications.

Note: For more information about establishing CAIENF/CAICCI networks, see the *CA Common Services for z/OS Administrator Guide*.

A CAICCI destination code may be entered if you want the output processor to route the label image to another common tape labels started task on another CPU. The value entered in this field must contain CA\$FO immediately followed by the four-character CPU identification where the receiving Common Tape labels started task is executing. The output processor validates the CAICCI Identification entered is up and operational before sending the label request through CAICCI Communications to the secondary label processor. No output assignments are propagated on label requests sent using CAICCI services. Any existing assignments can be modified by using the SET statement.

The ORIG_ID field contains the originating CAICCI Identification of the current request. SOURCE_ID contains a one-position value to identify the source of the current request. Possible values are B (Batch) or O (Online). These two fields can be used by the CA Earl program to select labels for printing.

If none of these outputs are requested, PRINTER1 is used by the output processor to print all label images.

Note: The delivered sample (TMELBLS) does not establish any output options, so label images for both REEL and CART densities (setup and actual labels) default to the same output (PRINTER1). If you use both REEL and CART densities, modify a copy of the sample to establish a different output for each type of label being generated.

Sample Labels

The following pages describe the sample labels that are delivered for your use. These samples illustrate a few of the ways you can generate labels using the input and output processors through CA Earl with the Online Label Interface.

Included in the sample is a section that automatically produces setup labels for printer alignment. The sample prints 10 setup labels using either the CART or REEL format. This section can be bypassed altogether with a minor change to the sample. Additionally, the section defaults to using the CART format and the default printer assignment. If this or other modifications to the section are required, you may want to modify the sample prior to using it.

These samples probably do not conform to your shop standards. You should copy these for your own use and make the necessary modifications to your own version.

Label formats for REELS and CARTS

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!  SAMPLE OUTPUT LABELS FOR REELS AND CARTRIDGES                                !
!                                                                                   !
!  THE FOLLOWING FORMATS ARE BEING PRODUCED IN THIS SAMPLE:                       !
!                                                                                   !
!  REELS:                                                                           !
!    ....+....1....+....2....+....3....+....4....+....5....+....6             !
!                                                                                   !
!  1  DSN.....                                                                    !
!  2                                                                                   !
!  3      CJOB....                        CDATE.....                            !
!  4                                                                                   !
!  5  DEN. TRTC LRECL. BLKSIZ RCFM EXPDT.....                                    !
!  6                                                                                   !
!  7              VOLSER VSEQ LBL          UNIT                                  !
!                                                                                   !
!  CARTS:                                                                           !
!    ....+....1....+....2....+....3....+....4....+....5....+....6             !
!                                                                                   !
!  1  DSN.....                                                                    !
!  2                                                                                   !
!  3                                                                                   !
!  4  CJOB....      CDATE.....                                                    !
!  5              EXPDT.....                                                       !
!  6  DEN. TRTC LRECL. BLKSIZ RCFM                                                 !
!  7                                                                                   !
!  8  VOLSER VSEQ LBL          UNIT                                                !
!                                                                                   !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

The above formats were extracted from the delivered sample for CART and REEL labels.

In the sample format, the field names occupy the location where the field values are placed. The CART format is used for cartridge densities and all others default to the REEL format.

Additionally, the REEL format occupies 7 rows, each containing 44 columns while the CART format occupies 8 rows with each containing 44 columns. In the delivered sample, one output label is generated for both formats with ROWS=10 and COLUMNS=80. This provides 3 blank rows between each label for REEL labels and 2 blank rows between each label for CART labels.

Sample REEL Setup Label Output

The following example is a sample output of setup labels generated using the REEL label format.

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
      XXXXXXXX                      XXXXXXXX
XXXX XXXX 999999 999999 XXXX XXXXXXXXXXXX
              XXXXXX 9999 XXX      XXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
      XXXXXXXX                      XXXXXXXX
XXXX XXXX 999999 999999 XXXX XXXXXXXXXXXX
              XXXXXX 9999 XXX      XXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
      XXXXXXXX                      XXXXXXXX
XXXX XXXX 999999 999999 XXXX XXXXXXXXXXXX
              XXXXXX 9999 XXX      XXXX

```

Sample REEL Label Output

The following example is a sample output of labels generated that contained a REEL density.

```

C884.PX077F.DJBLG.MG.GEO.MAP
      B05GE0MT                      YYYY/210
6250 9TRK 004112 004116 VB      YYYY/281
              100021 0001 SL          0480
U831.PSR756.CURADF.SORTFILE
      SDD26                      YYYY/184
6250 9TRK 002000 032000 FB      YYYY/189
              100022 0001 SL          0480
F884.P2100F.KIRSCH.BKUP.SENDP
      B14BKUP                      YYYY/048
6250 9TRK 000110 023430 FB      LDATE/005
              100023 0001 SL          0480

```

Sample CART Setup Label Output

The following example is a sample output of setup labels generated using the CART label format.

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXX      XXXXXXXXXX
              XXXXXXXXXX
XXXX XXXX 99999 999999 XXXX
XXXXXX 9999 XXX      XXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXX      XXXXXXXXXX
              XXXXXXXXXX
XXXX XXXX 99999 999999 XXXX
XXXXXX 9999 XXX      XXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXX      XXXXXXXXXX
              XXXXXXXXXX
XXXX XXXX 99999 999999 XXXX
XXXXXX 9999 XXX      XXXX
```

Sample CART Label Output

The following example is a sample output of labels generated that contained a CART density.

```
C831.CPDDX0.NONSILO.NSAALST.YR92
A05X001Y      YYYY/040
              YYYY/073
38K 18TK 00133 006118 FBA
500001 0001 SL      04A3
R888.PX4770.LHBM.WEEKLY.G0064V00
NDVDULH2      YYYY/045
              CATALOG
38K 18TK 01200 031204 VB
500002 0002 SL      049F
FDRABR.VGROUPF.B191362A
DPCUD555      YYYY/362
              YYYY/034
38K 18TK 00000 032760 U
500003 0001 SL      049E
```

Initialization Procedures

Input Requests

The input processor receives all input records and requests through the CA Common Services for z/OS services, CAICCI - Common Communicators Interface. These requests include the following:

V (Volume)

A volume lookup is performed against the current tape database

D (Data)

The common tape record is built based on the information available at the time of the request

L (Label)

The current record contains a preformatted label image ready to print

S (Spinoff)

This request closes and reopens any dynamically allocated printers

C (Close)

This request is used to ask the Output processor to close all files

The installed tape management system programmatically generates each label request or data record and sends the information to the label processor using the CA Common Services for z/OS Services CAICCI - Common Communicators Interface.

Output Labels

The label output processor supports up to seven outputs for a single label image. Up to three WTO route codes may be used and up to three label printers. Additionally, a CAICCI destination can be used to redirect the label image to another label processor running on another CPU.

In this case, start the CTS task and LAB subtask on the secondary CPU too. Additionally, the CA Common Services for z/OS Services CAICCI network must include both the primary and secondary CPU.

Once this is in place, the client can direct the label image to the secondary CPU by defining the CAICCI destination as an output assignment on the label request within the CA Earl component.

CA Common Services for z/OS Services CAICCI

CA Common Services for z/OS Services CAICCI/CAIENF are used throughout the label processing. The following CAICCI node prefixes are used:

CA\$FO...

Identifies the input label processor that receives all label requests.

CA\$FS...

Identifies the originating sender of the label request.

CA\$FW...

Identifies the output label processor when used to re-route a label image.

Note: For more information about CAICCI, see the *CA Common Services for z/OS Administrator Guide*.

LAB Description

When LAB is started, it initiates the label processor and waits for either the label processor to complete or a command from the operator to be entered.

The label processor completes under any of the following conditions:

- The label input processor received a CLOSE request
- An error was detected in the CA Earl component
- An end-of-file condition was forced in the CA Earl component
- A fatal error occurred in either the input or output label processor
- A STOP or CANCEL command was passed to the LAB subtask

The CLOSE request shuts down the entire label process in a normal fashion. CA Common Services for z/OS Services CAICCI used in the input processor is terminated. The CA Earl component receives the CLOSE request which in turn is passed to the output label processor, requesting it to close all opened files and terminate any use of CA Common Services for z/OS Services CAICCI.

The following command in the LAB subtask is used to send the CLOSE request to the label processor.

F CTS,STOP LAB

or

F CTS,MSG LAB,STOP

If the CA Earl component detects an error, it forces an END-OF-FILE condition which in turn requests the label processor to shut down. The same is true for any fatal errors that may occur.

Startup Procedure

The CTS startup procedure contains all JCL required to initiate the LAB subtask including the label processor and the CA Earl component.

LAB can be started manually or automatically when CTS is started. CA Common Services for z/OS Services CAIENF must have started and CAICCI communications must be up and operational before LAB is started.

Automatic Startup

To automatically start the LAB subtask when CTS is started, add the following command to the CTSSYSIN DD in the member CTSSTART file in CTS:

```
START LAB
```

Once the CTSSYSIN DD in the member CTSSTART file is updated to contain the command listed above, CTS automatically starts the LAB subtask when CTS is started.

Manual Startup

Once the CTS task has been started, you can communicate with CTS by using a MODIFY command or by WTOR.

The following example shows how to manually start the LAB subtask under CTS by using the MODIFY command:

```
F CTS,START LAB
```

LAB Commands

Commands are sent to LAB using the CTS Message command (MSG). The following format should be used on all commands issued to the LAB subtask:

MSG LAB,xxxxxxxx

Where xxxxxxxxx is the LAB command and must contain one of the following values:

STOP

to terminate label processing in a normal fashion

CANCEL

to cancel LAB including label processing with a User-111 Abend

STATUS

to request a current label processing status

VOLUME xxxxxx

to send a VOLUME request through the CA Common Services for z/OS Services CAICCI to the label processor.

SPINOFF |xxxxxxxx

to send a SPINOFF request through CA Common Services for z/OS Services CAICCI to the label processor.

The STOP command issues a CLOSE request to the label processor which terminates the process in a normal fashion. Once termination is complete, the LAB subtask ends.

The STATUS command issues a request that provides two messages indicating the time and date the label processor started with accumulated totals of VOLUME, DATA, LABEL and SPINOFF CAICCI requests that have been received. Additionally, an error count associated with VOLUME requests is provided.

The VOLUME command directs a request for a specific volume serial number to be processed. This command requires a parameter containing the volume serial number associated with the request. For this reason, the command and parameter entered must be separated by a blank and enclosed within single quotes or parentheses, for example, 'VOLUME 980010'.

The SPINOFF command can be entered with or without a parameter.

If the SPINOFF command is entered without a parameter, the label processor closes, reallocates, and reopens ALL printer files that were originally dynamically allocated as a SYSOUT file, thus freeing the current SYSOUT to be printed.

If the SPINOFF command is entered with a parameter, the parameter must be either PRINTER1, PRINTER2 or PRINTER3 indicating the specific printer to which the request applies. Additionally, only the printer specified is closed, reallocated and reopen, freeing the current SYSOUT to be printed. In this case, the command and parameter entered must be separated by a blank and enclosed within single quotes or parentheses, for example, 'SPINOFF PRINTER1'.

Command

The CTS address space is created by starting the CTS task. The CTS task is a multitasking supervisor, which provides support for CTS Services. It provides the means for starting, stopping, and controlling CTS subtasks. The CTS task also provides display and diagnostic facilities. The following applies to commands:

- The CTS task processes commands from console modify, CTSSYSIN in the member CTSSTART file, CTS internal message facility, and reply to an outstanding WTOR.
- Commands from any source are free-form in MACRO type or TSO type syntax.
- The syntax is a command-word followed by parameters separated by either blanks or commas.
- TSO style comments (/*.....*/) may appear anywhere in the command line or alone on the command line.
- Parameters can be keyword, positional or value, and they can be specified in any order.
- Keyword parameters can be of either the KWD= or KWD() style.
- Values are constants like SETUP, and can be specified as the alternate NOSETUP.
- Numbers specified for positional and keyword parameters can be from 1- to 15-digits with any number of leading zeros.
- Parameters which contain blanks, commas or special characters must be enclosed in '.....' or (.....).

Examples: CTS

```
cmd  pos1,pos2,pos3,value1,pos4,kwd1=??,kwd2=???  
cmd  pos1 pos2 pos3 value1 pos4 kwd1(??) kwd2(???)  
cmd  ,pos2                /* omit 1st PARM */  
cmd      value2  NOvalue1  kwd2=??  kwd1(???)
```

Command Descriptions

START or S

Start a CTS subtask for the task ID. If the task ID is defined to CTS the task ID is the only parameter required. If not defined to CTS the PGM parameter must be used to identify the program to attach. The other parameters may be used to specify or override defined values.

STOP or P

Request a subtask to complete.

DISPLAY or D

Request a display of CTS options, defined tasks, or active tasks.

SET

Set one or more CTS options, or a CTS subtask definition.

SPINOFF

Release SYSOUT for printing.

MSG or M

Send text to a CTS subtask.

CANCEL

Request a CTS subtask to ABEND with a user 111 and dump.

FORCE

Detach a CTS subtask causing its forceable termination.

STATUS

Request subtask to display status information.

The parameter and description for each command is as follows:

START Command

Task ID

Required. One to four characters must be unique to CTS address space.

PGM(nnnnnnnn)

Program name. Required if task not defined to CTS. This is the name of the program to be attached.

PARM(nnn)

Data to be sent to the subtask by internal message.

Default: no data

RETRY(nnn)

The number of times a subtask is to be restarted after completing with a nonzero completion code.

Default: 0

STOP Command

tid or ALL

Task ID. One to four characters must be unique to CTS address space. If ALL is specified a STOP command is sent to every CTS subtask. Only those tasks which have had a START command issued can be stopped.

DISPLAY Command

OPTS or O

Display CTS options.

TASKS or T

Display all tasks defined to CTS.

ACTIVE or A

Display all currently active subtasks.

SET Command for Options

WTOR or NOWTOR

Option for outstanding WTOR. If option is WTOR, then CTS keeps a WTOR issued to receive operator commands.

SNAP or NOSNAP

Option for recording diagnostic snap dumps. If the option is SNAP, the dumps are written to the dynamically allocated DD named CAISNP. This can be printed by using the SPINOFF SNAP command. The SYSOUT class is specified in the JCL statement named SPINOFF in the CTS started task.

LOG or NOLOG

Option for recording message log. If the option is LOG, a log of messages is written to the dynamically allocated DD named CQAIMSG. This can be printed by using the SPINOFF LOG command. The SYSOUT class is specified in the JCL statement named SPINOFF in the CTS started task.

SET Command to Define Subtasks

tid

Task ID. Required. One to four characters must be unique to CTS address space.

PGM(.....)

Program name. Required. This is the name of the program to be attached.

PARM(...)

Data to be sent to the subtask via internal message.

Default: no data

RETRY(...)

The number of times a subtask is to be restarted after completing with a nonzero completion code.

Default: 0

SPINOFF Command

tid

Task ID. One to four characters must be unique to CTS address space.

MSG Command

tid or ALL

Task ID. One to four characters must be unique to CTS address space. If ALL is specified the text command is sent to every CTS subtask.

text

Text to be sent to subtask. If this text contain blanks, commas or special characters it must be enclosed in '.....' or (.....).

CANCEL Command

tid or ALL

Task ID. One to four characters must be unique to CTS address space. If ALL is specified a CANCEL command is sent to every CTS subtask. Only those tasks which have had a START command issued may be canceled.

FORCE Command

tid

Task ID. One to four characters must be unique to CTS address space. Required. Only those tasks which had a START command may be forced.

STATUS Command

tid or ALL

Task ID. One to four characters must be unique to CTS address space. If ALL is specified a STATUS command is sent to every CTS subtask. Each task which can report status displays or logs its status. The format and content of status varies by task. The CMD task is the same as issuing a "Display OPT" and "Display Active".

Distributed Tape Support (DTS)

Distributed Tape Support (DTS) provides a mechanism for recording the tape inventories of distributed backup servers in the TMC. It provides a complete picture of all of the tapes in the entire enterprise, including both mainframe and distributed. Various information is collected based on the distributed backup product that is in use. This information includes the volume serial number, creation date, and other data relative to tape media. The following distributed backup products are currently supported:

- CA ARCserve® Backup for Windows r11
- CA ARCserve® Backup for Unix r11
- Veritas NetBackup for Windows
- Tivoli Storage Manager for Windows

This service runs in the CTS address space by the program named CTSDTS under the service name DTS.

iSponsor Technology CTSDTS uses the CA iSponsor/iGateway technology developed for the CA Portal product to retrieve data from the remote backup servers over TCP/IP. iSponsors are hosted by the iGateway that resides on the target server. The iGateway is a layer of software that communicates with the IP stack to receive and respond to requests for iSponsors.

The iSponsors are applets that consist of multiple methods. These methods extract data from the server and transmit the data over the iGateway to the requestor. Each method that belongs to a particular iSponsor provides a unique view of the backup server's database. For each supported distributed backup product, there is at least one iSponsor. Certain supported products have multiple iSponsors, allocating one for each supported release. To retrieve data from a remote backup server, two components must be installed on each remote server to be accessed:

- The iGateway
- An appropriate iSponsor.

iSponsors and iGateway components may be downloaded free of charge from supportconnect.ca.com. For more information on the iSponsor/iGateway technology refer to the Product Home Page for CA Portal.

Method of Operation The target distributed servers are defined to DTS using control statements. To query a remote server's inventory, DTS dispatches a query task to establish a session with the target server. DTS identifies what iSponsors are running on a remote server and selects the appropriate iSponsor and method. DTS uses the selected method to query the backup server's database. The retrieved data is then used to construct a Distributed Tape Inventory File and the current inventory is compared to the previous inventory. DTS then updates the TMC to reflect any changes detected in that server's inventory. Validations are performed in a sequential order to reduce redundant updates. DTS logs query activity to a sysout file. The level of logging is controlled with the MSGLEVEL parameter. Execution of data retrieval sessions can be triggered periodically using the CTS scheduling component, CTSSCHD. Data retrieval sessions can also be triggered manually using a console command.

Volume Serial Number Ranges in the TMC

Volume serial ranges that are retrieved from the remote backup servers must be defined to CA 1 and must be marked as DTS tapes. Tapes are marked as DTS tapes by using TMSUPDTE to set FLAG5=40. DTS tapes may also be referred to as Agent tapes. The ranges should be used for distributed systems only, and should not be shared with mainframe applications.

Note: It is important that you use care and ensure that duplicate volume serial number ranges are *not* used by different distributed backup systems.

CTS proc changes

A SYSTCPD DD statement is required for TCP/IP services. For the correct data set name, refer to your TCP/IP installation.

A CTSOPTNS DD statement is required to access the *hlq*.CTAPOPTN data set. This data set contains the parameters for the DTS feature. The sample CTS procedure contains this DD as a comment. You may uncomment this DD, or supply your own. The format is:

```
//CTSOPTNS DD DISP=SHR,DSN=hlq.CTAPOPTN
```

Security Requirements

The DTS feature dynamically creates MVS data sets for temporarily storing remote server data. The data set name is controlled by the PREFIX= operand on the GLOBAL statement appended with the SRVDEF name itself. You may need to update your security system to provide the CTS started task appropriate authority to create these data sets. Verify with your security administrator that the USERID assigned to the CTS started task has authority to create the MVS data sets.

The DTS feature makes use of TCP/IP for communicating with the remote servers. The USERID associated with the CTS started task must have a valid OMVS segment defined to allow use of TCP/IP facilities. The USERID assigned to your TCP/IP started task may be used as a model to construct a security profile for the CTS started task. Consult with your mainframe security administrator for this requirement.

Startup Procedure

You can start DTS manually when CTS is started. However, the recommended method is to have CTS automatically start the DTS subtask.

Note: Starting the DTS subtask does not automatically initiate communication with the defined backup servers. Scheduling regular communication with the backup servers is accomplished using the CTSSCHD subtask.

Automatic Startup

To automatically start the DTS subtask each time CTS is started, add the following command to the CTSSTART member in *hlq.CTAPOPTN*:

```
SET TASK(DTS) PGM(CTSDTS) PARM(XX)
START DTS
```

XX

Specifies the suffix for the CTODTSxx member to use.

Default: 00

Manual Startup

Once the CTS task has been started, you can communicate with CTS by using a MODIFY command or by replying to the outstanding CTS WTOR. The following is an example of how to manually start the DTS subtask under CTS by using the modify command:

```
F CTS,SET TASK(DTS) PGM(CTSDTS) PARM(XX)
F CTS,START DTS
```

where *xx* is the suffix for the CTODTSxx member to use. The default is 00 if not specified.

DTS Control Statements Contained in CTODTS00

DTS reads a member from the *hlq*.CTAPOPTN library on startup to define the remote servers and establish certain global defaults. The default name of the member is CTODTS00. The last two characters can be changed, and the PARM parameter on the START command can be changed to reflect the new member name.

Control statements

Selectable DTS defaults are established with the GLOBAL control statement. Remote server definitions are established with the SRVDEF control statement.

All control statement keywords and operands must be uppercase.

Note: A comment line (a separate line beginning with '/'* and ending with '/'*) causes a multiple line control statement to terminate. Specify these comment lines before or after the control statements.

GLOBAL Control Statement

The GLOBAL control statement is coded as follows:

```
GLOBAL      TCPNAME=tcipname,  
            [MAXTASK=nn,],  
            [PREFIX=prefixname,]  
            [UNIT=unitname,]  
            [VOL=volser,]  
            [MSGLEVEL=nn,]  
            [STORCLAS=storageclassname,]  
            [DATACLAS=dataclassname,]  
            [MGMTCLAS=managementclassname,]  
            [TRACE/NOTRACE]
```

TCPNAME=

A required parameter that specifies the name of the TCPIP address space on the mainframe where CTS is running.

MAXTASK=

An optional parameter from 1 to 10 that specifies the maximum number of concurrent data retrieval operations that can be active at any one time. This specification is used to stifle resource use when multiple servers require data retrieval at the same time. All data retrieval sessions are queued up and serviced by the available data retrieval subtasks.

Default: 3

PREFIX=

An optional parameter that will form the name of the Distributed Tape Inventory File. The SRVDEF name is appended to this operand to form the fully qualified data set name.

Default: CAI.ISPON

UNIT=

An optional parameter that specifies a valid unit name that is used to dynamically allocate the distributed tape inventory file.

VOL=

An optional parameter that specifies the volume serial number to use when allocating the distributed tape inventory file.

MSGLEVEL=

An optional parameter that specifies the level of messaging that will occur. Valid values are 0 through 9. The default is 0 which lists only the most important messages. This can be changed with a console command during execution. Setting this to 9 provides supplementary information that can be used to diagnose abnormalities.

Default: 0

STORCLAS=

An optional parameter that specifies a valid SMS storage class to be used when allocating the distributed tape inventory file.

DATACLAS=

An optional parameter that specifies a valid SMS data class to be used when allocating the distributed tape inventory file.

MGMTCLAS=

An optional parameter that specifies a valid SMS management class to be used when allocating the distributed tape inventory file.

TRACE/NOTRACE

An optional parameter that displays all iSponsors found on a remote backup server when DTS connects to it.

Default: NOTRACE

SRVDEF Control Statement

The SRVDEF control statement is coded as follows:

```
SRVDEF      NAME=srvdefname,  
SERVER=servername,  
[SCHD=schdname, ]  
[LOG/NOLOG]  
[TRACE/NOTRACE]
```

NAME=

A name for this SRVDEF entry. The first character must be alphabetic. This name is used as the lowest level qualifier in constructing the data set name of the Distributed Tape Inventory File for this server. DTS creates these files as needed.

Range: 1 to 8 characters

SERVER=

The DNS name or IP address of the remote server running the backup product that you want to manage.

SCHD=

The name of the CTSSCHD schedule event set to trigger a query session for this server. For example, if SCHD=HOURLY is specified, you must ensure the SCHD component of CTS is started and contains a definition for HOURLY.

LOG/NOLOG

(Optional) Specifies whether print logging of received data is desired. If set to LOG, a sysout DD with the name of this SRVDEF is dynamically allocated.

Default: NOLOG

TRACE/NOTRACE

(Optional) Displays all iSponsors found on a remote backup server when DTS connects to it. If this parameter is specified, it overrides the specification in the GLOBAL specification.

Default: NOTRACE

Operator Commands

DTS can receive commands from the console via the modify command, or from the outstanding CTS WTOR. The commands are as follows:

Start Command

F CTS,START,DTS[, PARM(xx)]

This command starts the DTS task. If you specify DTS to be started in the CTS startup options, this command is not required. The optional parm parameter is used to specify a different suffix for the CTODTS member to read.

Stop Command

F CTS,STOP,DTS

This command stops the DTS task. Any SRVDEF definitions that are currently being processed will be allowed to complete.

MSGLEVEL Command

F CTS,MSG,DTS, 'MSGLEVEL=n'

This command changes the level of messaging of the DTS task. Valid values of *n* are 0 through 9, where 0 indicates that only critical error messages are to be displayed up to 9, which requests all detailed messages be displayed.

Start Processing Command

```
F CTS,MSG,DTS, ' START NAME=srvdefname '
```

This command starts a data query session for the SRVDEF entry *srvdefname*.

Start a group of SRVDEF's

```
F CTS,MSG,DTS, ' EVENT SCHD=schdname '
```

This command starts a data retrieval session for all SRVDEF definitions that have specified *schdname* as an operand of the SCHD parameter.

Start or stop displaying all discovered iSponsors

```
F CTS,MSG,DTS,TRACE or NOTRACE
```

This command initiates or suspends the display of iSponsors found on all SRVDEF definitions when a data retrieval session is started.

List the defined SRVDEF definitions

```
F CTS,MSG,DTS,LIST
```

This command produces a list of all SRVDEF definitions defined. The list is written to the DTSLIST DD statement.

Turn logging on or off for a SRVDEF definition

```
F CTS,MSG,DTS, [LOG] [NOLOG], NAME=srvdefname
```

This command turns logging on for the SRVDEF definition *srvdefname*. If you want to suspend logging for this SRVDEF, specify NOLOG in place of LOG.

Sample SRVDEF Definitions

The following is an example that defines one server. The server name is server1.yourdomain.com.

```
/*                                                    */
/*                                                    */
GLOBAL TCPNAME=TCPIP90,
      MAXTASK=2,
      UNIT=3390,
      PREFIX=CTS.ISPONSOR.DATA
SRVDEF NAME=SERVER1,
      SERVER=SERVER1.YOURDOMAIN.COM,
      SCHD=HOURLY
```


This example defines one server using the IP address of the server. The server IP address is 138.42.10.105.

```
/*                                     */
/*                                     */
GLOBAL TCPNAME=TCPIP90,
        MAXTASK=2,
        UNIT=3390,
        PREFIX=CTS.ISPONSOR.DATA
SRVDEF  NAME=SERVER1,
        SERVER=138.42.10.105,
        SCHD=HOURLY
```

CTS Scheduler (SCHD)

CTSSCHD is the scheduling component of CTS that sends commands to other CTS tasks at pre-defined times. Schedules are defined using control statements. Each TOD element that is defined begins with the SCHD verb and must have an 8-character name and a time of day value. You also have the option to specify the day of the week. The WHEN statement describes necessary actions to take when an event expires. Multiple TOD elements can be grouped together and assigned a single name to minimize coding requirements. This is accomplished using the GROUP operand in place of the TOD operand.

CTS Proc Changes

A CTSOPTNS DD statement is required to access the *hlq.CTAPOPTN* data set. This data set contains the parameters for the SCHD feature. The sample CTS procedure contains this DD as a comment. You may uncomment this DD, or supply your own. The format is:

```
//CTSOPTNS DD DISP=SHR,DSN=hlq.CTAPOPTN
```

Automatic Startup

To automatically start the SCHD subtask each time CTS is started, add the following commands to the CTSSSTART member in *hlq.CTAPOPTN*:

```
SET TASK(SCHD) PGM(CTSSCHD) PARM(XX)
START SCHD
```

XX

Specifies the suffix for the CTOSCH member to use.

Default: 00

Manual Startup

Once the CTS task is started, you can communicate with CTS by using a MODIFY command for WTOR. The following is an example of how to manually start the SCHED subtask under CTS using the MODIFY command:

```
F CTS,SET TASK(SCHED) PGM(CTSSCHED) PARM(XX)
F CTS,START SCHED
```

Where *xx* is the suffix for the CTOSCH member to use. If a number is not specified, the default used is 00.

SCHED Control Statements Contained in CTOSCH00

SCHED reads a member from the *hlq*.CTAPOPTN library on startup to define schedules, groups and actions. The default name of the member is CTOSCH00. The last two characters may be changed, and the PARM parameter on the START command can be changed to reflect the new member name.

Event Definition Statements

TOD events, event groups and associated actions are defined in the CTOSCH00 member of *hlq*.CTAPOPTN. All control statement keywords and operands must be uppercase. The control statements are as follows:

TOD Event Definition

A TOD event is defined using the SCHD statement as follows:

```
SCHD NAME=event . name ,  
TOD=hh . mm ,  
[ DOW=( MON , TUE , WED , THU , FRI , SAT , SUN ) ]
```

NAME=

Names this SCHD definition. The event.name can be any unique alphanumeric character string up to 24 characters in length. The first character must be alphabetic.

TOD=

Specifies the time of day this event is to occur. It is specified as hh.mm.

Range: 00.00 to 23.59

DOW=

(Optional) Specifies the day of the week this event should occur. If multiple days are specified, enclose them in parentheses. If omitted, the TOD event occurs once each day.

Valid values: MON,TUE,WED,THU,FRI,SAT,SUN.

Group Definition

TOD events defined with the SCHD command can be grouped together under a single name. This enables you to code a single action (WHEN) command to be issued when any of the TOD events within the group occur. To group a set of TOD events, the SCHD command is used with the GROUP operand:

```
SCHD NAME=group . name ,  
GROUP=( event . name , event . name , . . . )
```

NAME=

A required parameter that names this SCHD definition. The event.name can be any unique alphanumeric character string up to 24 characters in length. The first character must be alphabetic.

GROUP=

This parameter is used to group multiple event definitions together and assign them a single name. The event names must be enclosed in parentheses. There is a limit of 8 event names that can be specified for the GROUP operand. If more than 8 event names are required to make up a group, use another SCHD statement with the same NAME operand as this one and specify the additional event entries. This process can be repeated as many times as necessary.

Action Statement Definition

When a discrete event occurs or an event occurs that is a member of a group, an action for this occurrence is triggered. This action notifies the target CTS task of the event. A default notification template is used unless the MSG parameter is coded.

```
WHEN(name) NOTIFY(cts.taskname) [MSG='message to send']
```

WHEN

Specifies a previously defined TOD event or GROUP.

NOTIFY

Specifies the name of the CTS task to notify that the event has occurred. The task name is the same name defined in a SET TASK(...) initialization statement.

MSG

(Optional) Specifies the text sent to the specific CTS task when this event occurs. If a message is not specified, the default message 'EVENT SCHD=name' is sent to the CTS task where name is replaced with the name on the WHEN statement.

Sample Schedules

A simple schedule with one TOD event and one action statement

```
/*                                                    */
/* The following statements describe one event at 08:00 */
/* and an action statement that will notify the DTS subtask. */
/*                                                    */
SCHD NAME(EIGHTAM),TOD(08.00)
/*                                                    */
WHEN(EIGHTAM) NOTIFY(DTS)
/*                                                    */
/* End of sample */
/*                                                    */
```

A simple schedule with two TOD events and two action statements

```

/* */
/* The following statements describe one event at 08:00 */
/* and an action statement that will notify the DTS subtask. */
/* */
SCHD NAME(EIGHTAM),TOD(08.00)
SCHD NAME(TENAM),TOD(10.00)
/* */
/* */
/* At 8 am, notify DTS. */
/* At 10 am, notify the APEC task to start a scan. */
/* */
WHEN(EIGHTAM) NOTIFY(DTS)
WHEN(TENAM) NOTIFY(APEC) MSG=SCAN
/* */
/* End of sample */
/* */

```

A schedule with multiple TOD events, a grouping entry, and one action statement.

```

/* */
/* The following statements define TOD events for hours 09 */
/* through 15, a grouping statement, and an action statement. */
/* */
SCHD NAME(NINEAM),TOD(09.00)
SCHD NAME(TENAM),TOD(10.00)
SCHD NAME(ELEVENAM),TOD(11.00)
SCHD NAME(NON),TOD(12.00)
SCHD NAME(ONEPM),TOD(13.00)
SCHD NAME(TWOPM),TOD(14.00)
SCHD NAME(THREEPM),TOD(15.00)
SCHD NAME(FOURPM),TOD(16.00)
/* */
/* */
/* Group all of the above TOD definitions to a single name, */
/* "DAY_SHIFT" */
/* */
/* */
SCHD NAME(DAY_SHIFT),
    GROUP(NINEAM,TENAM,ELEVENAM,NON,ONEPM,TWOPM,THREEPM,FOURPM)

```

```
/* */
/* */
/* The next statement will send the text "EVENT SCHD(DAY_SHIFT)" */
/* to cts task DTS. */
/* */
/* */
    WHEN(DAY_SHIFT) NOTIFY(DTS)
/* */
/* */
/* End of sample */
/* */
/* */
/* */
```

A schedule that defines every hour of the day, multiple grouping statements, and several action statements.

Only one action statement is enabled. TOD specifications and group operands may not be continued. The group specifications are cumulative. You may repeat the group name when grouping multiple TOD events into a single group, as shown below.

```
/* */
/* The following statements describe events on each hour. */
/* The name of each event corresponds with the hour */
/* */
SCHD NAME(ONEAM),TOD(01.00)
SCHD NAME(TWOAM),TOD(02.00)
SCHD NAME(THREEAM),TOD(03.00)
SCHD NAME(FOURAM),TOD(04.00)
SCHD NAME(FIVEAM),TOD(05.00)
SCHD NAME(SIXAM),TOD(06.00)
SCHD NAME(SEVENAM),TOD(07.00)
SCHD NAME(EIGHTAM),TOD(08.00)
SCHD NAME(NINEAM),TOD(09.00)
SCHD NAME(TENAM),TOD(10.00)
SCHD NAME(ELEVENAM),TOD(11.00)
SCHD NAME(NOON),TOD(12.00)
SCHD NAME(ONEPM),TOD(13.00)
SCHD NAME(TWOPM),TOD(14.00)
SCHD NAME(THREEDPM),TOD(15.00)
SCHD NAME(FOURPM),TOD(16.00)
SCHD NAME(FIVEPM),TOD(17.00)
SCHD NAME(SIXPM),TOD(18.00)
SCHD NAME(SEVENPM),TOD(19.00)
SCHD NAME(EIGHTPM),TOD(20.00)
SCHD NAME(NINEPM),TOD(21.00)
SCHD NAME(TENPM),TOD(22.00)
SCHD NAME(ELEVENPM),TOD(23.00)
SCHD NAME(MIDNIGHT),TOD(00.00)
```

```

/* */
/* The following statements group the events into several groups.*/
/* The first group is the night_shift group, with entries t1 */
/* through 8. */
/* */
SCHD NAME(NIGHT_SHIFT),
    GROUP(ONEAM,TWOAM,THREEAM,FOURAM,FIVEAM,SIXAM,SEVENAM,EIGHTAM)
/* */
/* The next group, day_shift, describes the hours 9 through 16. */
/* */
SCHD NAME(DAY_SHIFT),
    GROUP(NINEAM,TENAM,ELEVENAM,NOON,ONEPM,TWOPM,THREEPM,FOURPM)
/* */
/* Finally, the evening_shift group describes hours 17 through */
/* 24. The "EVENING_SHIFT" name is repeated in order to specify */
/* the additional entries. */
/* */
SCHD NAME(EVENING_SHIFT),
    GROUP(FIVEPM,SIXPM,SEVENPM,EIGHTPM)
SCHD NAME(EVENING_SHIFT),
    GROUP(NINEPM,TENPM,ELEVENPM,MIDNIGHT)
/* */
/* The next statements describe a group that has all hourly */
/* entries in it. You can only specify 8 entries on the */
/* group statement, so the entire hourly SCHD NAME(HOURLY) */
/* statement is repeated, each having the next 8 hourly */
/* entries in it. */
/* */
SCHD NAME(HOURLY),
    GROUP(ONEAM,TWOAM,THREEAM,FOURAM,FIVEAM,SIXAM,SEVENAM,EIGHTAM)
/* */
/* Continue specifying the hourly group with the next 8 hours */
/* in it. */
/* */
SCHD NAME(HOURLY),
    GROUP(NINEAM,TENAM,ELEVENAM,NOON,ONEPM,TWOPM,THREEPM,FOURPM)
/* */
/* Continue specifying the hourly group with the next 8 hours */
/* in it. */
/* */
SCHD NAME(HOURLY),
    GROUP(FIVEPM,SIXPM,SEVENPM,EIGHTPM)
SCHD NAME(HOURLY),
    GROUP(NINEPM,TENPM,ELEVENPM,MIDNIGHT)

```

```
/* */
/* The next statement will send the message: EVENT SCHD(HOURLY) */
/* to CTS task DTS. Since each TOD event is defined to HOURLY, */
/* a message will be sent at the start of each hour. */
/* */
/* */
WHEN(HOURLY) NOTIFY(DTS)
/* */
/* */
/* The format of a command to send a message for each event */
/* in the day shift would be: */
/* */
/* WHEN(DAY_SHIFT) NOTIFY(DTS) */
/* */
/* */
/* The format of a command to send a message for each event */
/* in the night shift would be: */
/* */
/* WHEN(NIGHT_SHIFT) NOTIFY(DTS) */
/* */
/* */
/* The format of a command to send a message for each event */
/* in the evening shift would be: */
/* */
/* WHEN(EVENING_SHIFT) NOTIFY(DTS) */
/* */
/* */
/* End of sample */
/* */
/* */
/* */
```


CA 1 SMF Queue (SMFQ)

Use the CTS SMFQ subtask if you process hundreds of tape jobs concurrently. SMFQ improves the performance of the TMC update processing for z/OS catalog events.

When SMFQ starts, CA 1® Tape Management does the following:

- Obtains storage for an in-memory (ECSA) queue.
- The CA 1® Tape Management SMF exit (IEFU83 exit) checks for the availability of the queue.
- The CA 1® Tape Management SMF exit places catalog and uncatalog requests in the queue.
- The SMFQ subtask processes the requests in the queue.

This results in more efficient processing of catalog and uncatalog activity because of the reduced RESERVE activity against the TMC and z/OS catalog.

To enable this feature start the new subtask; no additional system options are required.

Before starting SMFQ, ensure that the following conditions are met:

- TMSINIT has started and CA 1® Tape Management is either ACTIVE or BATCH ACTIVE.
- The CTS started task has UPDATE level access to the YSVCUNCD resource if YSVC is set to YES or EXT.

Startup Procedure

TMSSMFQ can be started manually or automatically when CTS is started. We recommend that CTS start the TMSSMFQ subtask automatically.

To automatically start the TMSSMFQ subtask each time CTS is started, add the following command to the CTSSTART member in CA1.CTAPOPTN:

```
SET TASK(SMFQ) PGM(TMSSMFQ)
START SMFQ
```

When the CTS task has started, you can communicate with CTS with a MODIFY command or by WTOR.

Example: Start TMSSMFQ manually

This example uses the MODIFY command to start the TMSSMFQ subtask manually under CTS:

```
F CTS,SET TASK(SMFQ) PGM(TMSSMFQ)
F CTS,START SMFQ
```

SMFQ Commands

Commands are sent to SMFQ using the CTS Message command (MSG). Commands can be issued at any time while TMSSMFQ is active.

Use the following format on all commands issued to the SMFQ subtask:

F CTS,MSG SMFQ, [STOP|CANCEL|STATUS]

STOP

Use the STOP command to request SMFQ to process any remaining SMF records and terminate.

CANCEL

Use the CANCEL command to request SMFQ to cancel itself with a User-111 Abend.

SMFQ will not process the remaining SMF records. To process these records, the operator must restart SMFQ.

STATUS

Use the STATUS command to request SMFQ to provide the current status of CA 1® Tape Management. This generates the following:

- The current preprocessing and post processing statistics
- The number of SMF records waiting to be processed
- The total number of SMF records processed since the last IPL

Note: Preprocessing statistics refer to the time required to add an SMF record to the queue and to prepare that record for processing once it is removed from the queue. Postprocessing statistics refer to the time required to add an SMF record to the queue and to then process the record. Use the difference between these two times to determine the time required to process an SMF record.

Tape Map (TMAP)

The Tape Map (TMAP) task provides a convenient way to list the contents of an ANSI Label (AL) or Standard Label (SL) tape. The task dynamically allocates one or more tape drives. The task mounts the requested volumes to produce a report of the files on each AL or SL tape. The tape drive is released after the requested tapes are processed. TMAP remains active and available for more tape map requests until it terminates by the STOP command.

You can run the CTSTMAP program as a batch job independent of the CTS procedure. The JCL to run CTSTMAP as a batch job is provided in *hlq.CTAPJCL* member CTSJTMAP.

Notes:

- For more information about executing CTSTMAP as a batch job, see the instructions in the CTSJTMAP member.
- BLP authorization (BLP=YES) is required for Batch and STC (started task).

Startup Procedure

TMAP can be started manually or automatically when CTS is started. Starting the TMAP subtask manually is recommended.

Start the TMAP Subtask Automatically

To have the TMAP subtask started each time CTS is started, set the subtask to start automatically.

To automatically start the TMAP subtask, add the following commands to the CTSSTART member in *hlq.CTAPOPTN*:

```
SET TASK(TMAP) PGM(CTSTMAP)
START TMAP
```

Start the TMAP Subtask Manually

To start the TMAP subtask yourself, do not set it to start automatically and start it manually each time.

Once the CTS task has been started, you can communicate with CTS by using a MODIFY command or by WTOR. The following is an example of how to start manually the TMAP subtask under CTS by using the MODIFY command:

```
F CTS,SET TASK(TMAP) PGM(CTSTMAP)
F CTS,START TMAP
```

TMAP Commands

Commands are sent to TMAP using the CTS Message command (MSG). Use the following format on all commands that are issued to the TMAP subtask.

```
STOP TMAP
CANCEL TMAP
F CTS,MSG TMAP, 'MAP volser1,...,volsern,UNIT=uuuu'
```

volser

Specifies the volume serial number of a tape that you want to map.

Range: 1 to 8

uuuu

Specifies the generic unit or unit address for the tape drive required.

Default: TAPE

You can use the following commands:

STOP

Requests TMAP to close all files and terminate.

CANCEL

Requests TMAP to cancel itself with a User 111abend.

MAP

Maps the tape and report on the files it contains.

Command Sample

```
| F CTS,MSG TMAP, 'MAP 101001,102033,534006,UNIT=3590' |
```

When you enter the preceding command, the TMAP subtask of the CTS address space allocates a 3590 tape unit. The TMAP subtask mounts the three volumes that you requested. After producing a tape map for each volume, the tape unit is deallocated. This example assumes that the TMAP subtask has already been started, either automatically or manually.

Volume Pool Monitor (VPM)

The CTSVPM subtask of the Common Tape System provides a convenient way to monitor availability of scratch tapes in user-defined volume pools. The Volume Pool Monitor (VPM) subtask uses volume pools, variables, and alerts that can trigger WTOs and emails. You define all of the items through the ISPF Volume Pool panels and save them in the VSAM Database (VDB).

When started, the VPM subtask reads the VDB to identify the following elements that are defined through CTSISPF panels:

- Volume pools
- Variables that CA 1 creates automatically or user-defined variables
- Alerts to trigger customizable emails or WTOs
- Variables to control VPM subtask processing

The VPM subtask builds tables for each of the preceding items. The VPM subtask keeps the items maintained as new volume pools, variables, and alerts are created through ISPF or ongoing processing. VPM then passes the TMC and develop counts for the defined volume pools, placing volumes in a category of Active, Scratch, and Out of Service. A total volume count for the volume pool and a separate count for Never Used volumes is maintained. The Active, Scratch, and Out of Service counts add up to the total number of volumes in the volume pool.

The CA 1 Audit file monitors the tape activity. As tapes change status, the accumulators update to reflect a status change of a volume. For example, the status can change from scratch to active, or from active to scratch. When you shut down the VPM subtask, either individually with a STOP command or by stopping the CTS task, the in-memory volume pool table is written to the VDB. The in-memory is also written periodically while VPM is active, as the OPT_VPM_TIMER variable defines.

Alerts that you defined through ISPF allow you to create a condition to test and what action to take when the condition is met. When the condition that is defined in an alert is met, you can either issue a customized WTO or an email. An example is to send an email to the tape librarian when a volume pool has fewer than 500 scratch volumes available.

The VPM subtask also creates volume pool history records to allow you to detect trends in your scratch availability. Both daily and weekly volume pool history records are created. Both types of history records are written to the VDB when the VPM subtask first wakes up on a new day. The subtask typically wakes up just after 12:00 midnight. These records can be viewed using the CA GMI user interface, which ships as part of CA 1.

No changes to the CTS PROC are required for the VPM subtask. The CTSVPM subtask dynamically allocates the VDB when the task starts.

VPM Prerequisites

Before you start the VPM subtask, create the VDB and customize the subtask and email variables. You update the variables using the CTS ISPF application, so install the CTS ISPF elements too.

Allocate the SEND data set and load it with the sample WTOs and emails provided. Jobs to allocate and initialize both the VDB and SEND data sets are provided in the *hlq.CTAPJCL* data set.

Note: For more information about the allocate and initialize jobs, see the *Programming Guide*.

More information:

[Using Online Inquiry and Update Facilities](#) (see page 95)

[VPM Subtask Variables](#) (see page 256)

[VPM Email Variables](#) (see page 257)

VPM Subtask Variables

The VPM subtask is controlled through variables entered in the CTS ISPF panels. These variables are set when the task comes up. You can change the variables while the VPM subtask is running. The VPM subtask checks if the variables have been changed since the last time the OPT_VPM_TIMER interval has expired. If the variables controlling VPM have changed, the new variables are used.

OPT_VPM_DISPLAY_WTO_MAX

Controls the maximum number of lines output when the **DISPLAY POOL** or the **STATUS** command is issued in the VPM subtask.

Default: 20 lines

OPT_VPM_RETAIN_HISTORY_DAILY

Controls how many days VPM DAILY History records are kept before they VPM subtask automatically deletes them. Daily History records are provided to allow CA Vantage and CA GMI to produce trend reports.

Default: 90 days

OPT_VPM_RETAIN_HISTORY_WEEKLY

Controls how many days VPM WEEKLY History records are kept before the VPM subtask automatically deletes them. Like the Daily History records, the Weekly records are provided to allow CA Vantage and CA GMI to produce trend reports for a longer time.

Default: 365 days

OPT_VPM_TIMER

Controls the interval at which the VPM subtask checks for activity from the Audit file or changes in the volume pools, variables, and alerts that are defined in ISPF. This variable is specified in hh:mm:ss format. To set the interval to perform these checks every 15 minutes, enter 00:15:00.

Default: 30 minutes (00:30:00)

VPM Email Variables

The VPM subtask can issue emails and WTOs in response to alerts that have been entered through the ISPF panels that are associated with VPM. To issue emails, provide information about your email environment and default values for emails that VPM issues. Before you trigger any alerts through VPM, update the following email variables through the CTS ISPF panels. To update any of these variables, use the **U** line command.

You can change the variables anytime. The VPM subtask checks if the variables have been changed since the last time the OPT_VPM_TIMER interval expired. If the variables controlling VPM emails have changed, the new variables are used.

OPT_EMAIL_DEFAULT_DSN

Identifies the partitioned data set that contains the WTO or email text to create emails or WTOs. This variable is the full data set name of a fixed block (FB) 80 character (LRECL) partitioned data set. Each member of this data set defines a single WTO or email that is issued when a VPM alert is triggered. The member name is specified in the Send field of the alert definition in ISPF. The VPM subtask processes the members. Configure the variable before you run VPM. The entered value is converted to uppercase.

Default: CAI.SEND

OPT_EMAIL_DEFAULT_FORMAT

Defines the default format for the body of emails created. The options are TEXT and HTML. In TEXT format, extra blanks are removed from the input records and the text is reformatted on fewer lines. HTML format can include HTML formatting tags in the email body. You can define a different format using the x-FMT: control statement.

Default: TEXT

OPT_EMAIL_DEFAULT_FROM

Identifies the "from" address on any emails that the VPM subtask generates. Configure this variable before you run VPM.

Default: SYSPROG@YOUR.SITE

OPT_EMAIL_DEFAULT_SUBCHAR

Defines the default substitution character that is used to delimit variables in email fields for variable substitution. You can define a different substitution character for an individual email using the x-SUB: control statement.

Default: &

OPT_EMAIL_DEFAULT_TITLE

Specifies the title of all emails that the VPM subtask generates. You can provide a more specific title in the email member itself with the TITLE: control statement.

Default: EMAIL FROM CA TAPE MANAGEMENT

OPT_EMAIL_DEFAULT_TO

Defines the default recipient of emails that the VPM subtask generates. You can define different recipients using the TO: control statement. Configure this variable before you run VPM.

Default: TAPEPROG@YOUR.SITE

More information:

[CA 1 ISPF Facilities](#) (see page 95)

Email Environment Setup

Create three user variables for each system where emails are sent from. A naming convention that includes the SMF ID of the systems where emails are sent from has been established. The three user variables are as follows:

EMAIL_XXXX_HOSTNAME

EMAIL_XXXX_SMTP

EMAIL_XXXX_SYSOOUT

Replace xxxx in the user variables with the SMF ID of the systems where emails are sent from. You can send emails by running the CTSMAIL utility or when the VPM subtask triggers an email as the result of an alert.

EMAIL_XXXX_HOSTNAME

Defines the host name for TCP/IP where emails are sent from. Create this user variable before you run VPM. You find the host name when you issue the HOMETEST TSO command. The host name returns in the following message:

TCP Host Name is: *yourhost*

EMAIL_XXXX_SMTP

Identifies the name of the SYSOOUT writer for the SMTP PROC. Create this user variable before you run VPM.

Default: SMTP

EMAIL_XXXX_SYSOOUT

Defines the SYSOOUT CLASS used to send emails with SMTP. Create this user variable before you run VPM.

Default: B

Note: SMTP must be active in your system. On most systems, the procedure is called SMTP and can be started with the console command **S SMTP**.

Email Environment Setup Example

A site has two LPARs with SMF ID names of SYP1 and SYP2. They have chosen to use the CA 1 email capabilities provided in the VPM subtask and the CTSMAIL utility. Use the ISPF CTS Variable Add panel to create six user variables to set up their environment (three for each system).

EMAIL_SYP1_HOSTNAME

EMAIL_SYP1_SMTP

EMAIL_SYP1_SYSOUT

EMAIL_SYP2_HOSTNAME

EMAIL_SYP2_SMTP

EMAIL_SYP2_SYSOUT

You can use examples of these user variables with xxxx in the place of the SMF ID for models. Do not update the examples. Create new user variables.

Start the VPM Subtask Automatically

To have the VPM subtask started each time CTS is started, set the subtask to start automatically.

To automatically start the VPM subtask, add the following commands to the CTSSTART member in *hlq.CTAPOPTN*:

```
SET TASK(VPM) PGM(CTSVPM)
START VPM
```

Start the VPM Subtask Manually

To start the VPM subtask yourself, do not set it to start automatically and start it manually each time.

Once the CTS task has been started, you can communicate with CTS by using a MODIFY command or by WTOR. The following is an example of how to start manually the VPM subtask under CTS by using the MODIFY command:

```
F CTS,SET TASK(VPM) PGM(CTSVPM)
F CTS,START VPM
```

Operator Commands

You can use the following operator commands:

DISPLAY Command

```
F CTS,MSG,VPM,'DISPLAY P'  
F CTS,MSG,VPM,'D P'  
F CTS,MSG,VPM,'D POOL=PRODPPOOL'  
F CTS,MSG,VPM,'D POOL=N*'
```

Displays the current counts for all volume pools in the following format:

```
F CTS,MSG,VPM,'D POOL=N*'  
      TO VPM D POOL=N*  
CA$F510I CTSVPM DISPLAY COMMAND RECEIVED: D,POOL=N*  
CA$F511I -----Volume Pool Name----- Type=REG  
CA$F512I --Total--- --Active-- --Scratch- Out-of-Srv  Never-Used  
CA$F513I NEWPOOL 2  
CA$F514I 99,999,999 99,999,999 99,999,999 99,999,999 99,999,999  
CA$F513I New STK T10000K Range  
CA$F514I 99,999,999 99,999,999 99,999,999 99,999,999 99,999,999
```

In the preceding example, all volume pools that begin with N are displayed up to the maximum number of lines. The OPT_VPM_DISPLAY_WTO_MAX variable controls this number.

You can also request the type of counts for these pools using the additional TYPE= parameter. These types are regular, low, and high values:

```
F CTS,MSG,VPM,'D P=N*,TYPE=LOW'  
F CTS,MSG,VPM,'D P=N*,TYPE=HIGH'  
F CTS,MSG,VPM,'D P=N*,TYPE=REG'
```

LOW

Gives the lowest values that the pool encounters.

HIGH

Gives the highest values that the pool encounters.

REG

Gives the regular value, or the current count.

If type is not specified, the command assumes TYPE=REG and displays the current count.

You can also view this information using the CTS ISPF panels VPM Pool Detail panel or VPM Range Detail panel for a specific range.

STATUS Command

```
F CTS,MSG,VPM,STATUS  
F CTS,MSG,VPM,S
```

Displays the status of the CTSVPM in the following format:

```

F CTS,MSG,VPM,S
  TO VPM S
CA$F510I CTSVPM COMMAND RECEIVED: STATUS
CA$F517I CTSVPM Subtask Status Display
CTSVPM is Active, awaiting command
*** CA 1 is batch active ***
CTSVPM maintenance level is T5Q3220 created on 11/29/xx 14.49
CTSVPM Timer last expired at:      20xx/341 16:11:21:40
CTSVPM Last Audit scan completed: 20xx/341 16:11:24:14
CTSVPM Parameters in effect
  OPT_VPM_DISPLAY_WTO_MAX          999
  OPT_VPM_RETAIN_HISTORY_DAILY     1,000
  OPT_VPM_RETAIN_HISTORY_WEEKLY    5,000
  OPT_VPM_TIMER                    00:15:00
CTSVPM Summary information
Alerts          5      Last Refreshed: 20xx/341 15:40:58:28
Variables       76      Last Refreshed: 20xx/341 16:01:22:55
Volume Pools    8      Last Refreshed: 20xx/340 12:36:22:79
CTSVPM Alerts currently being monitored
My-Alert-1
My-Alert-2
My-Alert-3
My-Alert-4
My-Alert-5
CA$F518I CTSVPM End of Status Display

In the preceding example, all of the alerts display up to the maximum number of
lines as the OPT_VPM_DISPLAY_WTO_MAX variable controls.

```

REBUILD Command

```
F CTS,MSG,VPM,REBUILD
```

Causes VPM to tear down existing volume pools, alerts, and variable tables and to recreate them using the most current definitions in the VDB. After the volume pools, alerts and variable tables have been built, VPM passes the TMC and develops counts for the defined volume pools. VPM places the volumes in a category of Active, Scratch, and Out of Service. To limit the use of CPU and IO that are associated with the scan of the TMC, use this command sparingly.

REFRESH Command

```
F CTS,MSG,VPM,REFRESH
F CTS,MSG,VPM,R
```

Requests that VPM checks to see if new alerts, variables, or volume pools (or volume pool ranges) have been entered in ISPF. If new or updated alerts, variables, or volume pools are found, the in-memory versions of these tables are updated. If the CTS VPM subtask option variables are updated, the new values are used when those variables are next referenced (after the current OPT_VPM_TIMER interval has expired or after you issue an UPDATE command).

Use care when defining new volume pools or new volume pool ranges because they can trigger alerts that are associated with percentage value conditions. The counters that are associated with a new volume pool or volume pool range are correct only after the VPM subtask stops and starts. Or the counters are correct after the REBUILD command is issued. These actions cause the TMC to be rescanned and all volume pool counters rebuilt.

START Command

F CTS,START,VPM

Starts the VPM subtask. The VDB is read to create the volume pools, variables, and alerts that you defined. The TMC is read and analyzed to update the in-memory volume pool table. Monitoring of volume status initiates.

STOP Command

F CTS,STOP,VPM

Stops the VPM subtask. The volume pools that are defined through the ISPF interface are monitored only after the VPM subtask is started again. You can start the VPM subtask either manually by the modify command or by restarting the CTS task. No history records are written while the VPM task is stopped.

UPDATE Command

F CTS,MSG,VPM,UPDATE

F CTS,MSG,VPM,U

Causes the VPM subtask to preempt the timer and check the alerts, variables, or volume pools (or volume pool ranges) definitions for any changes through ISPF. The timer loop that the OPT_VPM_TIMER value controls is reset and a new timer interval is checked for. The VPM subtask updates any alerts, variables, or volume pools that have been changed through ISPF. The CA 1 Audit file is scanned to identify any volume status change activity.

Chapter 11: Using Volume Pool Monitor Alert and Email Facilities

This section contains the following topics:

[Volume Pool Monitor Usage and Components](#) (see page 264)

[Volume Pools](#) (see page 265)

[Volume Pool Alerts](#) (see page 265)

[Variables](#) (see page 266)

[Create a WTO](#) (see page 276)

[Create an Email](#) (see page 278)

Volume Pool Monitor Usage and Components

The Volume Pool Monitor (VPM) feature is customizable and provides notification capabilities that are integrated with CA 1 daily processing. These capabilities are implemented using z/OS standard components such as ISPF and the CTS started task.

The VPM feature is optional and does not change any TMC or Audit records. The existing Audit Type 5 record passes data from various CA 1 daily utilities (TMSCOPY, TMSCLEAN, and TMSVMUPD) to the VPM subtask of CTS. You can include this data in the emails and WTOs that VPM generates.

The VPM feature enables you to do the following actions:

- Define pools of tapes to monitor tape activity
- Track tapes in active, scratch, or out of service categories
- Track scratch tape usage over time
- Create rules to generate emails or WTOs when specific conditions are met.
- Track the number of tape volumes in Never Used status
- Create batch jobs to create emails with data variables from CA 1 utilities that are run daily
- Retain daily and weekly historical records with volume pool details
- Create user variables for extensive customization of emails and WTOs

VPM provides this new capabilities through the integration of multiple components:

- VPM subtask of CTS
- VSAM Database (VDB)
- CTS ISPF Panels
- CTSMAIL utility program
- CAI.SEND data set
- CA GMI object displays for volume pools and their associated historical records

Note: For more information about the VDB and CAI.SEND data sets, see the *Programming Guide*. For more information about the CTSMAIL program, see the *Utilities and Reports Reference Guide*.

Volume Pools

A volume pool consists of one or more volume ranges. You can view the data in a volume pool by specific volume range or you can view all ranges for a volume pool summarized together. The VPM subtask processes the range definitions and determines how many volumes are found in each defined range and their status. Each volume can be in Active, Scratch, or Out of Service status. VPM monitors status changes of each volume. VPM examines the CA 1 Audit file for updates that change the status of a volume and it updates the volume range counters in real time. These counters are then written back to the VDB. You create the volume pools through the CTS ISPF application.

Use ISPF to view the volume pool status. You can also use the DISPLAY POOL command of VPM to show the higher-level pool information on the z/OS console. The following example shows a volume pool named STK_High_Capacity with three ranges defined:

Volume Pool Name	Volume Range
STK_High_Capacity	210000 – 219999
STK_High_Capacity	800000 – 899999
STK_High_Capacity	STK000 – STK999

VPM uses the volume pool name alone to assign all three ranges to the same volume pool STK_High_Capacity. The volume pool names are case-sensitive. STK_High_Capacity is a different volume pool than STK_HIGH_CAPACITY. VPM counts and classifies all volumes in the three ranges separately. To see the range information separately, enter **R** on the Volume Pool List ISPF display for the volume pool that you want to view. To see the combined total counts for all ranges in the volume pool, enter **S** on the Volume Pool List ISPF display. You can also use the DISPLAY POOL command of the VPM subtask.

More information:

[CTS ISPF Facilities](#) (see page 96)

Volume Pool Alerts

A volume pool alert is a condition that you define to be monitored by the VPM subtask. If the condition is met, you can cause an email to be sent or a WTO to be issued.

You create the volume pool alerts through the CTS ISPF application.

More information:

[CTS ISPF Facilities](#) (see page 96)

Variables

You can customize the VPM using variables. VPM supports four different types of variables:

User

Variables that the user can enter to customize emails and WTOs.

Option

Variables to control the VPM subtask of CTS and how email is configured on your system.

Product

Variables that CA 1 utilities update that you can use in your emails and WTOs.

System

System variables that describe the systems where CA 1 is installed.

The CTS VDB task and several CA 1 utilities create the variables. You can also create your own user variables. All variable types are saved in the VDB. The option, product, and system variables are created with defaults when the VDB is first initialized using the CTSVDBU utility. The variables are then updated with current information from the most recent run of CA 1 utilities. The VDB task also updates selected variables that are associated with alert processing being triggered.

More information:

[Variable Record Maintenance](#) (see page 125)

Variable Substitution

Both emails and WTOs support variable substitution. You can specify all types of variables – user, option, product, and system variables. Every line in the body of an email or WTO is examined for variables. Variables must begin and end with the substitution character (default is &). A variable must begin and end on the same line. If you do not specify a substitution character in the email header or WTO using the x-SUB: field, the variable uses the value that is defined in OPT_EMAIL_DEFAULT_SUBCHAR. Emails also allow the specification of variables in the header fields.

When a variable is found in an email or WTO, the VDB is searched for an exact match on the variable name. If the exact match is found, the value that is associated with that variable replaces the variable name. If the variable is not found in the VDB, the variable name stays the same in the email or WTO.

Variable names are case-sensitive. If the case does not match exactly between the variable in the email or WTO and the variable in the database, they are considered to be different variables. For example, &Thisvar& is not the same as &THISVAR&.

User and Option Variables

To make your emails and WTOs more meaningful for your site, you can specify user variables and option variables. Option variables control the VPM subtask and configure email support for your environment.

To more completely customize your emails and WTOs, you can create user variables. For example, you can reference a user variable On_Call_Manager in an email. You can create or update the variable On_Call_Manager with the name of your on-call manager each month. The name appears in emails or WTOs in place of the variable name. User variables can have any name except for the following restricted prefixes: OPT-, CTS-, SYS-, and TMS-. Variables are case-sensitive and must match exactly the variable in ISPF.

More information:

[VPM Subtask Variables](#) (see page 256)

[VPM Email Variables](#) (see page 257)

Product Variables

Selected CA 1 utilities create product variables for use in email and WTO customization. These utilities include TMSCLEAN, TMSCOPY, and TMSVMUPD. These utilities update the product variables by writing them to Audit file Type 5 (Exception) records, which the VPM subtask reads. VPM makes them available to alert processing and also writes them to the VDB. VPM uses two type 5 audit records:

011

Specifies the volume pool monitor record that is created to update volume status counters.

012

Specifies the volume pool monitor record that is created to update product variables that various CA 1 utilities create.

Note: You can print the Type 5 records with the TMSAUDIT utility. For more information about TMSAUDIT, see the *Utilities and Reports Guide*.

You can use the product variables to send emails or WTOs to key personnel supporting your tape environment, informing them of CA 1 processing. For example, you can create an email to send to the tape librarian. The email informs them that the TMSCLEAN utility has run and nn tapes were updated to scratch status. You reference the TMSCLEAN_CURRENT_SCRATCH_TAPES product variable.

Product Variables Created by TMSCLEAN

The following variables are updated every time the TMSCLEAN utility runs. These values can also be found in Report 85. If the program runs using the TEST parameter, these parameters are not updated.

TMSCLEAN_COND_CODE

Specifies the completion code from the last run of TMSCLEAN.

TMSCLEAN_CURRENT_SCRATCH_TAPES

Specifies the total number of scratch tapes that TMSCLEAN found.

TMSCLEAN_FILES_UNCATALOGED

Specifies the total number of data sets that TMSCLEAN uncataloged.

TMSCLEAN_JOB_COMPLETION_DATE

Specifies the completion date of the last run of TMSCLEAN.

TMSCLEAN_JOB_COMPLETION_TIME

Specifies the completion time of the last run of TMSCLEAN.

TMSCLEAN_TAPES_SCRATCHED

Specifies the total number of tape volumes that TMSCLEAN scratched.

TMSCLEAN_UNEXPIRED_TAPE_VOLUMES

Specifies the total number of unexpired tape volumes that TMSCLEAN found.

Product Variables Created by TMSCOPY

The TMSCOPY utility updates the following variables every time the TMSCOPY utility runs. These values are the same values as are reported in Report 31. If the program is run using the TEST parameter, these parameters are not updated.

TMSCOPY_ACTIVE_DSNBS

Specifies the total number of active or used DSNB records that TMSCOPY backed up.

TMSCOPY_AUDIT_RECORDS

Specifies the total number of audit records that TMSCOPY found in the TMC.

TMSCOPY_AUDIT_RECORDS_BACKED_UP

Specifies the total number of audit records that TMSCOPY backed up.

TMSCOPY_AVAILABLE_DSNBS

Specifies the total number of unused or available DSNB records that TMSCOPY backed up.

TMSCOPY_COND_CODE

Specifies the completion code from the last run of TMSCOPY.

TMSCOPY_DATE_OF_LAST_BACKUP

Specifies the date that TMSCOPY backed up the TMC.

TMSCOPY_DSNB_RECORDS_BACKED_UP

Specifies the total number of DSNB records that TMSCOPY backed up.

TMSCOPY_JOB_COMPLETION_DATE

Specifies the completion date of the last run of TMSCOPY.

TMSCOPY_JOB_COMPLETION_TIME

Specifies the completion time of the last run of TMSCOPY.

TMSCOPY_TIME_OF_LAST_BACKUP

Specifies the time that TMSCOPY backed up the TMC.

TMSCOPY_TMC_RECORDS_BACKED_UP

Specifies the total number of volume records that TMSCOPY backed up.

TMSCOPY_VOLUME_SERIAL_NUMBER

Specifies the tape volume serial number that the TMC backup is on.

Product Variables Created by TMSVMUPD

The following variables are updated every time the TMSVMUPD utility runs. If the program runs using the TEST parameter, these parameters are not updated.

TMSVMUPD_COND_CODE

Specifies the completion code from the last run of TMSVMUPD.

TMSVMUPD_JOB_COMPLETION_DATE

Specifies the completion date of the last run of TMSVMUPD.

TMSVMUPD_JOB_COMPLETION_TIME

Specifies the completion time of the last run of TMSVMUPD.

TMSVMUPD_TAPES_FROM_OFF_SITE

Specifies the total number of tape volumes returning from an offsite location. This value is new and is not on any report.

TMSVMUPD_TAPES_TO_OFF_SITE

Specifies the total number of tape volumes moved to an offsite location. This value is new and is not on any report.

TMSVMUPD_TAPES_MOVED

Specifies the total number of tape volumes that changed vault locations. You can find this value in Report 26 in the field TMC RECORDS UPDATED too.

Product Variables for Alerts

The VPM subtask updates the following product variables when an alert is triggered. They allow you to create emails and WTOs showing you the values in the alert that caused it to trigger. These product variables are updated in the VDB and always reflect the most recent alert that was triggered.

CTS_Alert_Interval

The value is taken from the Interval field on the VPD Alert Detail Panel. Valid intervals are ONCE, WHILE, or a number of hours (1-99).

CTS_Alert_ID

The value is taken from the Alert ID field on the VPD Alert Detail Panel. This value is basically the name that the user chose for the alert.

CTS_Alert_Operator

The value is taken from the Oper field on the VPD Alert Detail Panel. Valid Intervals are LT, LE, EQ, GE, or GT.

CTS_Alert_Pool_ID

The value is taken from the Pool ID field on the VPD Alert Detail Panel. If the alert is associated with a volume pool, the field contains the name of the volume pool that was triggered. If the alert is for a user variable, this field is blank.

CTS_Alert_Send

The value is taken from the Send field on the VPD Alert Detail Panel. This field contains the name of the email or WTO member in CAI.SEND that is issued when the alert is triggered.

CTS_Alert_Value

The value is taken from the Value field on the VPD Alert Detail Panel. This value is constant and it is the value that the CTS_Alert_Variable is compared to.

CTS_Alert_Variable

The value is taken from the Variable field on the VPD Alert Detail Panel. If the alert is for a volume pool, the field contains the name of the counter or percentage field that is being tested against. For volume pool alerts, the permitted values are:

Active_Count

Active_Percent

Never_Used_Count

Never_Used_Percent

Out_of_Service_Count

Out_of_Service_Percent

Scratch_Count

Scratch_Percent

If the alert is testing a numeric user variable, the user variable name displays here.

CTS_Alert_Variable_Value

The VPM subtask calculates this value. The value reflects the actual value of the tested field that triggered the alert. For example, if an alert is created to monitor the number of scratch tapes in a volume pool (Scratch_Count LT 500), the CTS_Alert_Variable_Value is the number of scratch tapes in the volume pool (499 or fewer).

System Variables

System variables are provided to describe basic information about the systems where CA 1 components are running. The values for the system variables are updated with current system information any time they are referenced (used in an email or WTO) by the VPM task, CTS ISPF application, or the CTSMail utility. A user cannot update these variables.

To make your emails and WTOs more meaningful, use the system variables. For example, you can identify the z/OS system where the email program CTSMail ran to create an email.

SYSDATE

Specifies the current system date of when the variables were referenced in MM/DD/YYYY format.

SYSDFSMS

Specifies the level of DFSMS that is installed on your system.

SYSEDATE

Specifies the current system date of when the variables were referenced in DD/MM/YYYY format.

SYSID

Specifies the SMF ID of the system you are running on.

SYSJOB

Specifies the name of the job that referenced the variables. If the variables are being referenced through the CTS System Variable List panels, this variable contains your TSO user ID.

SYSLEVEL

Specifies the z/OS version, release, modification level, and FMID.

SYSPPGM

Specifies the name of the program that appears on the EXEC statement.

SYSPPSTEP

Specifies the job-step PROC name.

SYSRBNM

Specifies the actual program name that referenced the variables.

SYSSTEP

Specifies the step name in the job that referenced the variables.

SYSTIME

Specifies the current system time of when the variables were referenced in HH:MM:SS format.

More information:

[Create an Email](#) (see page 278)

Create a WTO

To notify users of specific conditions that are tested for in VPM alerts, create a write-to-operator message (WTO) to the system consoles. Use the Volume Pool Monitor feature. You define WTOs in the CAI.SEND data set.

Follow these steps:

1. In the first line of the WTO, include the text WTO and one or more blanks.
2. Enter the ROUTCDE= parameter. You can also specify the ROUTCDE= parameter on a separate line.

The ROUTCDE parameter is a list of message routing codes that identify what types of consoles the WTO should be sent to. Enter at least one numeric code from 1 through 12 in parentheses. If you need more than one routing code, separate the values by commas only. The following IBM message routing codes are associated with tape consoles:

2

Specifies the operator information.

3

Specifies the tape pool.

5

Specifies the tape library.

Note: For a complete list of message routing codes, see the *z/OS System Messages Vol 1* guide.

3. Anywhere between columns 1 and 70, specify TEXT on a line by itself.
4. After the TEXT statement, include one or more lines of WTO text (no limit). Specify the WTO text in columns 1 through 70.

You can include variables that begin and end with the substitution character (default is &) in the WTO text. If the variables match the values in the VDB, the current values of the variables replace the variable names. If a variable is not found in the VDB, the variable name stays in the WTO text.

Text is reformatted to remove any extra blanks between words and variables. The reformatting results in a formatted WTO of one or more lines. The number of the lines depends upon the length of the data value that is associated with the variable.

5. (Optional) Include the following formatting characters:

&BL&

Terminates this line and insert a blank line.

&FM&

Turn off text formatting until the next &FM& is found.

Note: This keyword must begin in Column 1 of input. All other text on the same line is ignored.

&NL&

Terminates this line.

The WTO is created.

Every WTO that the VPM or the CTSMAIL utility issues has the same message identifier – CA\$F501I. The identifier is fixed to prevent the misuse of this utility. With the identifier in place, messages that appear to be issued by another product or system component are not created.

The ability to create a WTO can be used for other needs unrelated to tape management. If the VDB is allocated and available, the CTSMAIL utility can create WTOs for any purpose. You can use the system variables and any user variables in the text of these emails. You can include any product variable in a WTO only if the CTS VPM task updates the CA 1 product variables in the VDB.

WTO Example

For this example, member WTO001 is created in the CAI.SEND data set containing the following statements:

```
EDIT      CAI.SEND(WTO001) - 01.00                      Columns 00001 00072
Command ==>                                           Scroll ==> CSR
***** ***** Top of Data *****
000001 WTO  ROUTCDE=(2,5)
000002 TEXT
000003 A scratch tape alert has been triggered for &NL&
000004 Volume Pool = &CTS_Alert_Pool_ID& &NL&
000005 Scratch tapes available = &CTS_Alert_Variable_Value&
***** ***** Bottom of Data *****
```

Assuming that an alert was created which specified member WTO001 as the value for the Send field in the alert. When the alert condition is met, the following output is created from the WTO text.

```
17.10.56 JOB06069 CA$F501I CTSWTO Informational message 397
397              A scratch tape alert has been triggered for
397              Volume Pool = POOL_ABC
397              Scratch tapes available = 499
```

Create an Email

You can create an email in the CAI.SEND data set using the ISPF editor. You can send emails by VPM alert processing or by the CTSMAIL utility. You can test that the email looks and is distributed as you intended it to. Specify the email member as input to the CTSMAIL utility.

Follow these steps:

1. Configure the email options variables using the CTS ISPF applications. All email options variables begin with the name OPT_EMAIL_.
2. Compose the email header and the email body.

The header section includes information such as the sender, recipient, and title of the email. The email body is the actual text that you send. All header fields observe the MIME x- token convention and begin with x-. For example, the recipient of the email is identified with the following header statement:

x-TO: john.smith@xyzcorp.com

Create emails in HTML format or in plain text format. Text is the default format. To create an HTML format email, specify x-FMT: HTML in the header section. All header fields allow variable substitution except where specifically noted.

Note: For more information about the CTSMAIL utility, see the *Utilities and Reports Reference* guide.

More information:

[Volume Pool Monitor \(VPM\)](#) (see page 254)

Define Email Headers

Use the following rules in creating email members in the CAI.SEND data set. Any valid PDS member name can be used.

Follow these steps:

1. In the first nonblank record in an email, specify the text EMAIL in any position on the line.
2. Enter the header fields in any order, following the EMAIL record.

Every header field contains one value. You can specify selected header fields multiple times. Specify these header field records consecutively. If you do not specify the records, default header fields are used. The OPT_EMAIL_ variables provide default values for all required header fields.
3. (Optional) Create user variables with the ISPF CTS Variable Add panel and specify the variable name as the value on the header fields. For example, after adding a user variable MY_COMPANY_TAPE_LIBRARIAN through the CTS Variable Add panel with a valid email address. That email address is substituted when you specify the following header field:

x-TO: &MY_COMPANY_TAPE_LIBRARIAN&

The first record that does not begin with a x- header field initiates the start of the email body. Emails that are constructed correctly cause a WTO error in the range CA\$F560E to CA\$F565E.

Email Header Fields

The following email header fields are supported:

x-TO:

Defines the recipients of the email. You can replicate this field to specify more recipients. If you do not specify the value in the email, this header field uses the value that is set in the variable `OPT_EMAIL_DEFAULT_TO`.

Limits: 50 characters

x-PRNTTO:

Specifies the display name in the To: position for the sent email. If not specified, the x-TO: parameter appears by default.

Limits: 50 characters

x-FROM:

Specifies the address that defines the sender of the email. Specify a valid email address. Otherwise, the email address uses the default value from the variable `OPT_EMAIL_DEFAULT_FROM`.

Limits: 50 characters

x-PRNTFR:

Specifies the display name that appears in the From: position for the sent email. If not specified, the x-FROM: parameter is shown by default.

Limits: 50 characters

x-FMT:

Specifies the format of the email. Valid entries are HTML and TEXT. If not specified in the email, this header field uses the value that is set in the variable `OPT_EMAIL_DEFAULT_FMT`.

Limits: Four characters

x-TITLE:

Specifies the title for the Subject: line of the email. If not specified in the email, this header field uses the value that is set in the variable `OPT_EMAIL_DEFAULT_TITLE`.

Limits: 60 characters

x-SUB:

Specifies the substitution character that is used to delimit variables. Variables that you want to specify in header fields or the body of the email must begin and end with this substitution character. If not specified in the email, the value defined in `OPT_EMAIL_DEFAULT_SUBCHAR` is used.

Limits: One character

x-HELO

Specifies the TCP/IP HOST NAME of the service that is used to send the email. In multisystem environments, the service typically vary. If you do not specify x-HELO: in the email, the system uses the value that is defined in EMAIL_xxxx_HOSTNAME (where xxx is the SMF ID).

Limits: 60 characters

More information:

[Email Environment Setup](#) (see page 258)

Email Body

The first record in the member that does not have a x- header field is considered the start of the body of the email. How the body of the email is constructed depends on the value that you specify for the x-FMT: header field and the use of variables and special formatting keywords. You can use two format types for the body of the email, HTML and TEXT. You control the format by the x-FMT: header field.

HTML-formatted Email

To create an email that permits the use of HTML in formatting the body of the email, specify x-FMT: HTML in the header section. The HTML tags can control the detailed formatting. Unless another font is explicitly specified in the body of the email, the default font is used.

Plain Text-formatted Email

An email is constructed using plain text formatting when you specify x-FMT: TEXT or leave the default. In plain text, the formatting text wraps from line to line. Unless you specify either of the following formatting keywords, any extra blanks are removed.

&BL&

Specifies manual formatting. When you specify the &BL& keyword, the current line terminates and a blank line is inserted in the body of the email.

&FM&

Specifies manual formatting. Enter this keyword in column one of the email body. Any text on the same line as the keyword specification is ignored. When you specify the &FM& keyword, all text afterwards preserves the formatting and no variable substitution is performed. You can use this keyword to format small tables, reports, or data dumps that need original formatting preserved. To end the manual formatting mode, declare the FM keyword again. If not declared a second time, the format is preserved for the remainder of the email.

&NL&

Specifies a new line. Any text following this keyword prints on the next line.

If the substitution character changes with the x-SUB: header field, begin and end the formatting keywords with the new substitution character.

Variable Substitution in Emails

Variable substitution is supported in header fields and the body of both HTML and Plain Text format emails. You can specify all types of variables – user, option, product, and system variables.

Every line of text in the email body is examined for the presence of variables. Variables must begin and end with the substitution character and must begin and end on the same line. If you do not specify a substitution character in the email header using the x-SUB: header field, the value that is defined in OPT_EMAIL_DEFAULT_SUBCHAR is used. When a variable is found in the email body, the VDB is searched for an exact match on the variable name. If the match is found, the value that is associated with that variable is included in the email body in place of the variable name. If the variable is not found in the VDB, the variable name is left unchanged in the email body. All variable substitution is performed before any HTML or Plain Text formatting. No variable substitution is performed in a block of text that is being manually formatted with the &FM& keyword.

Variable names are case-sensitive. If the case does not match exactly between the variable in the email and the variable in the database, they are considered to be different variables.

Email Example 1

For this example, member EMAIL001 is created in the CAI.SEND data set containing the following statements:

```

EDIT          CAI.SEND(EMAIL001) - 01.00          Columns 00001 00072
Command ==>                                     Scroll ==> CSR
***** ***** Top of Data *****
000001 EMAIL
000002 x-TO: bill@example.com
000003 x-TITLE: Scratch tapes are low!
000004 Warning! There are only &CTS_Alert_Variable_Value& tapes
000005 left in volume pool
000006 &CTS_Alert_Pool_ID&.
000007 Please take action now to insure we have sufficient tapes for daily
000008 processing. Note - do not
000009 reply to this email.
***** ***** Bottom of Data *****

```

Assume that an alert was created which specified member EMAIL001 as the value for the "Send" field in the alert. The following email is created when the alert condition is met. The email information is defined in the headers and specified in the preceding illustration.

From: joe@example.com

To: bill@example.com

Cc:

Subject: Scratch tapes are low!

Warning! There are only 499 tapes left in volume pool ABC_Pool. Please take action now to insure we have sufficient tapes for daily processing. Note - do not reply to this email.

The VPM subtask used the email defaults for x-FMT: and x-FROM: defined in the VDB.

The value for OPT_EMAIL_DEFAULT_FORMAT is HTML. This value is used to select HTML formatting (although HTML tags are not included in the email body). The Sender (joe@example.com) was created using the value for the email option variable OPT_EMAIL_DEFAULT_FROM in the VDB.

The two variables in the text of the email (&CTS_Alert_Variable_Value& and &CTS_Alert_Pool_ID&) have been replaced with the actual values. The values are associated with the alert that was triggered because the condition in the alert was met.

Email Example 2

For this example, member EMAIL002 is created in the CAI.SEND data set containing the following statements:

```

EDIT          CAI.SEND(EMAIL002) - 01.00          Columns 00001 00072
Command ==>                                     Scroll ==> CSR
***** ***** Top of Data *****
000001 EMAIL
000002 x-SUB: -
000003 x-FROM: -MARIE_EMAIL-
000004 x-TO: bill@example.com
000005 x-TITLE: Tapes sent offsite on -SYSDATE-
000006 This email is generated nightly every 12 hours to list the number
000007 of tapes sent off site by CA 1. The most recent run of TMSVMULT
000008 identified -TMSVMUPD_TAPES_TO_OFF_SITE- tapes to be sent offsite.
000009 -FM-
000010 Contact list for CA 1 for this month
000011
000012   Joe Smith           866 555-1212
000013   Amber Logan        877 555-6789
000014 -FM-
000015 If the primary and secondary contacts listed above do not respond
000016 please contact the on call manager below: -NL-
000017 -On_Call_Manager-
***** ***** Bottom of Data *****

```

Assume that an alert was created with the member EMAIL002 as the value for the Send field. The following email is created from the email information in the headers when the alert condition is met.

From: marie@example.com

To: bill@example.com

Cc:

Subject: Tapes sent offsite on 08/13/2012

This email is generated nightly every 12 hours to list the number of tapes that CA 1 sends offsite. The most recent run of TMSVMUPD identified 32 tapes to send offsite.

Contact list for CA 1 for this month

Joe Smith 866 555-1212

Amber Logan 877 555-6789

If the primary and secondary contacts that are listed do not respond, contact the on-call manager:

Robert Smith at 877 555-4567

The VPM subtask used the email default for the x-FMT: parameter. You define the x-FMT: in the VDB that you set to HTML. The x-SUB: header field was used to change the substitution character to use dashes (-). Begin and end all variables and formatting keywords (NL and FM) with a dash (-). Do not use ampersands. The From: header field was created by replacing the user variable MARIE_EMAIL with the value in the VDB. The -FM- keyword starts and ends block formatting, which turns off all formatting in between the keywords. Block formatting is helpful when you format an email with a table or section of text that you do not want to format automatically. The last entry in the table reflects the user variable -On_Call_Manager-.

Appendix A: Integration with CA OPS/MVS EMA

This section contains the following topics:

[Overview](#) (see page 287)

[Ensure that CA OPS/MVS Is Enabled for Capturing These Events](#) (see page 288)

[CA 1 Active State Events](#) (see page 289)

[CA 1 Heartbeat Events](#) (see page 290)

Overview

CA 1 provides seamless integration with CA OPS/MVS by automatically communicating both active status events and heart beat events to CA OPS/MVS. The enabling technology for this is through a generic event API call that CA OPS/MVS provides the other mainframe products so that they can communicate events to CA OPS/MVS.

You do not need to do anything for CA 1 to enable this event communication interface to CA OPS/MVS. If CA 1 and CA OPS/MVS are active in the same z/OS image, CA 1 automatically communicates these automation events to CA OPS/MVS.

By generating active status events CA 1 and other CA products are able to communicate to CA OPS/MVS's System State Manager (SSM) component when they are starting, up, stopping or down.

SSM is a built-in feature that uses an internal relational data framework to proactively monitor and manage started tasks, online applications, subsystems, JES initiators, and other z/OS resources including your CA mainframe products. SSM compares the current state of online systems, hardware devices, and the other resources with their desired state, and then automatically makes the necessary corrections when a resource is not in its desired state. This provides proactive and reactive state management of critical resources.

Before the CA OPS/MVS interface existed, CA OPS/MVS could automate active status events for your CA products; however this typically required monitoring unique messages for each CA product. With this interface, CA OPS/MVS can capture these events for any of your CA products with a single automation event rule.

With the heart beat event, CA 1 can communicate a normal, warning, or problem overall health status and reasoning to CA OPS/MVS on a regular interval. Once CA 1 begins generating heart beat events for CA OPS/MVS, CA OPS/MVS can also react to the lack of a heart beat event from CA 1, treating this as an indication that there is either a potential problem with CA 1, or there is a larger system-level problem that is taking place.

When the TMSINIT procedure is run CA 1 issues an UP command. When TMSINIT terminates OPS/MVS is notified that CA 1 is up.

When the TMSINIT procedure is run and a password is supplied to bring CA 1 down, CA 1 issues a DOWN command. When executed in this mode, OPS/MVS is notified that CA 1 has terminated.

In most cases TMSINIT is executed to start CA 1 after an IPL or to refresh executable modules in common storage.

TMSINIT notifies OPS/MVS that it is UP and continues in this state for the life of the IPL with heartbeat events to confirm that it is active. CA 1 performs internal checks to verify its status based on a 30 minute timer and tape OPEN events. During the internal check CA 1 issues a heartbeat to report its condition to OPS/MVS. The first tape job and all subsequent tape jobs running 30 minutes after the prior heartbeat issue either a NORMAL or a PROBLEM heartbeat to OPS/MVS. This prevents CA 1 from flooding OPS/MVS with heartbeat commands during high tape OCEOV activity.

Ensure that CA OPS/MVS Is Enabled for Capturing These Events

To ensure that this CA OPS/MVS interface is active, make sure the CA OPS/MVS parameter APIACTIVE is set to its default of ON. This allows CA OPS/MVS to acknowledge and process the events generated by CA 1 and other CA products through this interface.

CA 1 Active State Events

CA 1 provides a direct interface to the CA OPS/MVS System State Manager (SSM) application to notify CA OPS/MVS of the current operating state of the given CA 1 address space. The CA OPS/MVS SSM application can use this information to automatically control the operation of the CA 1 address space, as well as any other address space that is dependent upon the CA 1 address space being active. For more information on using CA OPS/MVS SSM see the CA OPS/MVS User Guide.

The CA 1 product active state is presented to CA OPS/MVS and can be processed by the following rule:

)API CASTATE

The available OPS/REXX variables for CA 1 product state management are:

OPS/REXX Variable	Value
API.APPLICATION	CA 1
API.VERSION	Current release
API.LEVEL	00000
API.EVENTID	CASTATE
API.MSGID	CASTATE
API.TEXT	State of CA 1

The API.TEXT variable has the following format:

State of *appl_id* is *current_state*

appl_id

Specifies the same value as the API.APPLICATION variable

current_state

STARTING

Indicates that CA 1 is initializing

UP

Indicates that CA 1 is active

STOPPING

Indicates that CA 1 is terminating

DOWN

Indicates that CA 1 is exiting the system

For more information on how to use the CASTATE API, see the member SSMCAAPI of opsmvsHLQ.STATEMAN.RULES.

CA 1 Heartbeat Events

CA 1 provides a continuous heartbeat event directly to CA OPS/MVS. CA OPS/MVS can use this information in several ways to determine the operational health of the CA 1 product.

CA 1 issues a heartbeat update every nnnn seconds that notifies CA OPS/MVS of the current operational health of the CA 1 product.

If CA 1 detects a health state change, it immediately generates a heartbeat update without waiting for the nnnn second heartbeat interval to expire. In this way, CA 1 provides CA OPS/MVS with a constant operational health state view of the CA 1 product.

CA OPS/MVS can also react to the lack of a heartbeat update from CA 1 and an indication that there is either a potential problem with CA 1, or there is a larger system level problem that is taking place.

The CA 1 product heartbeat event is presented to CA OPS/MVS and can be processed by the following rule:

```
)API CAHEARTBT
```

The available OPS/REXX variables for CA 1 state management are:

OPS/REXX Variable	Value
API.APPLICATION	CA 1
API.VERSION	Current release
API.LEVEL	00000
API.EVENTID	CAHEARTBT
API.MSGID	CAHEARTBT
API.TEXT	State of CA 1

The API.TEXT variable has the following format:

appl_id Status: *heartbeat_state* Reason: reason_text

appl_id

Specifies the value of the API.APPLICATION variable.

heartbeat_state

Heart_beat_state can be one of the following:

NORMAL

Indicates that CA 1 is operating normally, without any detected problems.

WARNING

PROBLEM

reason_text

reason_text explains the problem as reported by the event API call.

For information on how you use the CAHEARTBT API, see members APIHRTB1, APIHRTB2, and APIHRTB3 of opsmvsHLQ.SAMPLE.RULES.

Appendix B: CA 1 Health Checker

This section contains the following topics:

[About the Health Checker](#) (see page 293)
[Parameter Overrides for CA 1 Checks](#) (see page 294)
[CA1_AUDIT_VRFY_WITHIN_LOW_THRSH](#) (see page 295)
[CA1_AUDIT_VRFY_WITHIN_MED_THRSH](#) (see page 296)
[CA1_FREE_DSNB_LOW_THRSH](#) (see page 297)
[CA1_FREE_DSNB_MEDIUM_THRSH](#) (see page 298)
[CA1_FREE_DSNB_QUICK_SCAN](#) (see page 299)
[CA1_TMC_AUDIT_PLACEMENT](#) (see page 300)
[CA1_USED_DSNB_FREE_CHAIN](#) (see page 301)
[CA1_VRFY_OPTION_DCHG](#) (see page 302)
[CA1_VRFY_OPTION_LCHG](#) (see page 303)
[CA1_VRFY_OPTION_TCHG](#) (see page 304)
[CA1_VRFY_MIXED_EXPDT_OPTION](#) (see page 305)
[CA1_VRFY_SECURITY_EXIT_FUNC](#) (see page 306)
[CA1_VRFY_SECURITY_EXIT_PSWD](#) (see page 307)
[CA1_VRFY_SECURITY_EXIT_YSVC](#) (see page 308)
[CA1_VRFY_SECURITY_PROFILE_CMD](#) (see page 309)
[CA1_VRFY_SECURITY_PROFILE_TAPE](#) (see page 310)

About the Health Checker

The IBM Health Checker for z/OS identifies potential problems in your z/OS environment by checking system or product parameters and system status against recommended settings. The IBM Health Checker for z/OS is structured as a framework that includes a health check started task and various separate *check* routines provided by IBM or other vendors. CA 1 provides health checks for CTAPOPTN settings, definitions, or real-time processing conditions that could result in a problem. When these conditions are found, the CA 1 checks provide detailed recommendations on how to correct the problem. The CA 1 health checks also recommend best practices for using CA 1.

Prerequisites

To take advantage of this feature, you must have the following components configured and running on your system:

- IBM Health Checker for z/OS-distributed with every supported level of z/OS
- CA Health Checker Common Service-a free CA Common Services component that provides the framework for managing the health checks plus the CAHCHECK address space where the health checks can be hosted
- Optionally Common Tape System (CTS) address space-If the CA Common Service CAHCHECK address space is not being used, then CTS must be used to host the CA 1 health checks

Parameter Overrides for CA 1 Checks

The IBM Health Checker for z/OS allows the override of selected default parameters by specifying desired defaults on the POLICY statement in the HZSPRMxx member of parmlib. This is useful in changing such values as INTERVAL to a value more appropriate for your installation. Individual checks can also be written to support parameter overrides using the PARM() parameter on the POLICY statement. For a complete list of parameters that can be overridden, see Managing Checks in *IBM Health Checker for z/OS User's Guide*.

These parameters can also be overridden by using the MODIFY command to pass the desired parameters to the IBM Health Checker for z/OS started task.

```
F hzsproc,UPDATE,CHECK(check_owner,check_name),PARM='chkparm'
```

Example:

```
F HZS,UPDATE,CHECK(CA_1,CA1_FREE_DSNB_QUICK_SCAN),  
PARM='SCANLIMIT(1000)'
```

If a CA 1 Check supports the override of a parameter, it is documented in the *Parameters Accepted* section in each check that follows. The default interval and the exception interval are provided in the *Description* section of each CA 1 check.

CA1_AUDIT_VRFY_WITHIN_LOW_THRSH

Description

This check monitors space in the CA 1 Audit file to help ensure that sufficient space is available to log all tape activity. This check is intended to provide the first alert that the Audit file may be filling up. A second check is also provided to trigger a medium level alert if space is not available in the Audit file. The default interval for this check is every 30 minutes and the exception interval is every 30 minutes.

Best Practice

Run the TMSCOPY utility regularly to help ensure that space is available. For most installations this can be done once a day. The TMSCOPY utility creates a backup of the Audit file and resets the internal pointers in the active Audit file to empty status. If this check triggers frequently, you may need to allocate a larger Audit file on disk.

Parameters Accepted

This check accepts one parameter: LOW_CHECK_THAUDT_MINUS(nn).

The value specified in LOW_CHECK_THAUDT_MINUS(nn) will be subtracted from the percentage specified in THAUDT in CTAPOPTNs to determine when to trigger this alert. This parameter must be an integer in the range of 1-30. The default is: LOW_CHECK_THAUDT_MINUS(20).

Debug Support

Yes

Verbose Support

Yes

Note: For information about TMSCOPY and Audit threshold/reset processing, see the *Utilities and Reports Reference Guide*.

Message

TMSH0101E Current Audit usage of xxxx%, is over the low threshold of xxxx%.

CA1_AUDIT_VRFY_WITHIN_MED_THRSH

Description

This is the secondary check to monitor space in the CA 1 Audit file. This check is triggered because no steps were taken after an exception was raised by the *low* check that space is running short in the Audit file. The default interval for this check is every 30 minutes and the exception interval is every 30 minutes.

Best Practice

Run the TMSCOPY utility regularly to help ensure that space is available. For most installations this can be done once a day. The TMSCOPY utility creates a backup of the Audit file and resets the internal pointers in the active Audit file to empty status. If this check triggers frequently, consider allocating a larger Audit file on disk.

Parameters Accepted

This check accepts one parameter: MED_CHECK_THAUDT_MINUS(nn).

The value specified in MED_CHECK_THAUDT_MINUS(nn) will be subtracted from the percentage specified in THAUDT in CTAPOPTNs to determine when to trigger this alert. This parameter must be an integer in the range of 1-30. The default is: MED_CHECK_THAUDT_MINUS(10).

Debug Support

Yes

Verbose Support

Yes

Reference

For more information about TMSCOPY and Audit threshold/reset processing, see the *Utilities and Reports Reference Guide* in the section titled TMSCOPY - Backing Up/Restoring the TMC and Audit Data Set.

Message

TMSH0091E Current Audit usage of xxxx%, is over the medium threshold of xxxx%.

For more information, see the *Message Reference Guide*.

CA1_FREE_DSNB_LOW_THRSH

Description

The CA 1 TMC must have free (unused) DSNB records available to track secondary files on tape. Two checks are provided to monitor the availability of free DSNBs as a percentage of the total number of DSNBs defined in the TMC. This first, *low* threshold check provides early notification that DSNBs are being used up. A second check is also provided to trigger a medium level alert, if steps are not taken to either free DSNBs by scratching tapes or add more DSNBs by *extending* the TMC. The default interval for this check is every 30 minutes and the exception interval is every 30 minutes.

Best Practice

As a best practice, you should have enough extra DSNB records formatted in the TMC to handle any unexpected increases in the number of tapes created containing secondary data sets. This check helps you determine if you should extend the TMC to add more DSNBs.

Parameters Accepted

This check accepts one parameter: LOW_CHECK_THDSNB_MINUS(nn)

The value specified in LOW_CHECK_THDSNB_MINUS(nn) will be subtracted from the percentage specified in THDSNB in CTAPOPTNs to determine when to trigger this alert. This parameter must be an integer in the range of 1-30. The default is: LOW_CHECK_THDSNB_MINUS(20).

Debug Support

Yes

Verbose Support

Yes

Reference

The TMSXTEND utility provides the ability to add DSNB records to the TMC, and other volume range or DSNB maintenance.

Note: For more information about extending the TMC, see the *CA 1 Utilities and Reports Reference Guide*.

Message

TMSH0111E Current DSNBs usage of xx%, is over the low threshold of xx%. For more information, see the *Message Reference Guide*.

CA1_FREE_DSNB_MEDIUM_THRSH

Description

This check is the secondary medium threshold check to alert users that they may be running short on free (unused) DSNB records in the TMC. This check is provided to trigger a medium level alert, if steps are not taken to either free DSNBs by scratching tapes or add more DSNBs by *extending* the TMC. The default interval for this check is every 30 minutes and the exception interval is every 30 minutes.

Best Practice

As a best practice, you should have enough extra DSNB records formatted in the TMC to handle any unexpected increases in the number of tapes created containing secondary data sets. This check helps you determine if you should extend the TMC to add more DSNBs.

Parameters Accepted

This check accepts one parameter: MED_CHECK_THDSNB_MINUS(nn)

The value specified in MED_CHECK_THDSNB_MINUS(nn) will be subtracted from the percentage specified in THDSNB in CTAPOPTNs to determine when to trigger this alert. This parameter must be an integer in the range of 1-30. The default is: MED_CHECK_THDSNB_MINUS(10).

Debug Support

Yes

Verbose Support

Yes

Reference

The TMSXTEND utility provides the ability to add DSNB records to the TMC, and other volume range or DSNB maintenance.

Note: For more information about extending the TMC, see the *Utilities and Reports Reference Guide*.

Message

TMSH0041E Current DSNBs usage of xx%, is over the medium threshold of xx%. For more information, see the *Message Reference Guide*.

CA1_FREE_DSNB_QUICK_SCAN

Description

CA 1 maintains a separate chain of active (used) and free (unused) DSNBs in the TMC. This check inspects the first n number of DSNBs in the free DSNB chain (known as the SCANLIMIT) to ensure that none of them are found to be in use. The problem of having DSNBs in use in the free chain is rare but can result from user updates being improperly done to the DSNB chains. The default interval for this check is every 30 minutes and the exception interval is every 15 minutes.

Best Practice

Updating the DSNB chains in the TMC should be avoided to prevent errors. The TMSAPEC task of the CTS address space monitors the integrity of the DSNB chains and can fix most errors found in the chains. Running the TMSAPEC task is a CA 1 best practice.

Parameters Accepted

This check accepts one parameter: SCANLIMIT. SCANLIMIT specifies the number of entries in the DSNB free chain to scan and must be an integer in the range of 500-9999. The default is: SCANLIMIT(1000).

Debug Support

Yes

Verbose Support

Yes

Reference

The Automatic Pointers Error Correction (TMSAPEC) subtask of the CTS address space is used to inspect and repair problems in the multi-volume and multi-data set chains in the TMC.

Message

TMSH0121E One or more used DSNBs were found within the first nnnn DSNBs in the free chain. Once the first of these used DSNBs has reached the top of the free DSNB chain, any attempt to create a secondary file will result in a 3YY-112 abend. For more information, see the *Message Reference Guide*.

CA1_TMC_AUDIT_PLACEMENT

Description

This check is to ensure that the CA 1 TMC and Audit data sets are placed on different volumes.

Best Practice

The Audit data set is used to recover the TMC in the case that the TMC is damaged, corrupted, or otherwise unavailable. Therefore, it should be placed on a different volume than the TMC to avoid losing the primary tape catalog and one of the files used to recover it at the same time. The default interval for this check is to run once at startup and the exception interval is every 24 hours.

Parameters Accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see Data Set Requirements in the *Programming Guide*

Message

TMSH0031E Your CA 1 Tape Management Catalog (TMC) and your Audit file are on the same volume. For more information, see the *Message Reference Guide*.

CA1_USED_DSNB_FREE_CHAIN

Description

This check is similar to the CA1_FREE_DSNB_QUICK_SCAN check, which inspects the free (unused) DSNB chain for any active (used) DSNBs. However, this check scans the entire free chain. This separate check is provided, so that a full scan can be performed less often. The default interval for this check is once a week and the exception interval is every three hours.

Best Practice

To prevent the errors, do not update the DSNB chains in the TMC. The TMSAPEC task of the CTS address space monitors the integrity of the DSNB chains and can fix most errors found in the chains. Running the TMSAPEC task is a CA 1 best practice.

Parameters Accepted

This check accepts two parameters:

SCANLIMIT

Specifies the number of DSNBs in the DSNB free chain to scan.

Limits: integer in the range 5000-10000

Default: 5000

WAITTIME

Specifies the amount of time in seconds the health check pauses after it reads the specified number of DSNBs.

Limits: integer in the range 0-300

Default: 5

Debug Support

Yes

Verbose Support

Yes

Reference

The Automatic Pointers Error Correction (TMSAPEC) subtask of the CTS address space is used to inspect and repair problems in the multivolume and multidata set chains in the TMC.

Messages

TMSH0061E Your CA 1 free DSNB chain has one or more used DSNBs in the chain. Once the first of these used DSNBs reaches the top of the free DSNB chain, any attempt to create a secondary file results in a 3YY-112 abend. For more information, see the *Message Reference Guide*.

TMSH0062E The health check terminated before the end of the DSNB free chain was reached. This termination was caused by an error in the DSNB chain. Run TMSAPEC to identify and correct the DSNB chain error. Rerun this health check after TMSAPEC has been run. Program stopped at DSNB xxxx when the last DSNB in the chain is yyyy.

TMSH0063E CA 1 terminated while this health check was running. This check has been suspended.

TMSH0064E A DSNB was encountered that is not formatted properly.

CA1_VRFY_OPTION_DCHG

Description

This check identifies possible problems associated with dynamic label change processing based on the CA 1 System Option DCHG. The default interval for this check is to run once at startup and the exception interval is every 24 hours.

Best Practice

CA recommends that you allow dynamic label change processing by setting the CA 1 System Option DCHG to YES. CA 1 can process requests that involve a density change or ion mode change without delays or problems.

Parameters accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see Density Change Action in the *Programming Guide*

Message

TMSH00131E Your CA 1 real-time processing options disallow dynamic density change. Any attempt to change the density mode of a scratch tape when opening it for output will fail. For more information, see the *Message Reference Guide*.

CA1_VRFY_OPTION_LCHG

Description

This check identifies possible problems associated with changing a label type that is based on the CA 1 System Option LCHG. The default interval for this check is run once at startup and the exception interval is every 24 hours.

Best Practice

CA recommends that you allow dynamic label change processing by setting the CA 1 System Option LCHG to YES or EXT. This will allow CA 1 to process requests that involve a label type change (for example, from AL to SL) without delays or problems. By coding EXT you have the ability to control which users can perform a label type change by setting up rules in your external security system.

Parameters accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see Label Change Action in the *Programming Guide*

Message

TMSH00141E Your CA 1 real-time processing options disallow dynamic label change. Any attempt to change the label type of a scratch tape when opening it for output will fail. For more information, see the *Message Reference Guide*.

CA1_VRFY_OPTION_TCHG

Description

This check identifies possible problems associated with changing the dynamic tape recording technique based on the CA 1 System Option TCHG. The default interval for this check is run once at startup and the exception interval is every 24 hours.

Best Practice

CA recommends that you allow the recording technique to be changed for new scratch tapes by setting the CA 1 System Option TCHG to YES. This will allow CA 1 to process requests that involve a recording technique change without delays or problems. With TCHG set to YES, for example, a 3590 volume in scratch status, which was previously created with 256 tracks, can be mounted on a 3590-H device for reuse with 384 tracks.

Parameters Accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

Note: For more information, see the *Programming Guide*.

CA1_VRFY_MIXED_EXPDT_OPTION

Description

This check alerts users of situations where the mixing expiration date is being done so that it can be followed up and corrected. The default interval for this check is run once at startup and the exception interval is every 24 hours.

Best Practice

Mixed expiration dates should be avoided as they can result in situations where a file is scratched prematurely or retained longer than necessary. CA 1 resets the expiration date in the TMC volume record, if a secondary data set is created using a higher date. This higher date is set in the volume record and becomes the controlling expiration date for all files on the volume. CA recommends that you set the CA 1 System Option MIXEXP to WARNING or DISALLOW. You can identify these situations and make appropriate changes to the application that is creating the tapes.

Parameters Accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see Mixed EXPDT Action in the *Programming Guide*

Message

TMSH0081E Your CA 1 option MIXEXP is not set to WARNING or DISALLOW. Mixing expiration dates during the creation of secondary files on a single or multi-volume data sets may result in premature scratching of volumes. For more information, see the *Message Reference Guide*.

CA1_VRFY_SECURITY_EXIT_FUNC

Description

This check is provided to warn users that basic security setup has not been performed to protect their assets on tape. This exposure is associated with the setting of the CA 1 System Option FUNC, which controls the security check to be done in the real-time nonresident (foreign) and label processing areas. The default interval for this check is run once at startup and the exception interval is every 24 hours.

Best Practice

CA recommends that tape resources be properly protected by setting the CA 1 System Option FUNC to YES or EXT and setting up rules in the external security system to control access. This check generates an exception when FUNC is set to NO which indicates that any user job reading a tape as a foreign tape (EXPDT=98000) and specifying a data set name that matches on the last 17 characters of the DSN, can access data on the tape.

Parameters Accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see Function Call Processing (FUNC) and related information in the *Programming Guide*

Message

TMSH0011E Your CA 1 security options allow unauthorized users to access privileged tape data that they would otherwise be unable to access. Any user specifying EXPDT=98000 can access any tape and read or update it. For more information, see the *Message Reference Guide*.

CA1_VRFY_SECURITY_EXIT_PSWD

Description

This check raises an exception when it finds that security has not been set up for the CA 1 ISPF panels. The default interval for this check is to run once at startup and the exception interval is every 24 hours.

Best Practice

The CA 1 ISPF online system provides access to critical data about the volumes and files protected and managed by CA 1. If this data is not properly protected through calls to the external security system, tape data can be expired prematurely and critical tape data can be lost. The CA 1 System Option PSWD controls whether calls are made to the external security system to restrict access to the CA 1 ISPF online. CA strongly recommends that PSWD should be set to YES, and rules to control access should be set up in your external security system.

Parameters Accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see Password Call Processing (PSWD) and related information in the *Programming Guide*

Message

TMSH0051E Your CA 1 security options allow unauthorized users access to the CA 1 ISPF online system passwords. Anyone with access to the CA 1 ISPF interface may be able to access the TMC. For more information, see the *Message Reference Guide*.

CA1_VRFY_SECURITY_EXIT_YSVC

Description

This check verifies that access to the TMC is protected by the setting of CA 1 System Option YSVC and rules in the external security system. The default interval for this check is to run once at startup and the exception interval is every 24 hours.

Best Practice

The Tape Management Catalog (TMC) contains critical data about the volumes and files protected and managed by CA 1. If this data is not properly protected through calls to the external security system, tape volumes can be expired prematurely and critical tape data can be lost. The CA 1 System Option YSVC controls access to the CA 1 SVC and the volume and file data in the TMC. We recommend that access to the CA 1 SVC and TMC should be controlled by setting this option to YES and creating access rules in the external security system.

Parameters Accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see CA 1 SVC Call Processing (YSVC), Batch Call Processing (BATCH), and related information in the *Programming Guide*

Message

TMSH0021E Your CA 1 security options allow unauthorized users to update your Tape Management Catalog (TMC) using the YSVC services of CA 1. Any user can access and update any record in the TMC. For more information, see the *Message Reference Guide*.

CA1_VRFY_SECURITY_PROFILE_CMD

Description

This check is provided to warn users that basic security setup has not been performed to control access to tapes and commands. This exposure is associated with defining the resource class CA@MD. If the external security is set to IBM RACF but the CA@MD resource class is not defined, users are not able to issue commands in the CA 1 ISPF panels. The default interval for this check is run once at startup and the exception interval is every 24 hours.

Best Practice

When you set the CA 1 external security option to YES, you are requesting the external security system to control access to tapes and commands. In order for the external security system to allow use of commands by the user in the CA 1 ISPF panels, define the CA@MD resource class. This check generates an exception when RACF is active but resource class CA@MD is not defined.

Parameters Accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see the *Programming Guide*.

Message

TMSH0171E RACF External Security is active and the CA 1 external security option is set to YES, but resource class CA@MD resource class has not been defined.

CA1_VRFY_SECURITY_PROFILE_TAPE

Description

This check is provided to warn users that basic security setup has not been performed to protect against unauthorized access to tapes. This exposure is associated with defining the resource class CA@APE. If the external security is set to IBM RACF but the CA@APE resource class is not defined, tape jobs begin to fail. The default interval for this check is run once at startup and the exception interval is every 24 hours.

Best Practice

In order for the external security system to allow access to tapes, define the resource class of CA@APE. Then grant individual users access to resources within that class. This check generates an exception when RACF is active but resource class CA@APE is not defined.

Parameters Accepted

This check does not accept any parameters.

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see the *Programming Guide*.

Message

TMSH0161E RACF External Security is active and the CA 1 external security option is set to YES, but resource class CA@APE has not been defined.

Index

3

- 3480/3490 • 47
 - cartridge tape drive • 47

A

- Activating • 14, 57
 - CA 1 • 14
 - TMC Volume records • 57
- ADGDG • 84
 - parameter • 84
- AIVS • 48, 50, 51
 - duplicate volume tracking • 50
 - external volser • 51
 - foreign volume tracking • 50
 - realtime processing • 48
 - specific mount message • 48, 51
- Alternate Internal Volume Substitution • 50
 - AIVS • 50
 - foreign volume tracking • 50
- Analyzing problems • 64, 65, 66
 - data recovery • 65
 - NOT SCRATCH demounts • 64
 - scratch tapes • 65
 - using reporting programs • 66
 - when tape failures occur • 65
 - when tapes are not scratched • 64
- Audit • 33, 95
 - related field updates • 95
 - threshold • 33
- Audit data set • 33, 38
 - backup • 33
 - reports (TMSAUDIT) • 38
 - threshold • 33
- Automatic • 199, 226
 - startup • 199, 226
 - CTS • 226
 - DBS • 199
- Auxiliary disposition • 67, 69, 70, 71, 72, 73, 74
 - batch • 72, 73, 74
 - control statements • 73
 - example • 74
 - JCL • 73
 - uncataloging data sets • 72
 - console messages • 67

- introduction to • 67
- realtime • 69, 70, 71
 - control statements • 70
 - example • 71
 - JCL • 69

B

- Batch • 16, 22, 23, 38, 67, 72, 73, 74
 - auxiliary disposition • 67, 72, 73, 74
 - control statements • 73
 - example • 74
 - introduction to • 67, 72
 - JCL • 73
 - uncataloging data sets • 72
- inquiry report (TMSBINQ) • 38
- processing • 16, 22, 23
 - daily • 16
 - monthly • 23
 - weekly • 22
- BTLS • 178
 - 3495/3494 • 178
 - eject cartridges • 178
 - synchronize • 178
- synchronize • 178
 - tape management catalog • 178
- Build TMC record • 55, 57, 58, 59, 112
 - maintenance • 57, 58, 59
 - adding DSNB records • 58
 - adding TMC volume ranges • 58
 - extending the TMC • 58
 - removing TMC volume ranges • 59
 - updating, accessing • 55
 - using CA 1 ISPF panels • 112

C

- CA 1 • 16, 100, 136, 137
 - daily batch processing • 16
 - online inquiry and update • 100, 137
 - ISPF • 100
 - TIQ Online Inquiry/Update • 137
 - tape inquiry display • 136
 - ISPF • 136
- CA 1 ISPF • 100, 101, 102, 103, 105, 106, 107, 109, 112, 120, 122
 - build new TMC records • 112

-
- AIVS • 112
 - FOREIGN • 112
 - help/tutorial • 101
 - inquiry • 103, 107, 109
 - control records • 109
 - DSNB number • 107
 - volume serial number • 103
 - inquiry/update • 102, 103, 120, 122
 - configuration tables • 122
 - session options • 102
 - TMC records • 103, 120
 - TMC records, ERASE • 120
 - TMDEGAU • 120
 - panel description • 100
 - request • 105, 106
 - multiple file listing • 106
 - multivolume listing • 105
 - utility commands • 120
 - CA Common Services for z/OS and OS/390 • 224
 - services • 224
 - CAICCI, CTS • 224
 - CA Dynam/T interface • 159, 160, 161
 - communications • 160
 - database • 161
 - defining volumes • 161
 - synchronization • 161
 - introduction to • 159
 - queue • 160
 - allocation • 160
 - locking • 160
 - processing • 160
 - serialization • 160
 - TMSFORMT • 161
 - utilities • 160, 161
 - DYNCOMM • 161
 - DYNEXTRA • 161
 - DYNFORMAT • 160
 - DYNRELSE • 161
 - DYNSYNC • 160, 161
 - DYNUPDTE • 161
 - vault management • 159
 - volume ownership • 159
 - CA Earl • 39, 84
 - customized reports • 39
 - TMS Report • 84
 - 25 - Distribution List for Vault • 84
 - 25 - Inventory List for Vault • 84
 - 25 - Picking List for Vault • 84
 - VMS overview • 84
 - CAI.CTAPOPTN member • 14, 43, 62
 - CAUTOCMD • 14
 - TMONSMxx • 43
 - defining tape pool access rules • 43
 - TMOOPTxx • 62
 - LAB parameter • 62
 - TMOSCRxx • 43
 - defining tape pool access rules • 43
 - Catalog control • 16, 26, 77
 - LABEL=EXPDT=99000 • 16, 26
 - processing • 77
 - TMC and system catalog synchronization • 77
 - Chaining • 59
 - multi-data set • 59
 - multivolume data sets • 59
 - TMSPTRS utility • 59
 - Common Tape System (CTS) • 187
 - interface • 187
 - CTS • 189, 199, 200, 203, 216, 218, 220, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 293
 - accessing user data • 216
 - automatic startup • 226
 - CA Common Services for z/OS and OS/390 Services CAICCI • 224
 - command descriptions • 229, 230, 231, 232
 - CANCEL • 232
 - DISPLAY • 230
 - FORCE • 232
 - MSG • 232
 - SET, options • 231
 - SET, subtasks • 231
 - SPINOFF • 231
 - START • 230
 - STATUS • 232
 - STOP • 230
 - DBS subtask • 199, 200
 - automatic startup • 199
 - commands • 200
 - CTS message command • 200
 - manual startup • 199
 - startup procedure • 199
 - description • 228
 - initialization procedures • 223
 - LAB commands • 227
 - LAB description • 225
 - manual startup • 226
 - output labels • 223
 - processing steps • 203
 - realtime auxiliary disposition • 216
-

- sample labels • 220
- started task • 189, 218
 - example • 189
 - output assignments • 218
- startup procedure • 225
- Cycle control • 16, 25
 - LABEL=EXPDT=99ccc • 16, 25

D

- Daily batch processing • 16
 - IDCAMS/IEHLIST • 16
 - reporting programs • 16
 - TMSCLEAN utility • 16
 - TMSCOPY utility • 16
 - TMSCTLG utility • 16
 - TMSCYCLE utility • 16
 - TMSEXPDT utility • 16
 - vault management • 16
- Data Base Services (DBS) • 198
 - subtask • 198
 - TMSINIT • 198
- Data set(s) • 16, 24
 - retention • 16, 24
 - resetting expiration dates • 24
- Days since last used control • 16, 28
 - LABEL=EXPDT=98ddd • 16, 28
- DBS • 198, 199
 - automatic startup • 199
 - data base services • 198
 - subtask • 198
 - startup procedure • 199
- Deactivating • 57
 - TMC Volume records • 57
- Default • 24
 - retention • 24
- DYNCOMM • 161
 - utility • 161
- DYNEXTRA • 161
 - utility • 161
- DYNFORMAT • 160
 - utility • 160
- DYNRELSE • 161
 - utility • 161
- DYNSYNC • 160, 161
 - utility • 160, 161
- DYNUPTDE • 161
 - utility • 161

E

- EXPDT keyword in JCL • 16, 25, 26, 28
 - LABEL=EXPDT=98ddd • 16, 28
 - LABEL=EXPDT=99000 • 16, 26
 - LABEL=EXPDT=99ccc • 16, 25
- Extending • 58
 - TMC • 58
- External • 51
 - volser • 51
 - AIVS • 51

F

- Field updates • 95
 - audit related fields • 95
- Flowcharts • 16, 22, 23
 - daily batch processing • 16
 - monthly batch processing • 23
 - weekly batch processing • 22
- Foreign tapes • 51
 - controlling • 51
 - ISPF panels • 51
- Formatting • 34
 - Audit data sets • 34

G

- Generalized Report Writer • 39
 - CA EARL • 39

I

- IBM WTORs • 52
 - IEC534D • 52
 - IEC704A • 52
- Initialization • 14, 60
 - CA 1 • 14
 - tape volumes • 60
- Initializing • 156, 157, 158
 - TIQ • 156, 157, 158
 - from an MVS Console • 156
 - using CA Roscoe • 158
 - using TSO • 157
- ISPF • 51, 95, 100, 101, 102, 103, 120, 122
 - audit related field updates • 95
 - help/tutorial • 101
 - inquiry/update • 102, 103, 120, 122
 - configuration tables • 122
 - session options • 102
 - TMC records • 103, 120

- panel description • 100
- panels • 51
 - foreign tapes • 51
- utility commands • 120

ISPF DSLIST • 136

- Tape inquiry • 136
- status check • 136

K

Key Tape procedure • 32

- non-CA 1 system • 32

KEYTAP • 32

- parameter • 32

Keyword expiration • 16, 25, 26, 28

- JCL keywords • 16, 25, 26, 28
 - LABEL=EXPDT=98ddd • 28
 - LABEL=EXPDT=99000 • 16, 26
 - LABEL=EXPDT=99ccc • 16, 25

L

LAB • 62, 225, 227

- CTS message command • 227
- description • 225
 - CTS • 225
- parameter • 62

Label • 61, 62, 143, 202

- methods for creating • 61, 62, 143
 - auxiliary disposition • 62
 - CA 1. system option • 62
 - EXLBL • 143
 - online facilities • 62
 - TMSLBLPR • 61
- modifying • 202

M

Maintenance • 55, 57, 58, 59, 60, 61

- activating/deactivating TMC Volume records • 57
- adding • 58
 - DSNB records • 58
 - TMC volume ranges • 58
- analyzing multivolume/multi-data set chains • 59
- extending the TMC • 58
- initializing tape volumes • 60
- inquiring on/updating TMC records • 55
- introduction to • 55
- processing header label information • 61
- producing external tape labels • 61
- removing • 59

- TMC volume ranges • 59
- reporting on tape volumes • 59

Manual startup • 226

- CTS • 226

Modifying • 215

- labels • 215

Multi-data set • 59

- chains • 59
- volumes • 59

Multivolume • 59

- chains • 59
- data sets • 59

N

NOT SCRATCH • 64

- demounts • 64

O

Online • 63, 95, 100, 101, 102, 103, 120, 122, 137, 138, 139, 144, 145, 146, 147, 156, 202

- ISPF • 95, 100, 101, 102, 103, 120, 122
 - help/tutorial • 101
 - inquiry/update of configuration tables • 122
 - inquiry/update of session options • 102
 - inquiry/update of TMC records • 103, 120
 - introduction to • 95, 100
 - panel description • 100
 - utility commands • 120
- label • 63, 202
 - interface • 63, 202
- TIQ • 95, 137, 138, 139, 147, 156
 - accessing TMC records • 137
 - console-unique procedures • 156
 - driver programs • 138
 - inquiring on TMC records • 139
 - introduction to • 95, 137
 - specific field access • 138
 - utility commands • 147
- TIQ updating • 144, 145, 146
 - DSNB records • 146
 - TMC records • 144
 - volume records • 145

OUTCODE • 81

- pseudo-vaults • 81

Output • 223

- labels • 223
 - CTS • 223

Overriding • 24

EXPDT or RETPD values in JCL • 24

P

Parameter • 28

UNCATA • 28

Processing • 203

steps • 203

CTS • 203

Producing • 39

customized reports • 39

R

Realtime • 41, 48, 49, 50, 51, 67, 68, 69, 70, 71

auxiliary disposition • 67, 68, 69, 70, 71

control statements • 70

example • 71

introduction to • 67, 68

JCL • 69

processing • 41, 48, 49, 50, 51

AIVS • 48, 51

auxiliary disposition • 49

modifying • 41

NL and BLP tape • 49

nonresident tape • 50

reject for processing • 49

release changing • 41

specific mount message • 41, 48

temporary data set • 49

Removing • 59

TMC Volume ranges • 59

Reporting • 16, 37, 38, 39

programs • 16, 38, 39

CA Earl • 39

TMEDSN02 • 16

TMEJOB01 • 16

TMEVSNM • 16

TMSAUDIT • 16, 38

TMSBINQ • 38

TMSPTRS • 39

TMSPULL • 16

Vault Management System (VMS) • 37

VMS • 37

CA Earl • 37

CA Easytrieve • 37

Reports, TMS • 35, 84

24 - Exceptional Conditions Encountered During Processing • 84

25 - Distribution List for Vault • 84

25 - Inventory List for Vault • 84

25 - Picking List for Vault • 84

26 - Vault Update Listing • 84

31 - Backup/Restore the TMC File • 35

Resetting • 24

expiration dates (TMSEXPDT) • 24

Restoring the TMC • 34

reallocate • 34

Retention • 16, 24

Data Set (RDS) • 16, 24

daily batch processing • 16

resetting expiration dates • 24

S

Sample • 220

labels, CTS • 220

Scratch pool management • 28, 42, 43, 44, 45, 46

assigning tapes to scratch pools • 28

assignment hierarchy • 44

defining tape pool access rules • 43

describing the scratch pool • 43

handling output requests • 46

introduction to • 42

pulling scratch tapes • 43

realtime stacking • 45

restrictions • 44

storing and identifying scratch tapes • 42

tape mount messages • 45

see=RetentionDataSet RDS • 16

Specific mount message • 48, 51

AIVS • 48, 51

realtime processing • 48

Startup • 199, 225

procedure • 199, 225

CTS • 225

DBS • 199

Synchronizing • 77, 79

TMC and system catalog • 77, 79

identifying discrepancies • 79

introduction to • 77

using catalog control • 77

System options • 16, 24, 25, 26, 28, 32, 33, 62, 78, 79, 84, 137

ADGDG parameter • 84

CDAY parameter • 25

CJOB parameter • 25

CYD parameter • 84

KEYTAP parameter • 32

- LAB parameter • 62
- OCATLG parameter • 79
- R9 parameter • 26
- RC parameter • 16, 25
- RO parameter • 24
- RP parameter • 24
- THAUDT parameter • 33
- TW parameter • 137
- UNCATA parameter • 28, 78

T

- Tape • 59, 178
 - Management Catalog • 178
 - 3595/3494, BTLS • 178
 - volume labels • 59
- Tape inquiry • 136
 - status check • 136
 - introduction to • 136
 - ISPF DSLIST • 136
- Tape library • 55, 57, 58, 59, 60, 61
 - maintenance • 55, 57, 58, 59, 60, 61
 - activating/deactivating TMC Volume records • 57
 - adding DSNBrecords • 58
 - adding TMCvolumeranges • 58
 - analyzing multivolume/multi-data set chains • 59
 - extending the TMC • 58
 - initializing tape volumes • 60
 - inquiring on/updating TMC records • 55
 - introduction to • 55
 - processing header label information • 61
 - producing external tape labels • 61
 - removing TMC Volume ranges • 59
 - reporting on tape volumes • 59
- Tape Management Catalog (TMC) • 33, 34, 55, 57, 58, 59
 - activating TMC Volume records • 57
 - adding, DSNB records • 58
 - adding, TMC volume ranges • 58
 - backing up the TMC • 33
 - deactivating TMC Volume records • 57
 - extending the TMC • 58
 - inquiring on/updating TMC records • 55
 - maintaining chains • 59
 - multi-dataset • 59
 - multivolume • 59
 - removing • 59
 - TMC volume ranges • 59
 - restoring • 34
- Tapes • 43, 59
 - pool access rules • 43
 - volume • 59
 - labels • 59
- TIQ online inquiry/update • 137, 138, 139, 140, 141, 142, 143, 144, 147, 156, 157, 158
 - accessing TMC records • 137
 - console-unique procedures • 156
 - see also initializing/terminating TIQ • 156
 - creating external labels • 143
 - driver programs • 138
 - initializing/terminating TIQ • 156, 157, 158
 - from an MVS console • 156
 - using CA Roscoe • 158
 - using TSO • 157
 - inquiring on control records • 140
 - inquiring on TMC records • 139, 140, 141, 142
 - by cataloged data set name • 141
 - by DSNB number • 140
 - by volume serial number • 139
 - using specific fields • 142
 - updating TMC records • 144
 - utility commands • 147
- TMC • 37
 - reporting on tape status • 37
- TMOOPTxx • 62
 - LAB parameter • 62
- TMOSCRxx • 28, 43
 - defining • 43
 - tape pool access rules • 43
 - scratch pool management • 28
 - assigning tapes • 28
- TMS Reports • 35, 84
 - 23 - Vault Pattern Data Set • 84
 - 24 - Exceptional Conditions Encountered During Processing • 84
 - 25 - Distribution List for Vault • 84
 - 25 - Inventory List for Vault • 84
 - 25 - Picking List for Vault • 84
 - 26 - Vault Update Listing • 84
 - 31 - Backup/Restore the TMC File • 35
- TMSAUDIT • 16, 38, 66
 - analyzing problems • 66
 - AUDB4 • 66
 - utility • 16, 38
- TMSBINQ • 38, 55
 - overview • 38

- reporting on TMC records • 55
- TMSCLEAN • 16, 28, 31, 78
 - daily batch processing • 16
 - prerequisites for scratching tapes • 28
 - scratch processing • 28
 - SCRATCHLIST parameter • 28
 - uncatalog • 78
 - processing • 78
 - uncatalog processing • 78
 - utility • 28, 31
 - reporting • 31
- TMSCOPY • 16, 32, 33, 34, 35
 - Audit data set threshold • 33
 - backing up • 33
 - Audit data set • 33
 - TMC • 33
 - formatting Audit data set • 34
 - TMSFORMT • 34
 - TMS Report • 35
 - 31 - Backup/Restore the TMC File • 35
 - utility • 16, 32
- TMSCTLG • 16, 26, 27, 28
 - utility • 16, 26, 27, 28
 - listing mode • 27
 - reporting • 27
 - two-day hold • 28
- TMSCYCLE • 16, 25, 26
 - utility • 16, 25
 - reporting • 25
- TMSDELET • 57
 - utility • 57
- TMSDISP • 67, 68, 72
 - ddname • 67, 68
 - utility • 72
- TMSEXPDT • 16, 24, 25
 - utility • 16, 24, 25
 - reporting • 25
- TMSFORMT • 57, 161
 - activating TMC Volume records • 57
 - CA Dynam/T interface • 161
 - deactivating TMC Volume records • 57
- TMSINIT • 14
 - activating CA 1 • 14
 - CAIRIM • 14
- TMSLBLPR • 61
 - utility • 61
- TMSOSCAT • 79
 - new bit • 79
 - DSNBISCA • 79

- TMISCAT • 79
 - utility • 79
- TMSPTRS • 39, 59
 - utility • 39, 59
 - multivolume/multi-data set chains • 59
- TMSPULL • 16
 - utility • 16
- TMSTPNIT • 59, 60
 - initializing tape volumes • 60
 - utility • 59
- TMSTPPRO • 59, 61
 - processing header label information • 61
 - utility • 59
- TMSVMEDT • 16, 36, 84
 - TMS Report • 84
 - 23 - Vault Pattern Data Set • 84
 - utility • 16, 36
 - VMS overview • 84
- TMSVMUPD • 16, 37, 84
 - TMS Report • 84
 - 26 - Vault Update Listing • 84
 - utility • 16, 37
 - VMS overview • 84
- TMSVMVLT • 16, 36, 84
 - condition code checks • 36
 - TMS Report • 84
 - 24 - Exceptional Conditions Encountered During Processing • 84
 - utility • 16
 - VMS overview • 84
- Tracking • 32
 - CA 1 tapes • 32

U

- UNCATA • 28, 78
 - parameter • 28, 78
- Uncataloging data sets • 28, 72
 - batch auxiliary disposition • 72
 - from the MVS catalog • 28
- Utility • 16, 24, 25, 160, 161
 - DYNCOMM • 161
 - DYNEXTRA • 161
 - DYNFORMT • 160
 - DYNRELSE • 161
 - DYNSYNC • 160, 161
 - DYNUPDTE • 161
 - TMEDSN02 • 16
 - TMEJOB01 • 16

- TMEVSNM • 16
- TMSAUDIT • 16
- TMSCLEAN • 16
- TMSCOPY • 16
- TMSCTLG • 16
- TMSCYCLE • 25
- TMSEXPDT • 16, 24
- TMSPULL • 16
- TMSVMEDT • 16

- picking list • 89

V

- Vault Management System (VMS) • 16, 36, 37, 81, 82, 84, 85, 86, 87, 92, 93, 94, 159
 - CA Dynam/T interface • 159
 - criteria for • 86, 87
 - rejection • 87
 - selection • 86
 - executing • 36
 - features • 81, 82
 - 3480 cartridge subvaulting • 81
 - extensive reporting • 82
 - on-site protection • 81
 - introduction to • 81
 - overview • 16, 84
 - TMSVMEDT • 16, 84
 - TMSVMUPD • 16, 84
 - TMSVMVLT • 16, 84
 - program execution • 84, 85
 - rejection criteria • 87
 - reporting • 37
 - selection criteria • 86
 - special considerations • 93, 94
 - compressing off-site vaults • 93
 - removing data sets from VMS control • 94
 - testing vault pattern changes • 37
 - using vaulted tapes • 92
 - preventing tapes from being revaulted • 92
 - recommended procedure • 92
 - without checking them in • 92
- Vault Pattern • 37
 - description (VPD) • 37
 - data set • 37
- Vault Pattern Description (VPD) data set • 37
 - testing pattern changes • 37
- Vault reports • 88, 89, 90, 91
 - distribution list • 90
 - examples • 88
 - inventory list • 91